Alibaba Cloud MaxCompute

Tools and Downloads

Issue: 20190612

Legal disclaimer

Generic conventions

Table -1: Style conventions

Style	Description	Example
⛔	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	⛔ Danger: Resetting will result in the loss of user configuration data.
⚠	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	⚠ Warning: Restarting will cause business interruption. About 10 minutes are required to restore business.
📋	This indicates warning informatio n, supplementary instructions, and other content that the user must understand.	❗ Notice: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	📋 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus , page names, and other UI elements.	Click OK.
`Courier font`	It is used for commands.	Run the `cd / d C :/ windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*

Style	Description	Example
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich` *{stand | slave}*

Contents

1 Client

This article describes how to use the basic functions of the MaxCompute using client command line tool. Before using the MaxCompute client, you must *install and configure the client* .

Note:

- Do not perform the analysis operation based on the output format of the client. The output format of the client is not ensured for forward compatibility. Clients of different versions are different in their command formats and behaviors.
- For more information about basic commands of the client, see *Basic commands*.
- *Click here* to download the new version of MaxCompute client.
- The client supports JDK 1.9 from the 0.28.0 version, and the previous version can only use JDK 1.8.
- The client supports MaxCompute 2.0 from the 0.27.0 version *New data type*.

After the client is installed and configured, you can use a command line to perform the following operations.

Get Help

To view the help information of the console, the command format is as follows:

```
odps  @>./ bin / odpscmd - h ;
```

You can also input `h ;` or `help ;`(case-insensitive) in an interactive mode.

The console also provides the `help  [ keyword ];` command to get the command prompts related to the keyword. For example, input `help   table ;` to get command prompts related to the table operation as follows:

```
odps  @  odps >  help   table ;
Usage :  alter   table   merge   smallfiles
Usage :  show   tables  [ in ]
      a   list   of   tables  |  ls  [- p ,- project ]
Usage :  describe   deserves   mention  | [.] [ partition  ()]
Usage :  read  [.]  [ partition  ()] [ line_num ]
```

Start parameters

When start the console, you can specify a series of parameters as follows:

```
Usage :  odpscmd  [ option ]...
where   options   include :
   -- help  (- h )  for   help
   -- project =  use   project
   -- endpoint =  set   endpoint
   - u - p   user   name   and   password
   - k   will   skip   begining   queries   and   start   from
specified   position
   - r   set   retry   times
   - f  <" file_path ;">  execute   command   in   file
   - e  <" command ; [ command ;]...">  execute   command ,  include
  sql   command
   - C   will   display   job   counters
```

Take the -f parameter as an example, the operation is as follows:

1. Prepare the local script file script.txt. Suppose that the file is located in the disk D, and the content is shown as follows:

   ```
   DROP   TABLE   IF   EXISTS   test_table  _mj ;
   CREATE   TABLE   test_table  _mj  ( id   string ,  name   string );
   DROP   TABLE   test_table  _mj ;
   ```

2. Run the following command:

   ```
   odpscmd \ bin > odpscmd  - f   D :/ script . txt ;
   ```

Interactive mode

Run the console to directly enter the interactive mode:

```
[ admin : ~]$ odpscmd
 Aliyun   ODPS   Command   Line   Tool
 Version   1 . 0
@ Copyright   2012   Alibaba   Cloud   Computing   Co .,  Ltd .  All
 rights   reserved .
 odps @  odps >  INSERT   OVERWRITE   TABLE   DUAL   SELECT  *  FROM
 DUAL ;
```

Enter the command at the cursor position (use a semicolon as a statement terminator), and press Enter to run.

Continuous running

- When using -e or -f option to run a command, if there are multiple statements, and you want to start running from a middle statement, you can specify the parameter -k, indicating to ignore the previous statements and to start running from the specified position. When the parameter <= 0 is specified, the execution starts from the first statement.
- Each statement separated by a semicolon is considered as a valid statement. The statements which run successfully or fail to run are printed out at runtime.

For example,

suppose there are three SQL statements in the file /tmp/dual.sql:

```
drop   table   dual ;
create   table   dual  ( dummy   string );
insert   overwrite   table   dual   select   count (*)  from   dual ;
```

To ignore the first two statements, and start running from the third statement, the command format is as follows:

```
odpscmd  - k   3  - f   dual . sql
```

Get current logon user

To get current logon user, the command format is as follows:

```
whoami ;
```

Use example:

```
odps @  hiveut > whoami ;
Name :  odpstest @ aliyun . com
End_Point :  http :// service . odps . aliyun . com / api
Project :  lijunsecur  itytest
```

Use the preceding command to get the current logon user Alibaba Cloud account, endpoint configuration, and project name.

Exit

To exit the console, the command format is as follows:

```
odps @ >  quit ;
```

You can also use the following command to exit the console:

```
odps @ >  q ;
```

2 MaxCompute Studio

2.1 What is Studio

MaxCompute Studio is a big data integrated development environment (IDE) tool that is provided by the Alibaba Cloud MaxCompute platform and installed on the developer's client. It is a development plug-in based on the popular integrated development platform *IntelliJ IDEA*, helping users develop data conveniently. This article describes functional interfaces and common application scenarios of MaxCompute Studio.

Basic user interface

MaxCompute Studio is a plug-in on the IntelliJ IDEA platform, which shares basic development interfaces with IntelliJ IDEA. For more information about the IntelliJ IDEA interfaces, see *the Interface operation guide*.

Based on the IntelliJ IDEA interfaces, MaxCompute Studio provides the following functional interfaces.

- SQL Editor: Provides features such as SQL syntax highlighting, code complementing, real-time error prompting, local compilation, and job submission.

  Compiler View: Displays locally compiled prompts and error messages, and locates the code in the editor.
- Project Explorer: Connects to a MaxCompute project, and browses table structures, custom functions, and resource files in the project.

  Table Details View: Displays details and sample data of tables, views, and other resources.
- Job Explorer: Browses and searches for historical jobs of MaxCompute.
  - Job Details View: Displays running details of a job, including the execution plan and details of each execution task.
  - Job Output View: Displays output information of a running job.
  - Job Result View: Displays the output result of the SELECT job.
- MaxCompute Console: Integrates the *MaxCompute client*, on which MaxCompute client commands can be input and executed.

Connect to MaxCompute project

Before using most features of MaxCompute Studio, you must *Create a project connection*. After the project connection is created, you can view related data structures and resource information in the Project Explorer. MaxCompute Studio automatically creates a local metadata backup task for each project to increase the access frequency to MaxCompute metadata and reduce the latency.

Note:

- You must specify the target project connection to modify SQL scripts, submit jobs, view job information, open the MaxCompute console, and implement other functions using MaxCompute Studio. Therefore, creating a connection to the MaxCompute project is necessary.
- For more information about MaxCompute projects, see *Project*.
- For more information about project management using MaxCompute Studio, see *Project space connection management*.

Manage data

You can use the Project Explorer of MaxCompute Studio to quickly browse table structures, custom functions, and resource files in the project. The tree control can be used to list data tables, columns, partition columns, virtual views, custom functions, function signatures, and resource files and types of all project connections. It also supports fast locating.

You can double-click a data table to open the Table Details View and view metadata, structure, and sample data of the data table. If you do not have the permission for a project, an error message is prompted.

MaxCompute Studio integrates *MaxCompute Tunnel* and supports local data upload and download. For more information, see *Import and export data*.

Write SQL scripts

You can easily compile a MaxCompute SQL script on MaxCompute Studio.

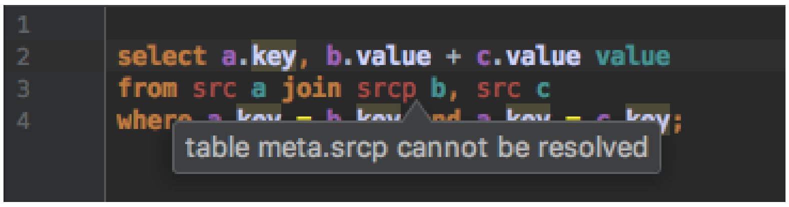
1. Open MaxCompute Studio and select File > New > Project or File > New > Module⋯.
2. Create a MaxCompute Studio project or module.

3. Select File > New > MaxCompute Script or right-click the menu and select New > MaxCompute Script , to create a maxcompute SQL script file.

   Note:
   When a MaxCompute SQL script is created, MaxCompute Studio prompts you to select an associated MaxCompute project. You can also modify the associated project using the project selector on the right of the toolbar on the SQL editor. The editor automatically checks metadata (such as the table structure) and reports errors of an SQL statement based on the project associated with the SQL script. The editor also sends the SQL statement to the associated project for execution when it submits the SQL statement for running. For more information, see *Compile an SQL script*.

SQL code intelligent prompt

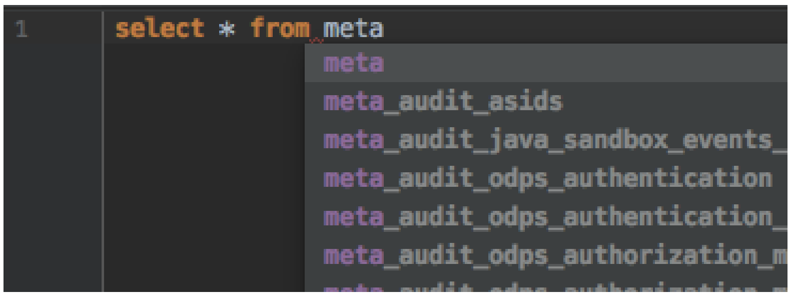
After you enter the code, the SQL editor provided by MaxCompute Studio intelligently prompts the syntax errors, type matching errors, or warnings of SQL statements, and marks them on the code in real time. as shown in the following figure.



```
select a.key, b.value + c.value value
from src a join srcp b, src c
where a.key = b.key and a.key = c.key;
```

table meta.srcp cannot be resolved

By using the code complementing function, MaxCompute Studio prompts you the name, table, field, function, type, and code keyword of a project based on the code context, and automatically complements the code based on your selections. as shown in the following figure.



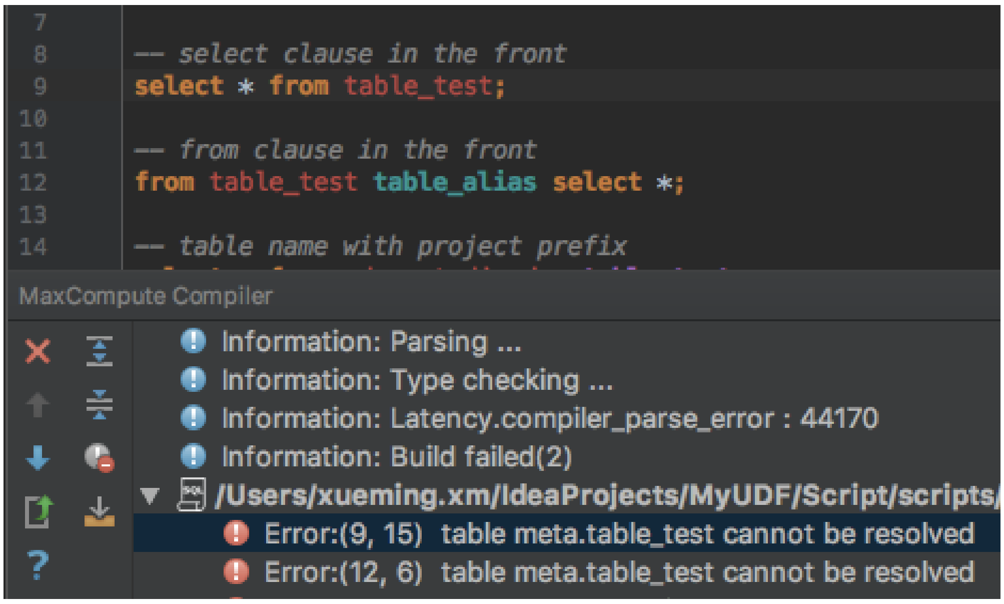
```
select * from meta
```

meta
meta_audit_asids
meta_audit_java_sandbox_events_
meta_audit_odps_authentication
meta_audit_odps_authentication_
meta_audit_odps_authorization_m

Compile and submit a job

- Compile a job

  Click the icon on the toolbar of the SQL editor to locally compile an SQL script. If syntax or semantic errors occur, the editor reports it.



```
-- select clause in the front
select * from table_test;

-- from clause in the front
from table_test table_alias select *;

-- table name with project prefix
```

MaxCompute Compiler

```
Information: Parsing ...
Information: Type checking ...
Information: Latency.compiler_parse_error : 44170
Information: Build failed(2)
/Users/xueming.xm/IdeaProjects/MyUDF/Script/scripts/
    Error:(9, 15)  table meta.table_test cannot be resolved
    Error:(12, 6)  table meta.table_test cannot be resolved
```

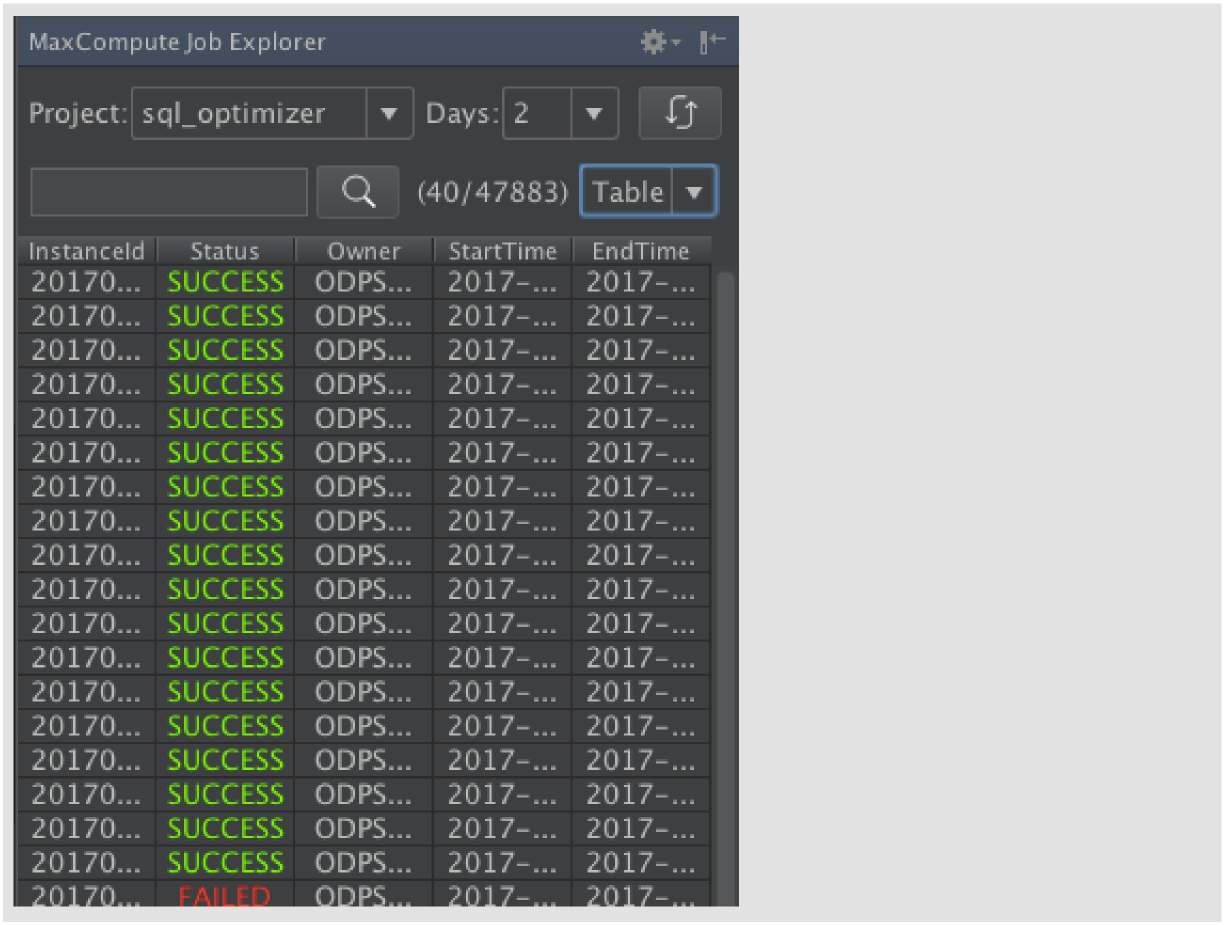
- Submit a job

  Click the icon on the toolbar of the SQL editor to submit an SQL script to the queue of the project specified by MaxCompute.

View history jobs

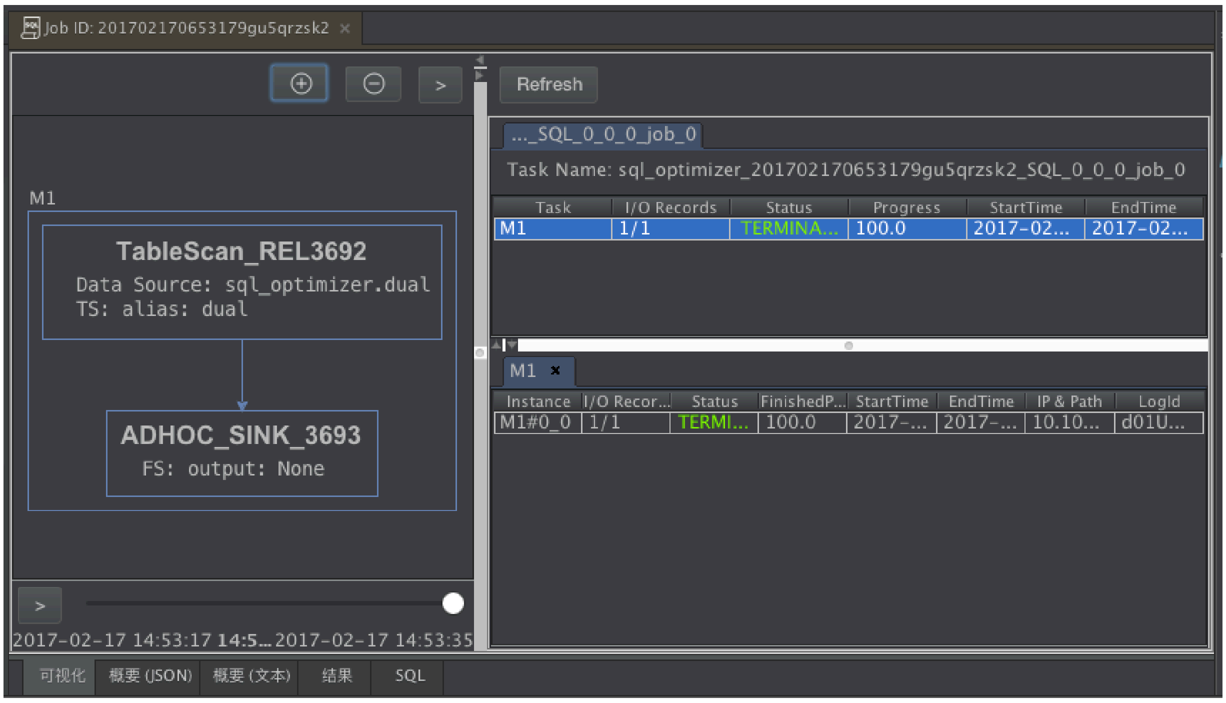
Open Job Explorer to view recently executed jobs in the specified project.

Note:
List only displays jobs submitted by the user ID of the current connection.



InstanceId	Status	Owner	StartTime	EndTime
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	SUCCESS	ODPS...	2017-...	2017-...
20170...	FAILED	ODPS...	2017-...	2017-...

Double-click a job to view the job details. as shown in the following figure.



Task Name: sql_optimizer_201702170653179gu5qrzsk2_SQL_0_0_0_job_0

Task	I/O Records	Status	Progress	StartTime	EndTime
M1	1/1	TERMINA...	100.0	2017-02...	2017-02...

Instance	I/O Recor...	Status	FinishedP...	StartTime	EndTime	IP & Path	LogId
M1#0_0	1/1	TERMI...	100.0	2017-...	2017-...	10.10...	d01U...

If you have the Log view URL of a job, you can select MaxCompute > Open Logview from the menu to go to the details page of the job.

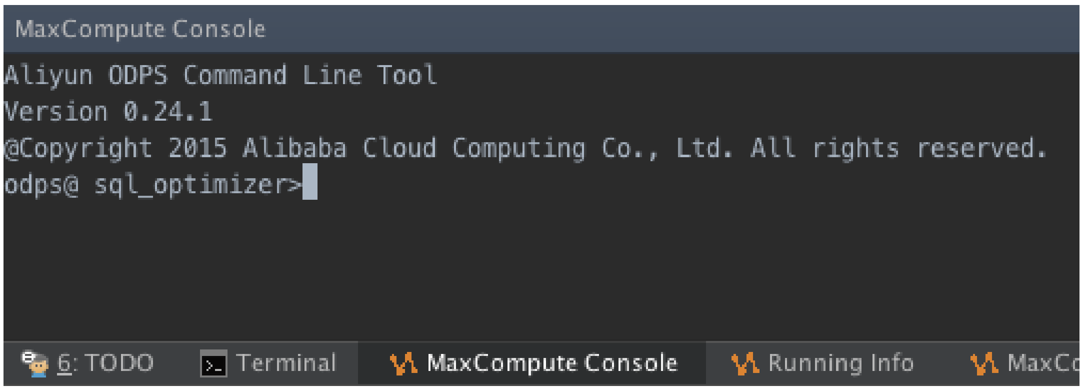
Develop a MapReduce program and UDF program

MaxCompute Studio also allows you to develop MapReduce and Java UDF programs.

Connect to a MaxCompute client

MaxCompute Studio is integrated with the MaxCompute Client of the latest version. Alternatively, you can specify the path of the locally installed MaxCompute client on the Configuration page of MaxCompute Studio.

On the Project Explorer, right-click a project and select Open in Console to open the MaxCompute Console window.



```
Aliyun ODPS Command Line Tool
Version 0.24.1
@Copyright 2015 Alibaba Cloud Computing Co., Ltd. All rights reserved.
odps@ sql_optimizer>
```

Next step

Now, you know the functional interfaces and common application scenarios of MaxCompute Studio. Continue to the next tutorial. In this tutorial, you will learn how to install MaxCompute Studio. For more information, see Install IntelliJ IDEA.

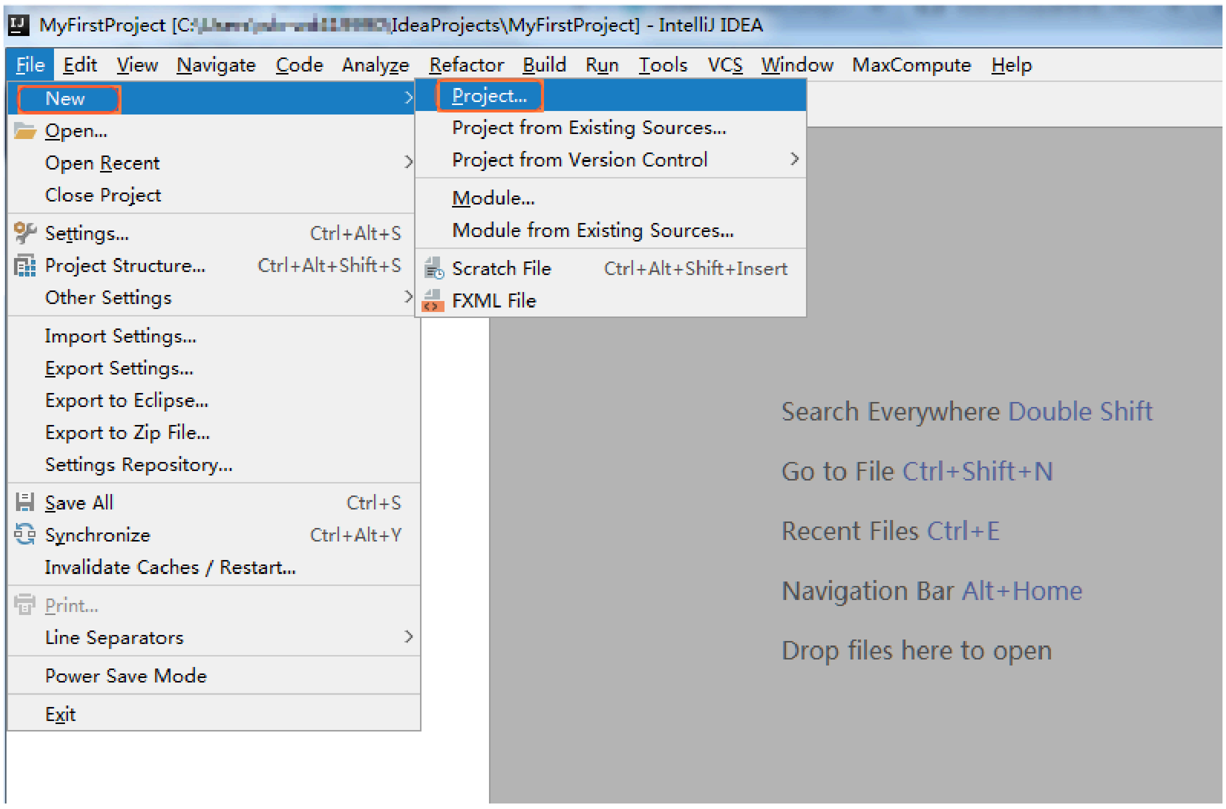
2.2 Project space connection management

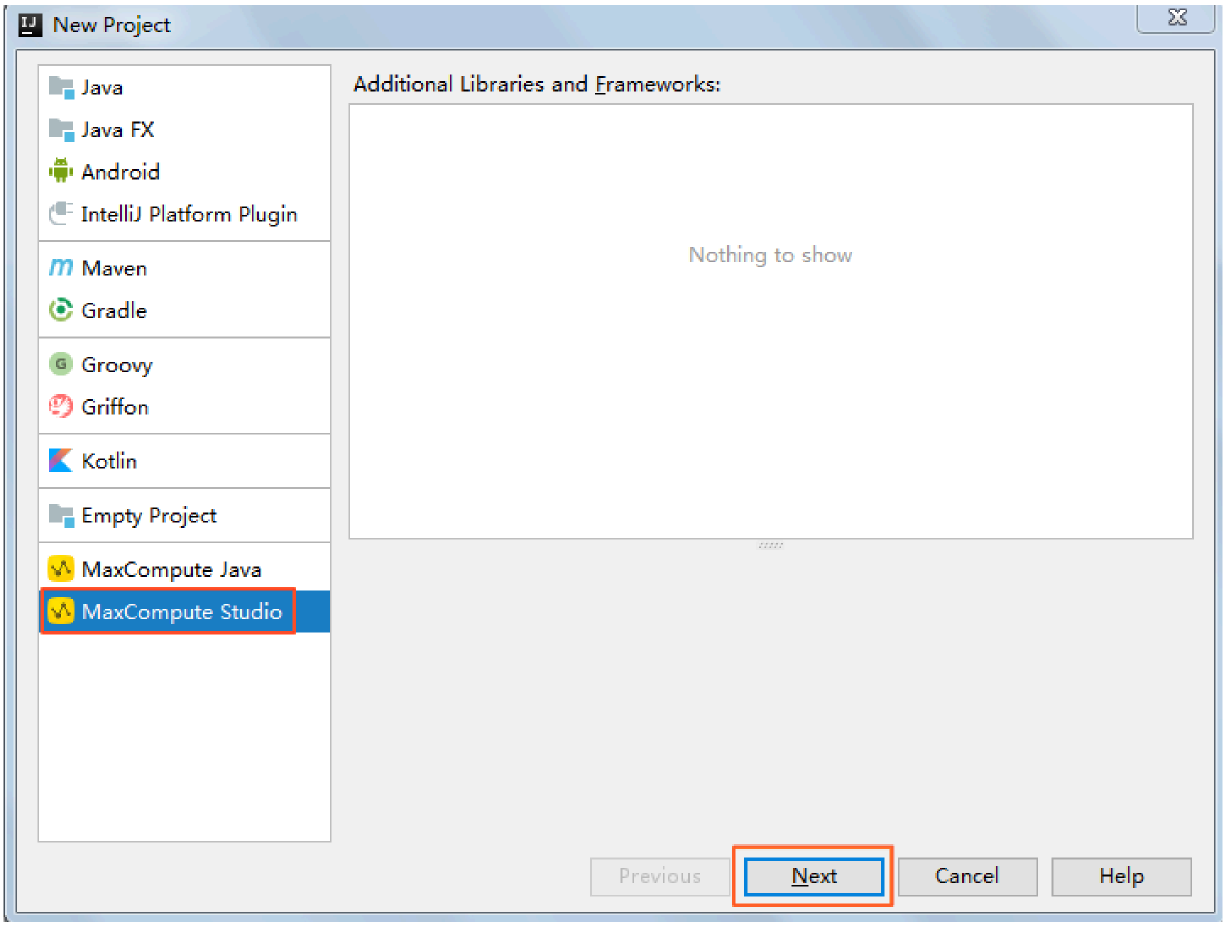
MaxCompute One of the core features of MaxCompute Studio is to browse resources of a MaxCompute project, including Table, UDF, and Resource. To realize this feature, create a project connection first.

Prerequisites

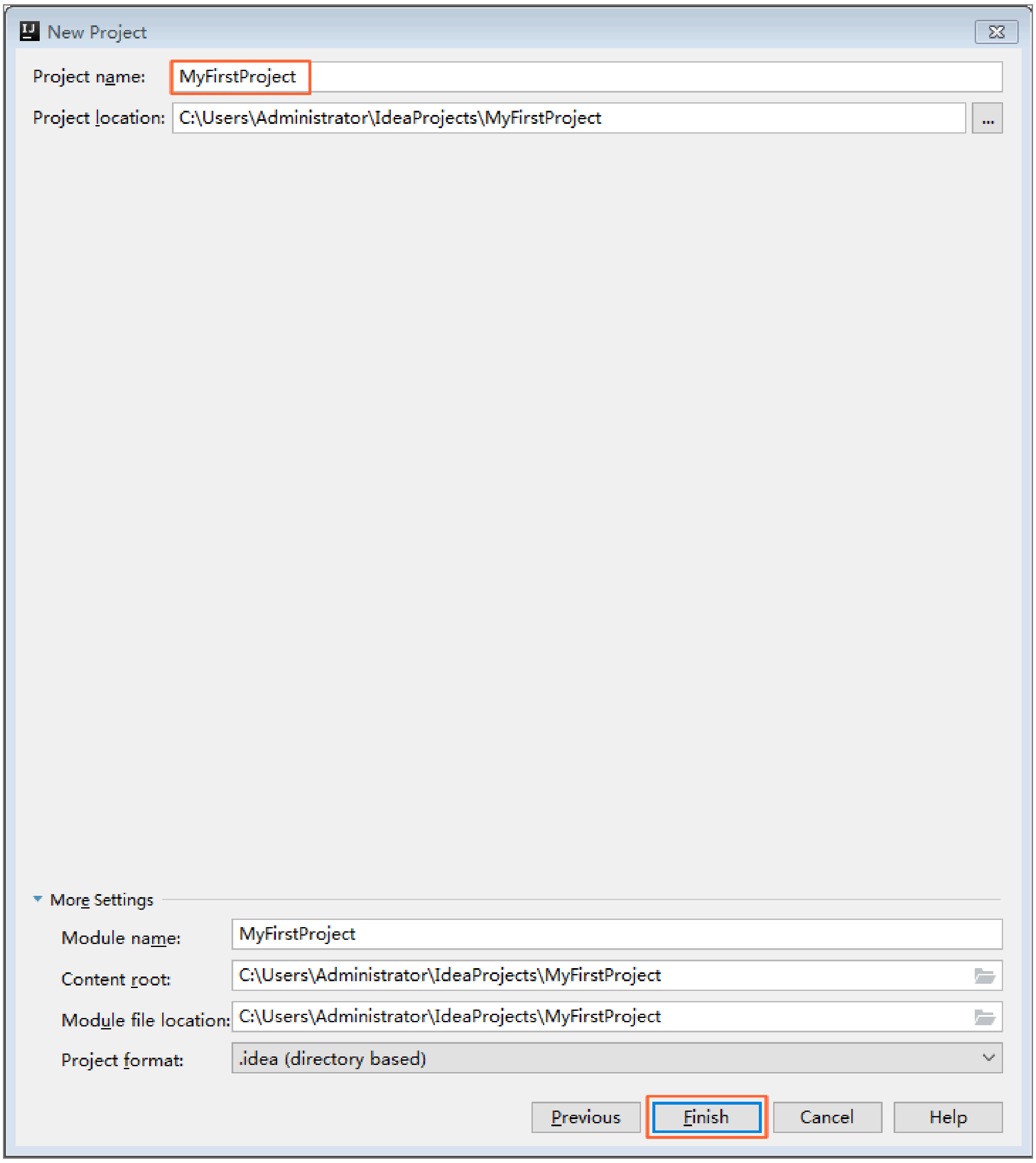
To display Tool Windows of IntelliJ IDEA, you must open an IntelliJ IDEA project, and the MaxCompute Project must be configured on IntelliJ IDEA using MaxCompute Project Explorer in Tool Windows. Therefore, before creating a MaxCompute Project connection, add or import an IntelliJ IDEA project. This document uses adding a project under Windows as an example.

Open IntelliJ IDEA, click New > Project, select MaxCompute Studio on the displayed page, and click Next.


MyFirstProject [C:\[redacted]\IdeaProjects\MyFirstProject] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute Help
New
Open...
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Export to Eclipse...
Export to Zip File...
Settings Repository...
Save All Ctrl+S
Synchronize Ctrl+Alt+Y
Invalidate Caches / Restart...
Print...
Line Separators
Power Save Mode
Exit
Project...
Project from Existing Sources...
Project from Version Control
Module...
Module from Existing Sources...
Scratch File Ctrl+Alt+Shift+Insert
FXML File
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open



Enter the project name, and click Finish.



Project name: MyFirstProject

Project location: C:\Users\Administrator\IdeaProjects\MyFirstProject

More Settings

Module name: MyFirstProject

Content root: C:\Users\Administrator\IdeaProjects\MyFirstProject

Module file location: C:\Users\Administrator\IdeaProjects\MyFirstProject

Project format: .idea (directory based)

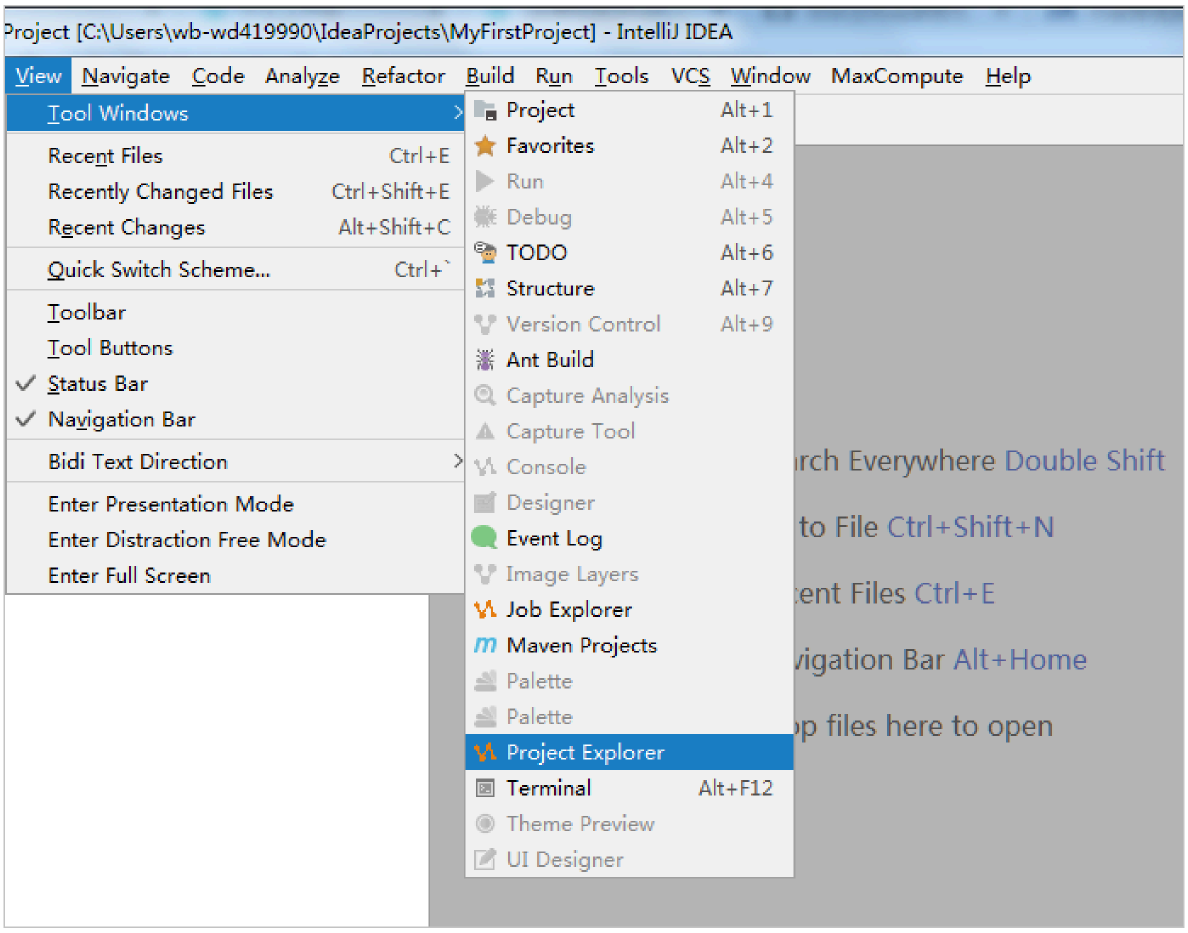
Create a MaxCompute Project Connection

We recommend that you configure the MaxCompute project connection according to your region strictly. Otherwise, some errors may occur.

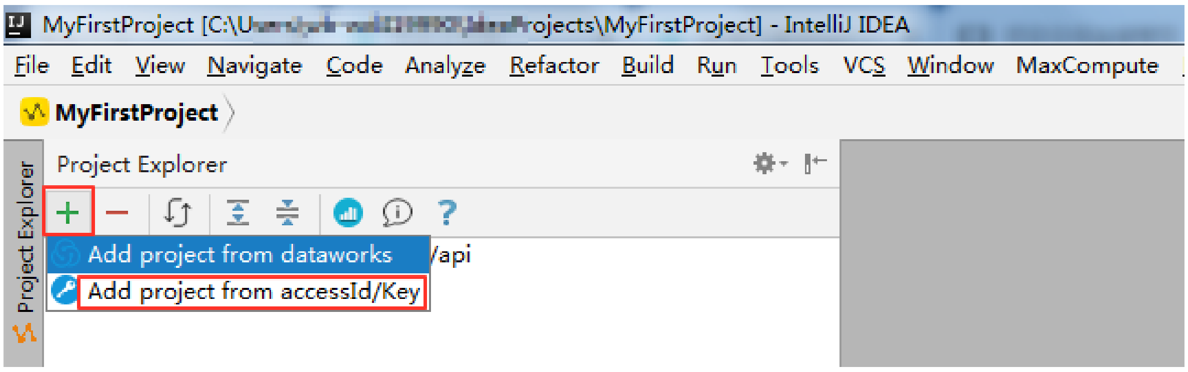
Procedure

1. Click view, select Tool Windows.

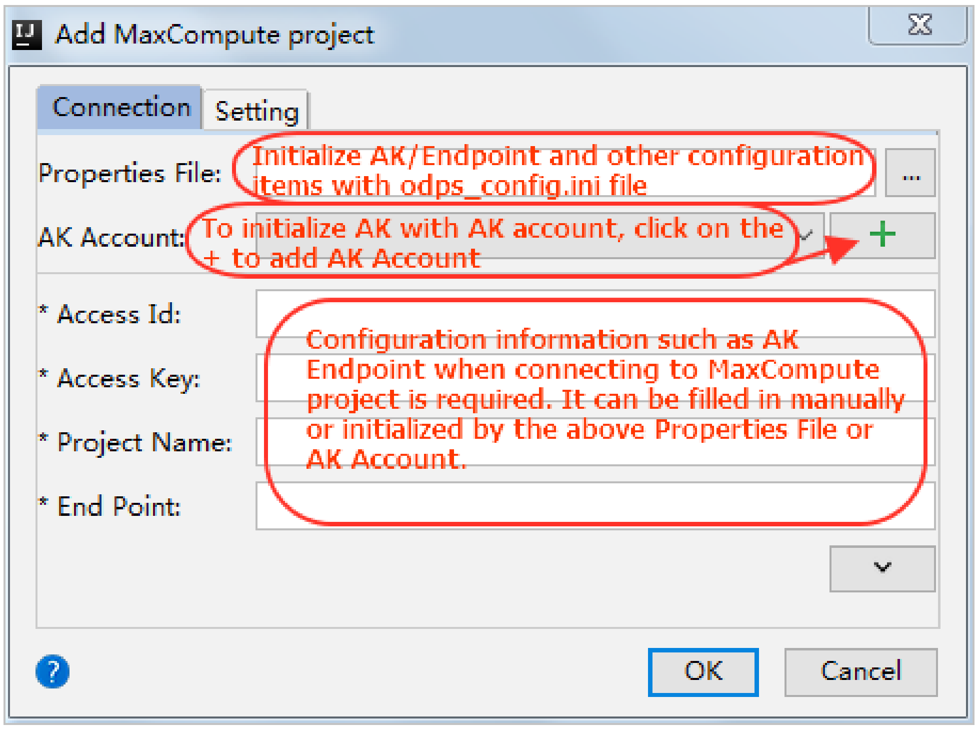
2. Click Project Explorer.



3. Click plus sign + at the upper left corner to add a MaxCompute project.


MyFirstProject [C:\U…rojects\MyFirstProject] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute
MyFirstProject
Project Explorer
Add project from dataworks
Add project from accessId/Key

4. In the Add MaxCompute Project dialog box, set configuration options.



Add MaxCompute project

Connection | Setting

Properties File: Initialize AK/Endpoint and other configuration items with odps_config.ini file

AK Account: To initialize AK with AK account, click on the + to add AK Account

* Access Id:

* Access Key:

* Project Name:

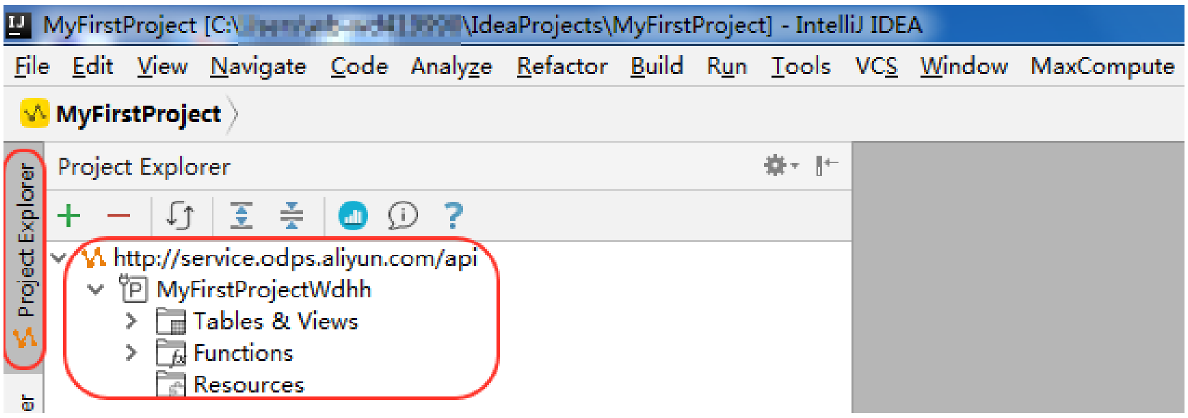
* End Point:

Configuration information such as AK Endpoint when connecting to MaxCompute project is required. It can be filled in manually or initialized by the above Properties File or AK Account.

OK | Cancel

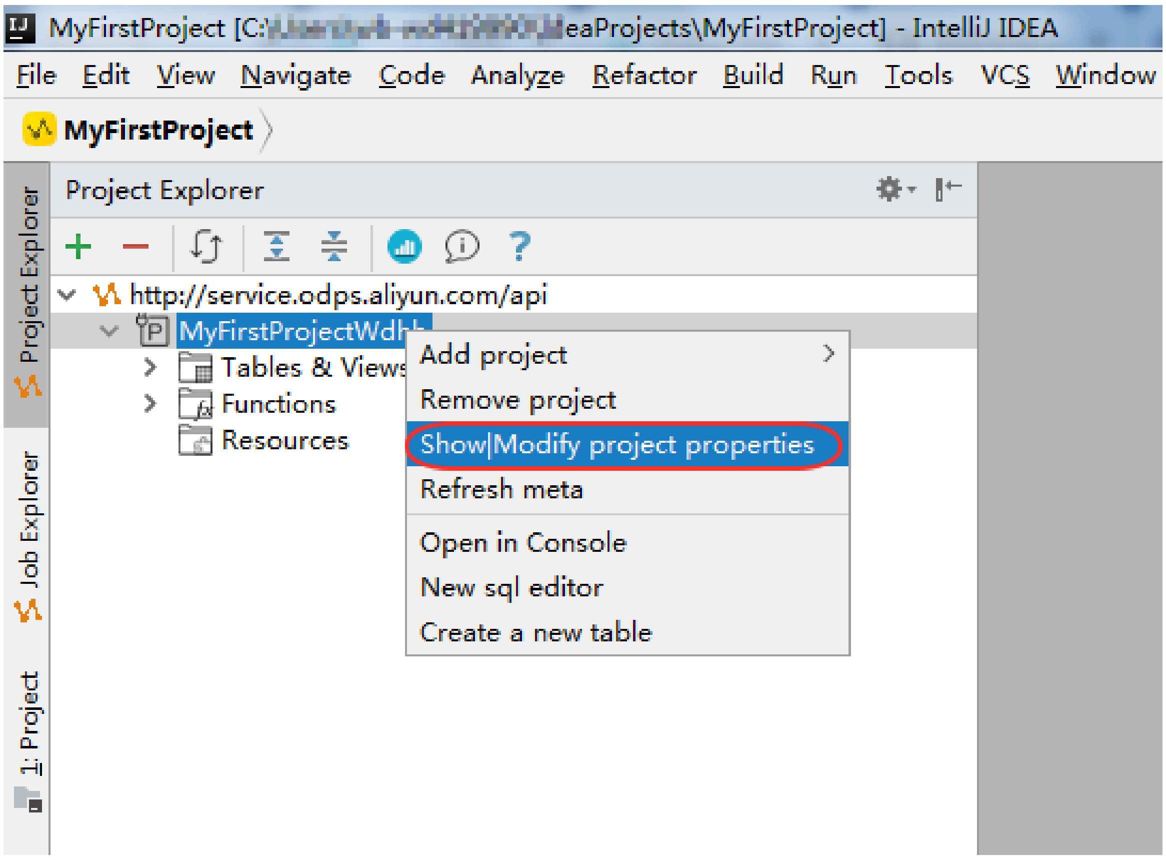
> **Note:**
>
> - Click question mark (?) at the lower left corner of the dialog box to go to the online document page.
> - If the synchronization times out, you can consider increasing the time-out duration for synchronizing metadata to the local host on the Setting tab.

5. After the preceding settings, click OK. Information about the MaxCompute project is displayed on the left of MaxCompute Project Explorer. You can click Tables, Views, Functions, and Resources to view tables, views, functions, and resources of the project.


MyFirstProject [C:\[redacted]\IdeaProjects\MyFirstProject] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute
MyFirstProject
Project Explorer
http://service.odps.aliyun.com/api
MyFirstProjectWdhh
Tables & Views
Functions
Resources
Project Explorer

View and modify a MaxCompute connection

In MaxCompute Project Explorer, right-click a MaxCompupte project and select Show|Modify project properties.


MyFirstProject [C:[redacted]eaProjects\MyFirstProject] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window
MyFirstProject
Project Explorer
Job Explorer
1: Project
Project Explorer
http://service.odps.aliyun.com/api
MyFirstProjectWdh
Tables & Views
Functions
Resources
Add project
Remove project
Show|Modify project properties
Refresh meta
Open in Console
New sql editor
Create a new table

In the displayed dialog box, you can view or modify connections and settings of the MaxCompute project.

Subsequent operations

Now, you know how to create and manage a project connection. You can continue to the next tutorial. In the tutorial, you will learn how to query metadata, clear data, and upload and download data to manage data and resources. For more information, see *Manage data and resources*.

2.3 Tools Installation and version history

2.3.1 Install IntelliJ IDEA

This document describes how to install the basic platform IntelliJ IDEA of MaxCompute Studio.

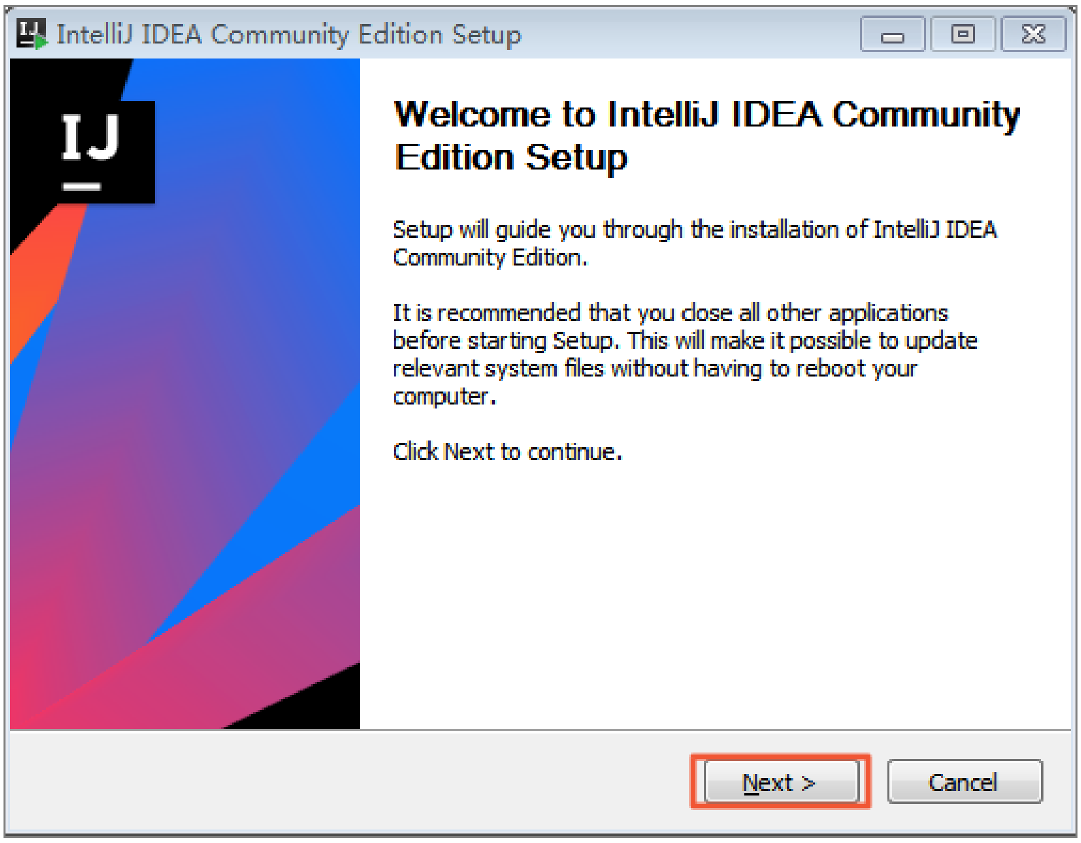
Context

Procedure

Procedure

1. Click *here* to download the IntelliJ IDEA of the version corresponding to your operating system (Windows, macOS, or Linux). The following assumes that the Windows operating system is used.

   Download IntelliJ IDEA 14.1.4 or a later version. (The Ultimate version, PyCharm version, and free Community version are supported.)
2. After the download is complete, double-click the installation program to enter the installation page, and click Next, as shown in the following figure.



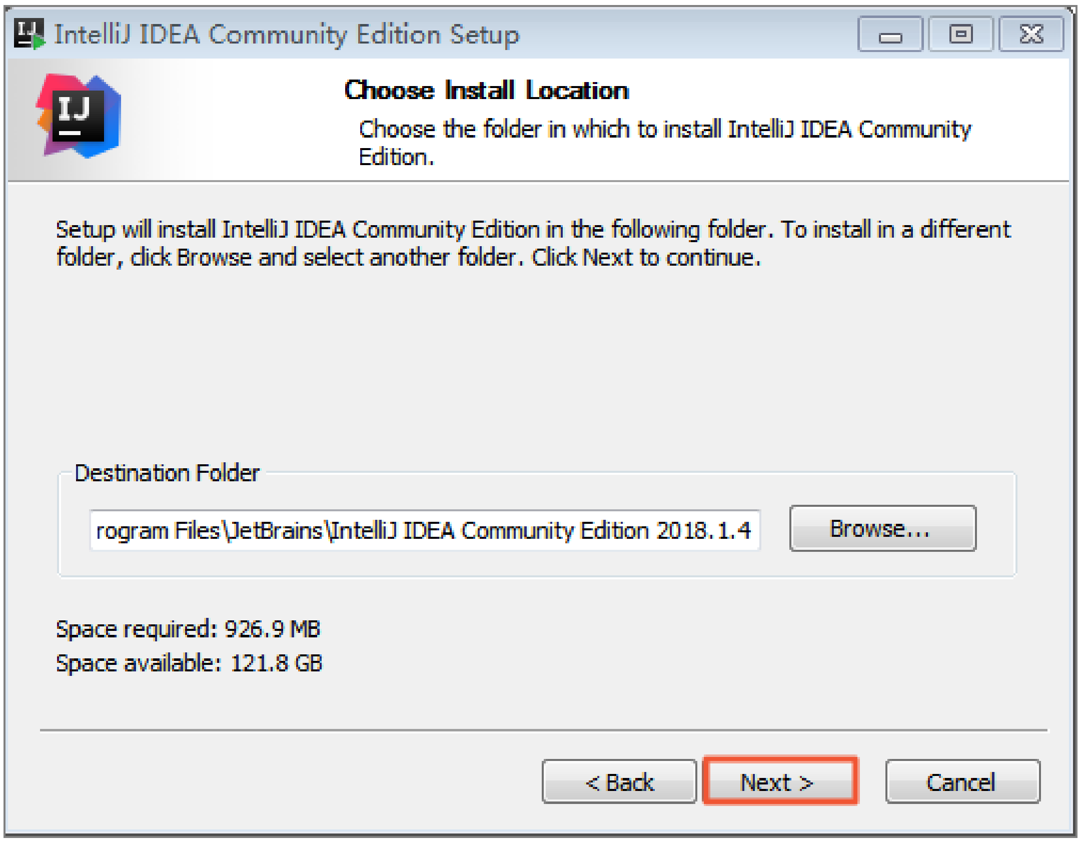
Welcome to IntelliJ IDEA Community Edition Setup

Setup will guide you through the installation of IntelliJ IDEA Community Edition.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

3. Specify the installation directory, and click Next, as shown in the following figure.



Choose the folder in which to install IntelliJ IDEA Community Edition.

Setup will install IntelliJ IDEA Community Edition in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

rogram Files\JetBrains\IntelliJ IDEA Community Edition 2018.1.4

Space required: 926.9 MB
Space available: 121.8 GB

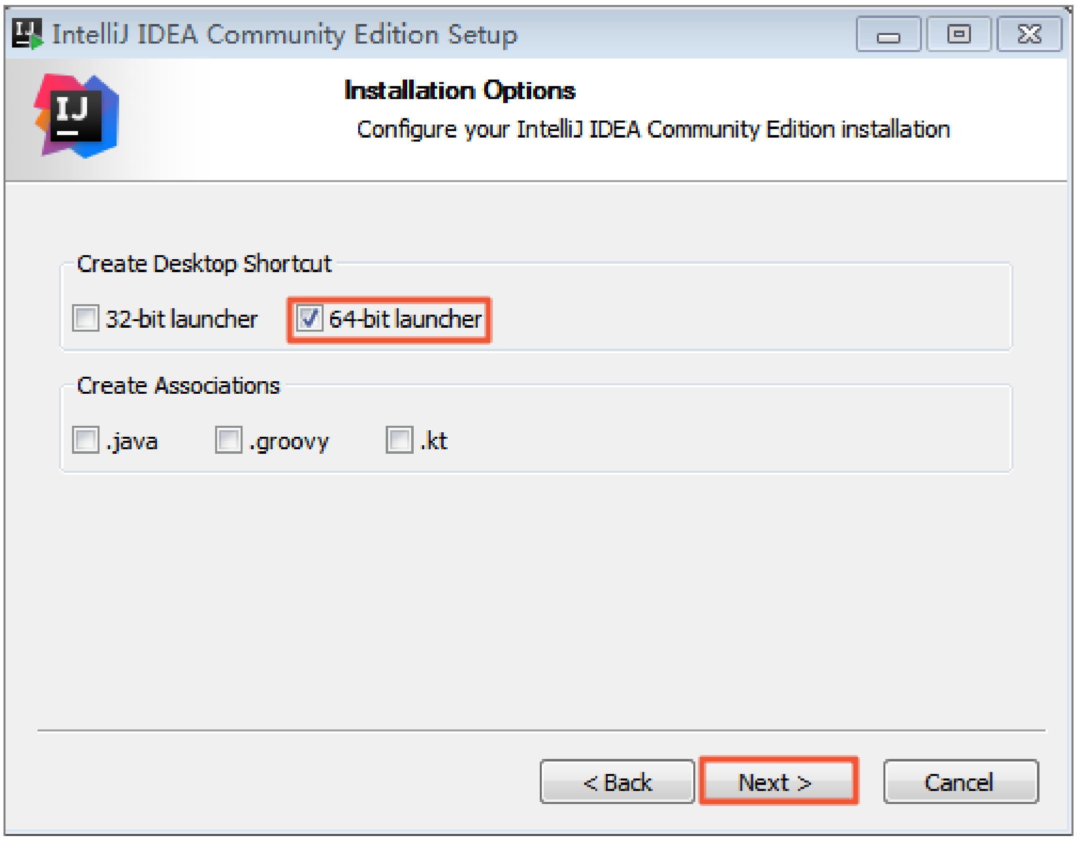
4. Select the 32-bit or 64-bit IntelliJ IDEA based on the version of the local operating system.

   You can query the local operating system version by following these steps:

   a) Open Windows Resource Manager, right-click Computer and select Properties from the shortcut menu. as shown in the following figure.

   b) In the displayed window, check the type of the operating system.

5. Select the corresponding system type and click Next , as shown in the following figure.


IntelliJ IDEA Community Edition Setup

Installation Options

Configure your IntelliJ IDEA Community Edition installation

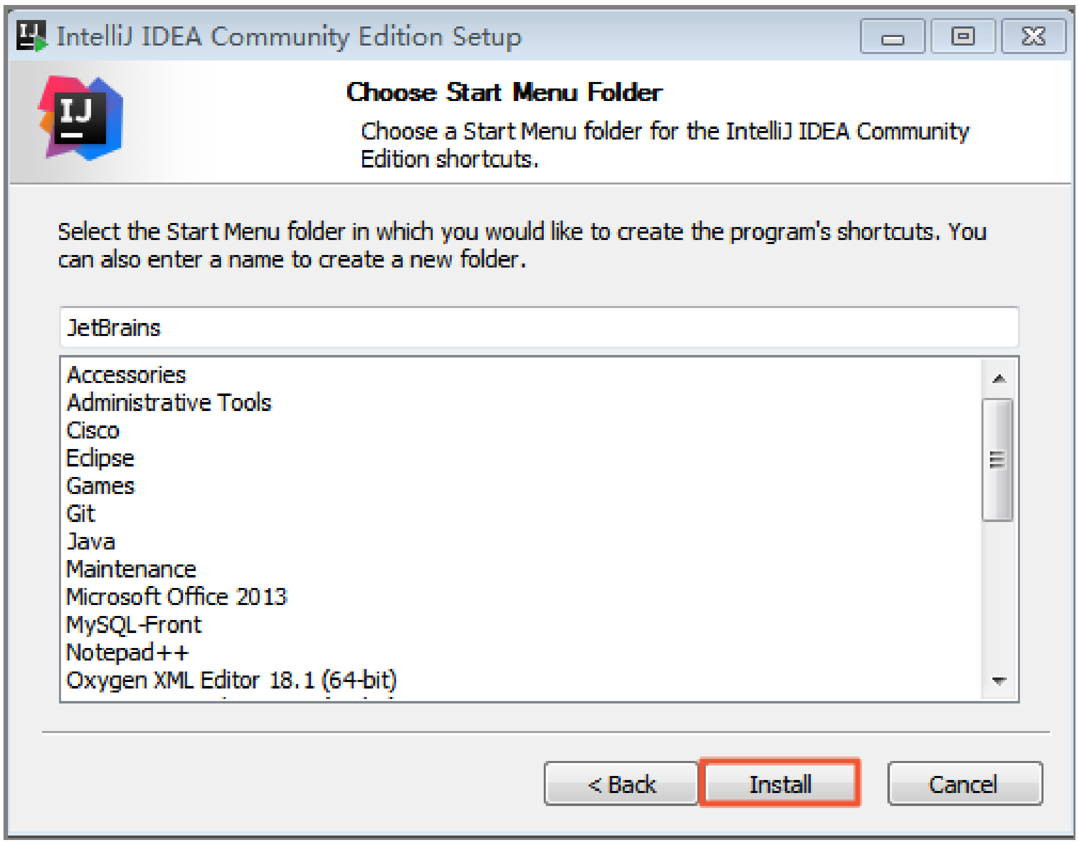
Create Desktop Shortcut

- [ ] 32-bit launcher
- [✓] 64-bit launcher

Create Associations

- [ ] .java
- [ ] .groovy
- [ ] .kt

6. Click Install to start installation, as shown in the following figure.



IntelliJ IDEA Community Edition Setup

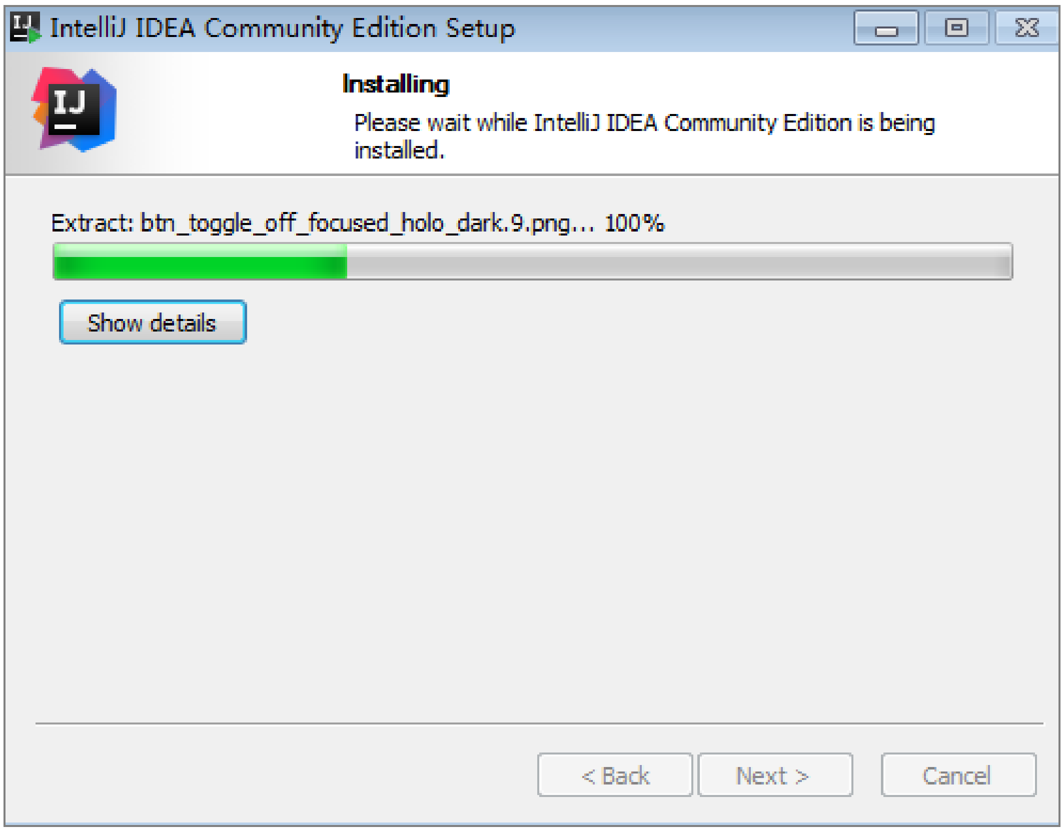
Choose Start Menu Folder

Choose a Start Menu folder for the IntelliJ IDEA Community Edition shortcuts.

Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

JetBrains

Accessories
Administrative Tools
Cisco
Eclipse
Games
Git
Java
Maintenance
Microsoft Office 2013
MySQL-Front
Notepad++
Oxygen XML Editor 18.1 (64-bit)



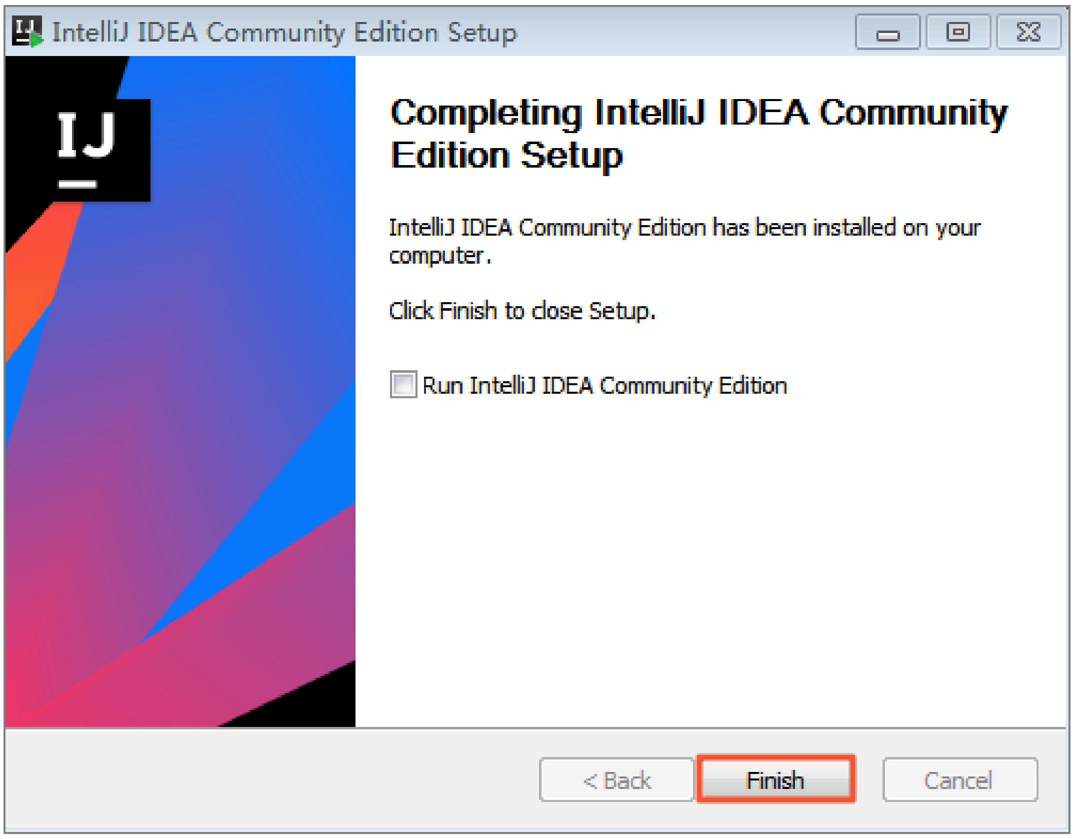
IntelliJ IDEA Community Edition Setup

Installing

Please wait while IntelliJ IDEA Community Edition is being installed.

Extract: btn_toggle_off_focused_holo_dark.9.png... 100%

7. After the installation is complete, click Finish.


IntelliJ IDEA Community Edition Setup

Completing IntelliJ IDEA Community Edition Setup

IntelliJ IDEA Community Edition has been installed on your computer.

Click Finish to close Setup.

☐ Run IntelliJ IDEA Community Edition

What's next

Now you know how to install IntelliJ IDEA. Continue to the next tutorial. In this tutorial, you will learn how to install the MaxCompute Studio plugin. For more information, see *Install the MaxCompute Studio plugin*.

2.3.2 Installation procedure

Environment requirements

IntelliJ IDEA can be installed on *Windows*, *Mac*, *Linux*. For more information about the hardware and system environment requirements, click *here* . IntelliJ IDEA-based MaxCompute Studio can also be installed on clients running these operating systems.

MaxCompute Studio has the following requirements on the your environment:

- A client running Windows, macOS, or Linux.
- IntelliJ IDEA 14.1.4 or a later version is installed. (The Ultimate version, PyCharm version, and free *Community version* are supported.)
- JRE 1.8 is installed. (JRE 1.8 has been bound to the latest IntelliJ IDEA.)

- JDK 1.8 is installed. (*Optional*: JDK is required if you need to develop and debug Java UDF.)

> **Note:**
> The client supports JDK 1.9 from the 0.28.0 version. The previous version only supports JDK 1.8.

Installation method

MaxCompute Studio is a plugin of IntelliJ IDEA, which can be installed using either of the following two methods:

- Online installation using the plugin library (recommended)
- Installation using a local file

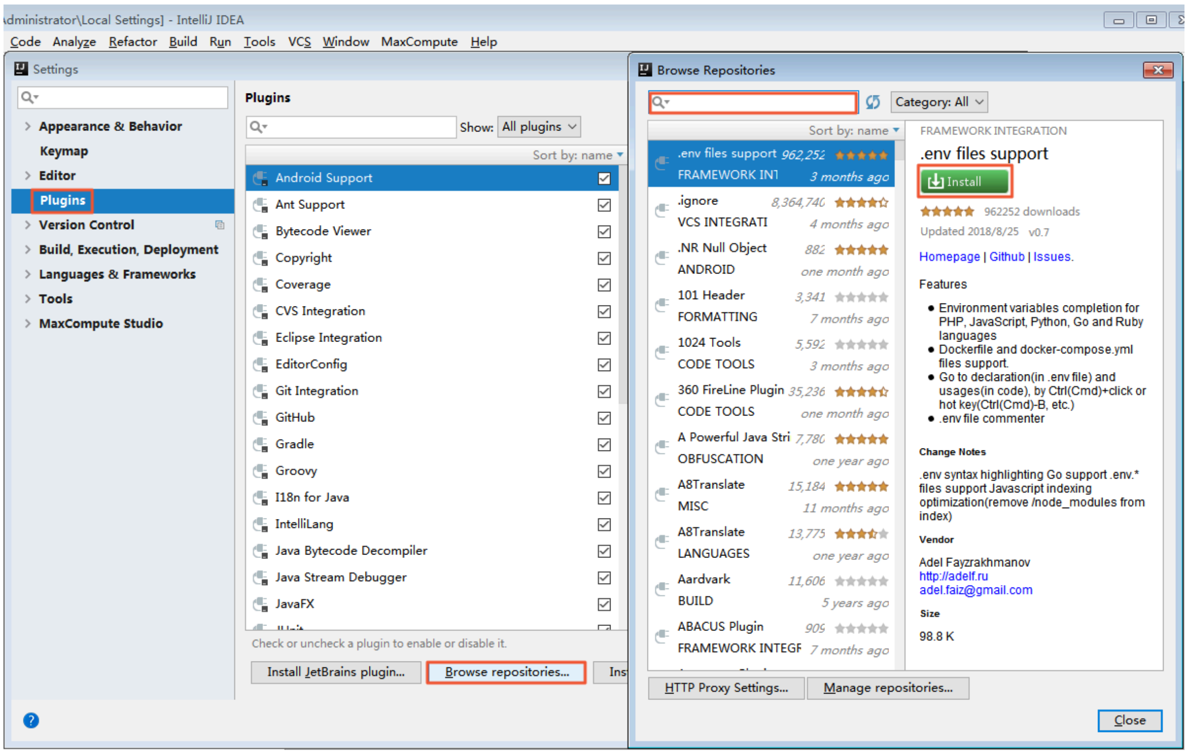
Online installation (recommended)

MaxCompute Studio The MaxCompute Studio plugin has been opened for all users on the Internet. You can install MaxCompute Studio using the official IntelliJ IDEA plugin library.

Procedure

1. Open the plugin configuration page on IntelliJ IDEA. (If you are a Windows/Linux user, choose File > Settings > > Plugins. If you are a macOS user, choose IntelliJ IDEA > Preferences > Plugins).
2. Click Browse repositories… and search for `MaxCompute Studio`.
3. On the MaxCompute Studio plugin page, click Install.

4. After the installation is confirmed, restart IntelliJ IDEA to complete installation.



Local installation

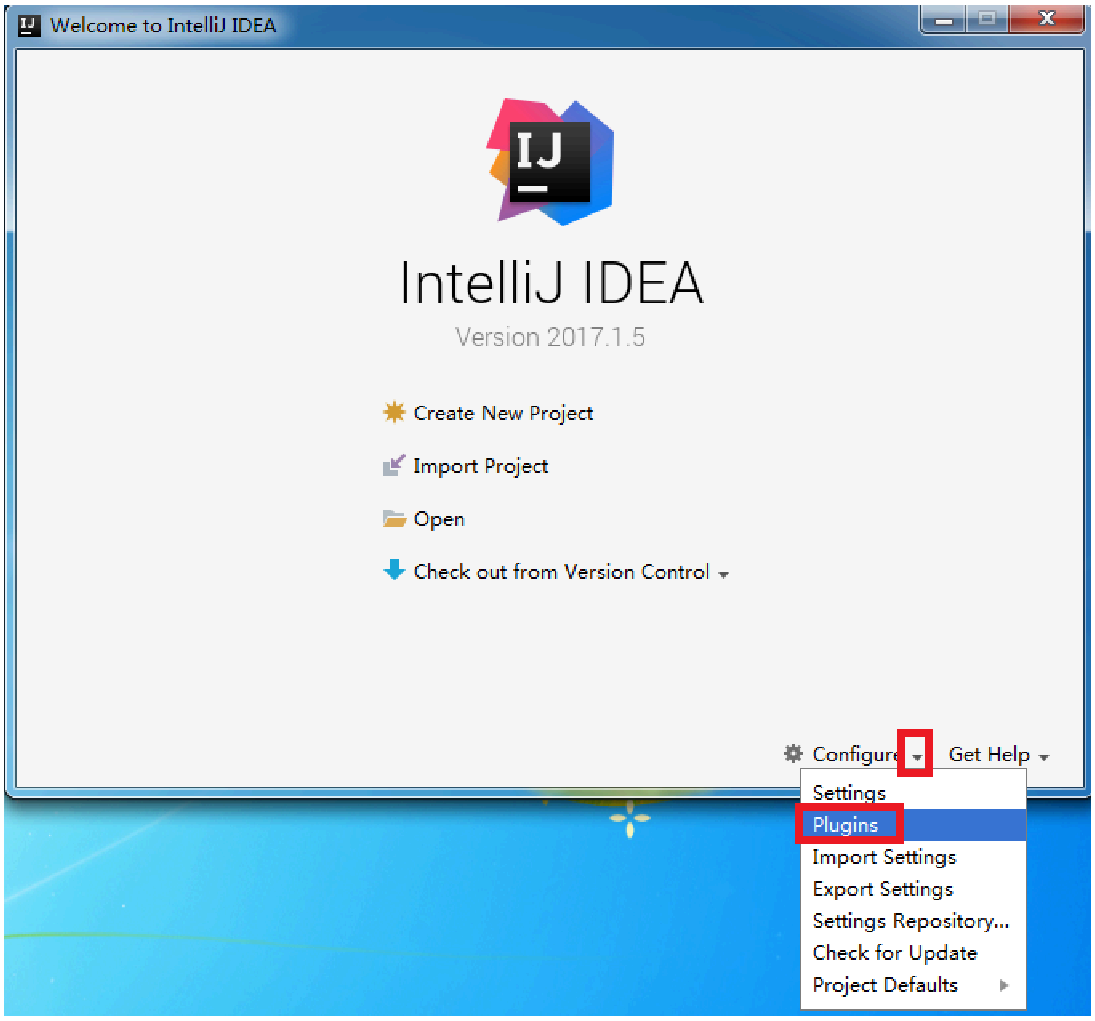
MaxCompute Studio MaxCompute Studio can also be installed in a local environment.

Procedure

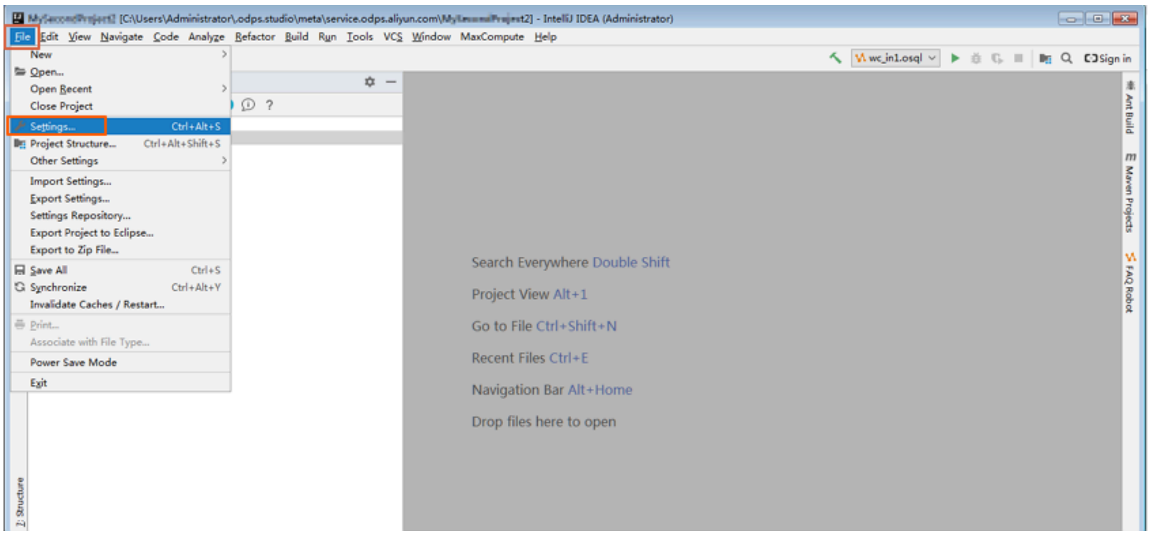
1. Go to the *MaxCompute Studio plugin page* to download the plugin package.

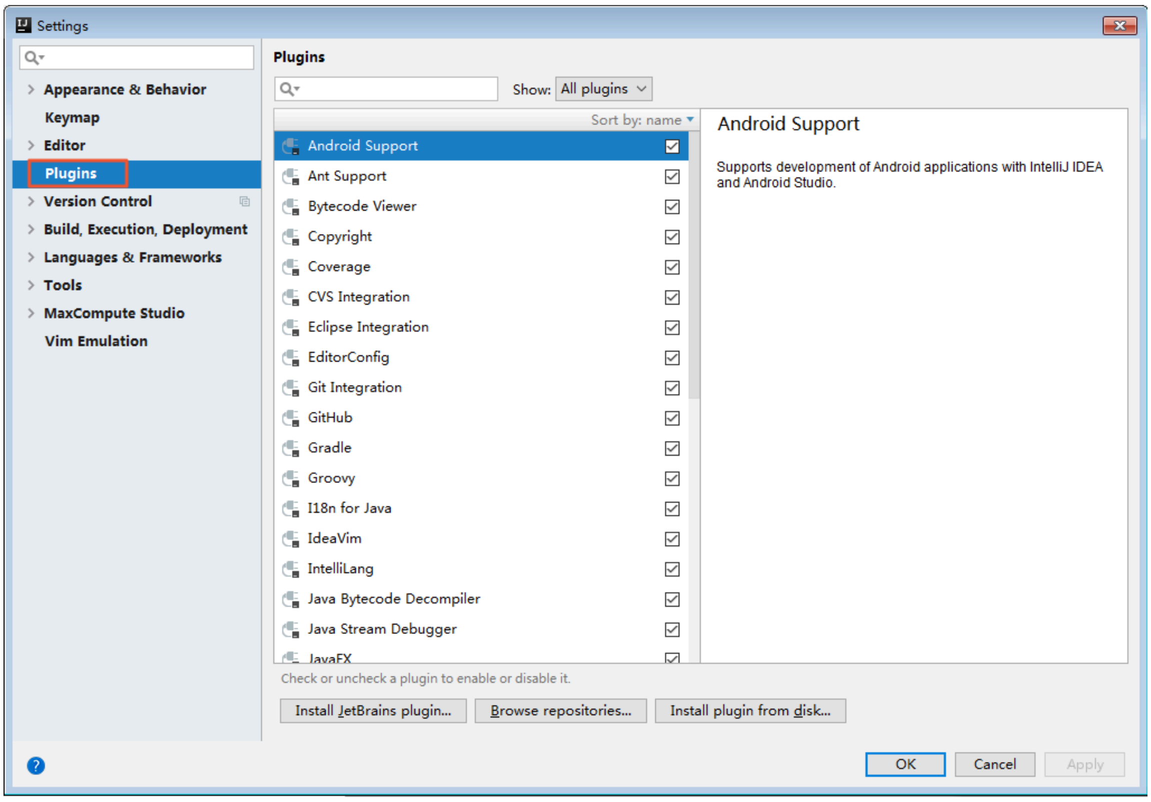
2. Run IntelliJ IDEA.

    - If you access IntelliJ IDEA for the first time, a welcome page is displayed. Click Configure and select Plugins from the shortcut menu, as shown in the following figure.

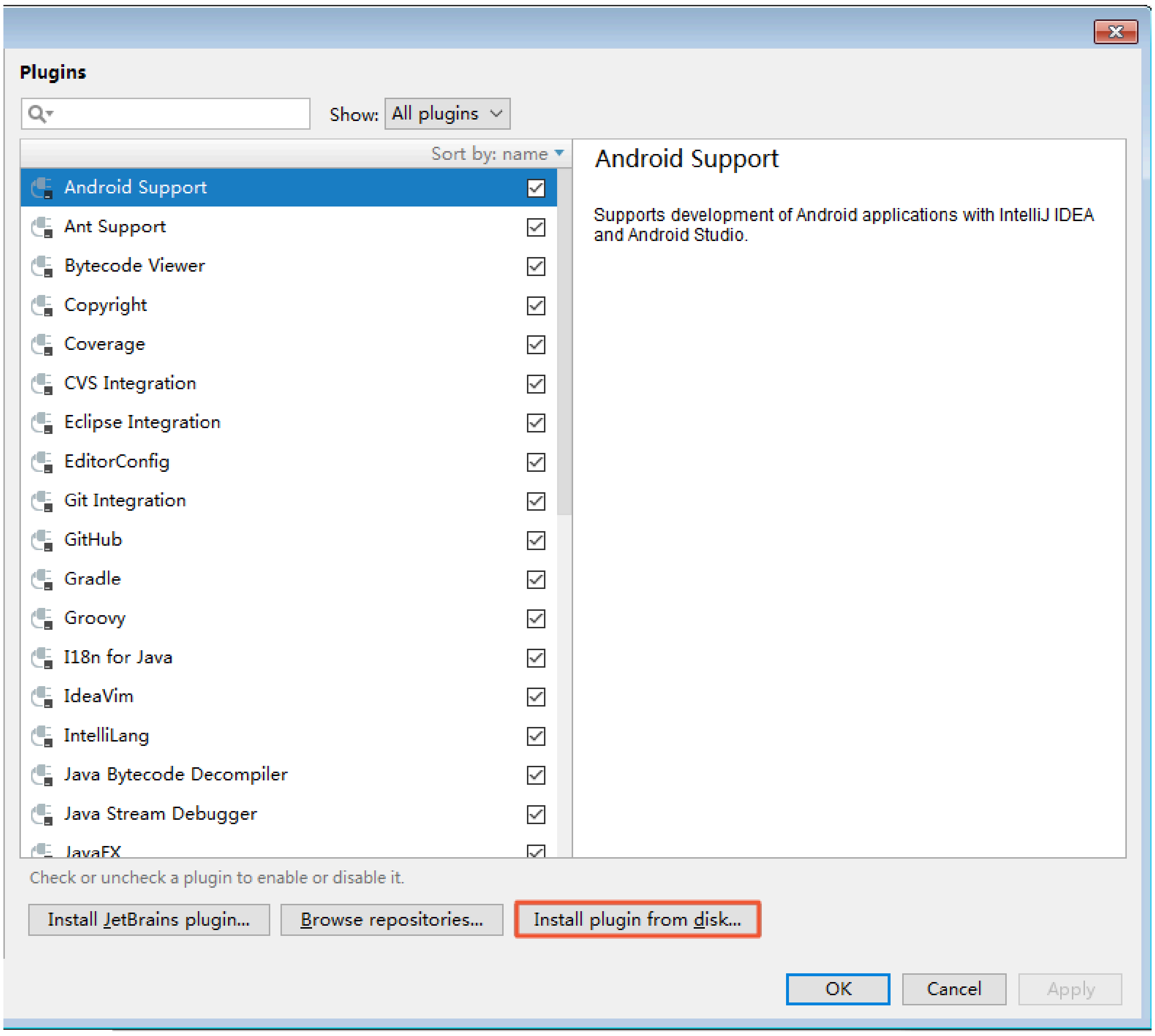


    - If you have accessed IntelliJ IDEA before, choose File > Settings > Plugins to enter the same page, as shown in the following figure.


[redacted] [C:\Users\Administrator\.odps.studio\meta\service.odps.aliyun.com\[redacted]] - IntelliJ IDEA (Administrator)
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute Help
New
Open...
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Settings Repository...
Export Project to Eclipse...
Export to Zip File...
Save All Ctrl+S
Synchronize Ctrl+Alt+Y
Invalidate Caches / Restart...
Print...
Associate with File Type...
Power Save Mode
Exit
wc_in1.osql
Sign in
Search Everywhere Double Shift
Project View Alt+1
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open
Ant Build
Maven Projects
FAQ Robot
Structure


Settings
Appearance & Behavior
Keymap
Editor
Plugins
Version Control
Build, Execution, Deployment
Languages & Frameworks
Tools
MaxCompute Studio
Vim Emulation
Plugins
Show: All plugins
Sort by: name
Android Support
Ant Support
Bytecode Viewer
Copyright
Coverage
CVS Integration
Eclipse Integration
EditorConfig
Git Integration
GitHub
Gradle
Groovy
I18n for Java
IdeaVim
IntelliLang
Java Bytecode Decompiler
Java Stream Debugger
JavaFX
Android Support
Supports development of Android applications with IntelliJ IDEA and Android Studio.
Check or uncheck a plugin to enable or disable it.
Install JetBrains plugin...
Browse repositories...
Install plugin from disk...
OK
Cancel
Apply

3. On the Plugins page, click Install plugin from disk…, as shown in the following figure.


Plugins

Show: All plugins

Sort by: name

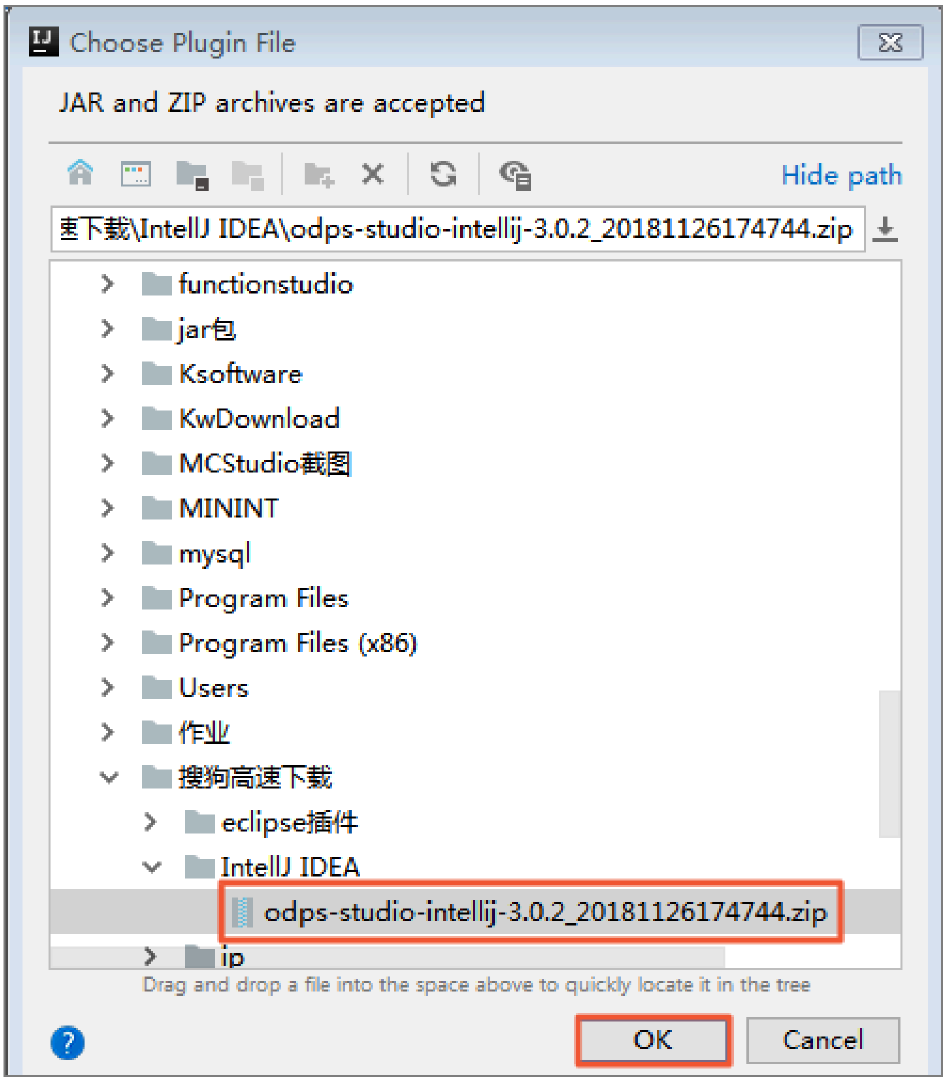
Android Support ☑
Ant Support ☑
Bytecode Viewer ☑
Copyright ☑
Coverage ☑
CVS Integration ☑
Eclipse Integration ☑
EditorConfig ☑
Git Integration ☑
GitHub ☑
Gradle ☑
Groovy ☑
I18n for Java ☑
IdeaVim ☑
IntelliLang ☑
Java Bytecode Decompiler ☑
Java Stream Debugger ☑
JavaFX ☑

Android Support

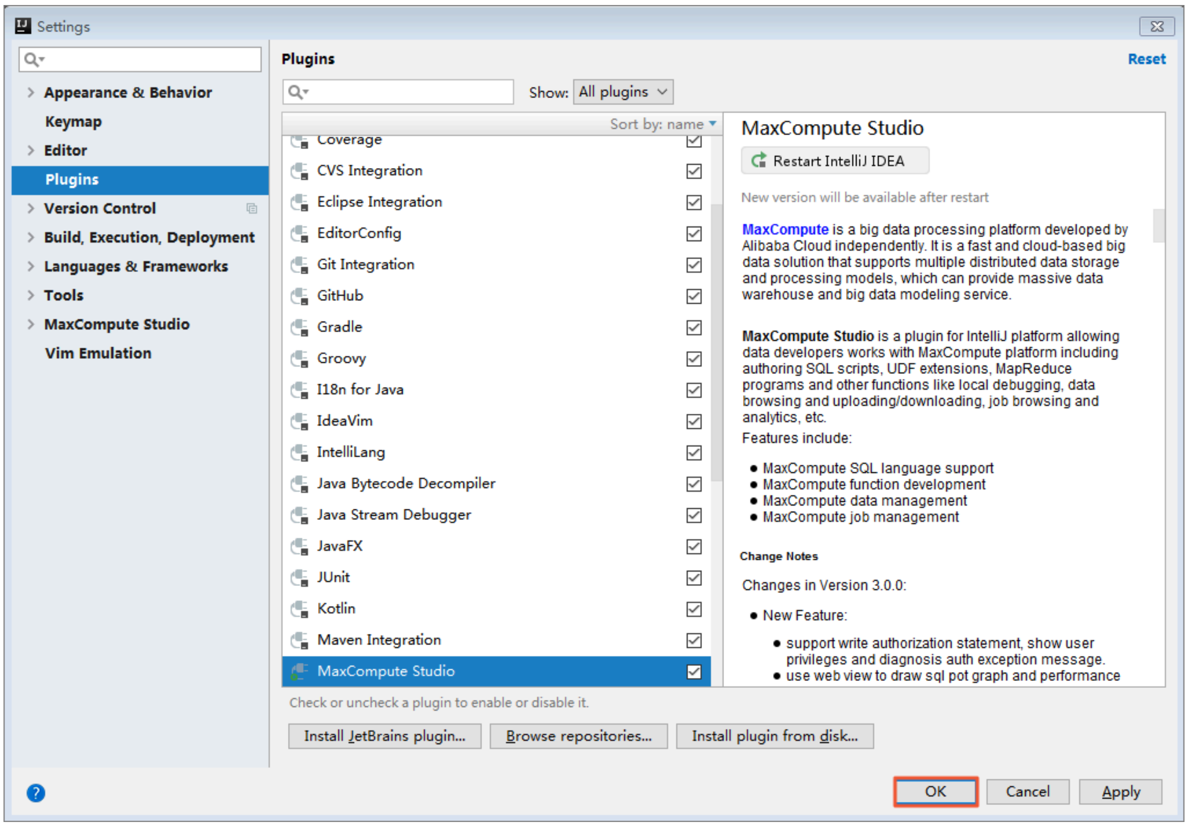
Supports development of Android applications with IntelliJ IDEA and Android Studio.

Check or uncheck a plugin to enable or disable it.

4. In the displayed window, click the gray icon before a directory for navigation, find the plugin file, select it, and click OK.


Choose Plugin File
JAR and ZIP archives are accepted
Hide path
速下载\IntelliJ IDEA\odps-studio-intellij-3.0.2_20181126174744.zip
functionstudio
jar包
Ksoftware
KwDownload
MCStudio截图
MININT
mysql
Program Files
Program Files (x86)
Users
作业
搜狗高速下载
eclipse插件
IntelliJ IDEA
odps-studio-intellij-3.0.2_20181126174744.zip
ip
Drag and drop a file into the space above to quickly locate it in the tree
OK
Cancel

5. Return to the Plugins page and click OK to install the local plugin.



MaxCompute Studio

New version will be available after restart

MaxCompute is a big data processing platform developed by Alibaba Cloud independently. It is a fast and cloud-based big data solution that supports multiple distributed data storage and processing models, which can provide massive data warehouse and big data modeling service.

MaxCompute Studio is a plugin for IntelliJ platform allowing data developers works with MaxCompute platform including authoring SQL scripts, UDF extensions, MapReduce programs and other functions like local debugging, data browsing and uploading/downloading, job browsing and analytics, etc.

Features include:

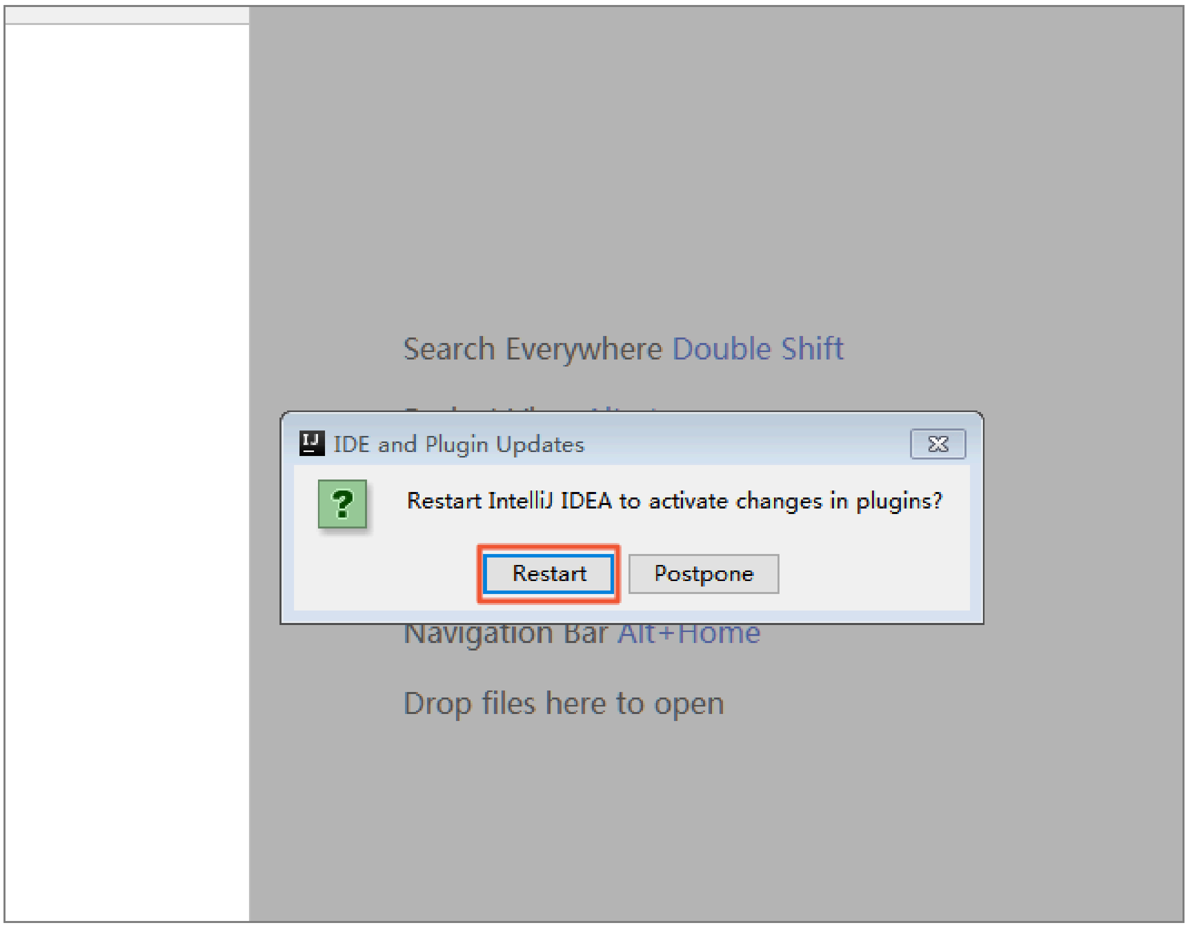
- MaxCompute SQL language support
- MaxCompute function development
- MaxCompute data management
- MaxCompute job management

Change Notes

Changes in Version 3.0.0:

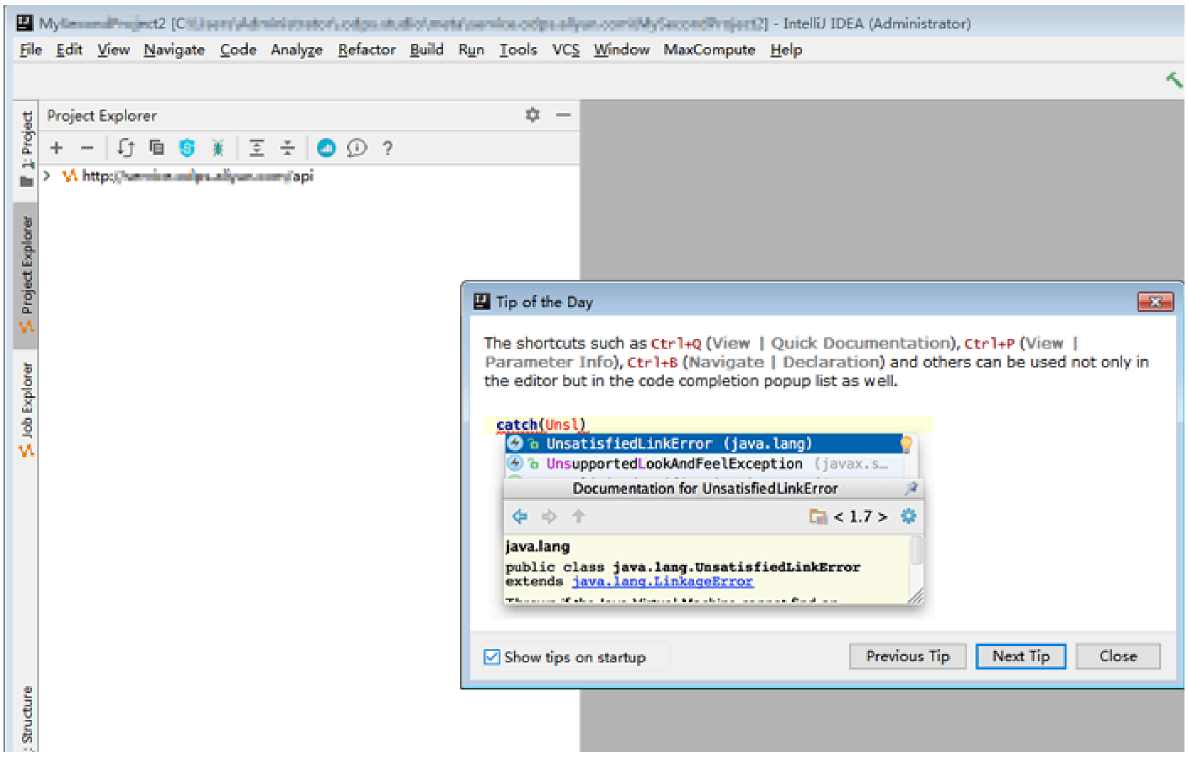
- New Feature:
  - support write authorization statement, show user privileges and diagnosis auth exception message.
  - use web view to draw sql pot graph and performance

6. After the installation is complete, a dialog box is displayed, prompting you to restart IntelliJ IDEA. Click Restart.



Restart IntelliJ IDEA to activate changes in plugins?

7. After IntelliJ IDEA is restarted, the page is displayed as shown in the following figure.


My[redacted]ct2 [C:\[redacted]t2] - IntelliJ IDEA (Administrator)
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute Help
Project Explorer
http://[redacted]/api
1: Project
Project Explorer
Job Explorer
Structure
Tip of the Day
Show tips on startup
Previous Tip
Next Tip
Close

The shortcuts such as Ctrl+Q (View | Quick Documentation), Ctrl+P (View | Parameter Info), Ctrl+B (Navigate | Declaration) and others can be used not only in the editor but in the code completion popup list as well.

```
catch(Unsl)
UnsatisfiedLinkError (java.lang)
UnsupportedLookAndFeelException (javax.s…
Documentation for UnsatisfiedLinkError
< 1.7 >
java.lang
public class java.lang.UnsatisfiedLinkError
extends java.lang.LinkageError
```

Next step

Now, you know how to install the MaxCompute Studio plugin. Continue to the next tutorial. In the tutorial, you will learn how to configure a MaxCompute project connection to manage data and resources. For more information, see *Create a MaxCompute project connection*.

2.3.3 View and upgrade the version

View the MaxCompute Studio version

Perform the following steps to view the Studio version:

1. Go to the Settings/Preferences page (by pressing `Ctrl - Alt - S` in Windows or ⌘ in macOS).
2. Select Plugins on the left bar of the dialog box and search for *MaxCompute Studio*.
3. View the MaxCompute Studio version number and release information.

Alternatively, you can select MaxCompute Studio on the left bar of the Settings page to view the current version number.

Check new versions

By default, MaxCompute Studio automatically detects new versions. If a new version is available, MaxCompute Studio automatically notifies you.

After receiving an update prompt, you can select:

- Install: Click the Install link in the update prompt. The new version is automatically downloaded and installed. After the installation is complete, restart IntelliJ IDEA.
- Configure: Click the Configure link in the update prompt. You can configure whether to detect new versions automatically.

If the automatic update function is disabled, you can perform the following steps to check and install a new version of MaxCompute Studio.

1. Go to the Settings/Preferences page (by pressing `Ctrl - Alt - S` in Windows or ⌘ in macOS).
2. Select MaxCompute Studio on the left bar of the dialog box.
3. On the MaxCompute Studio configuration page, click Check new versions.
4. If a new available version is detected, Studio notifies you of the new version number. Click Install new version and restart IntelliJ IDEA to complete installation.

You can use the Automatically checks for new version check box to control the switch for automatic version update check.

Next step

Create a MaxCompute project connection

2.4 Manage data and resources

2.4.1 View tables and UDF

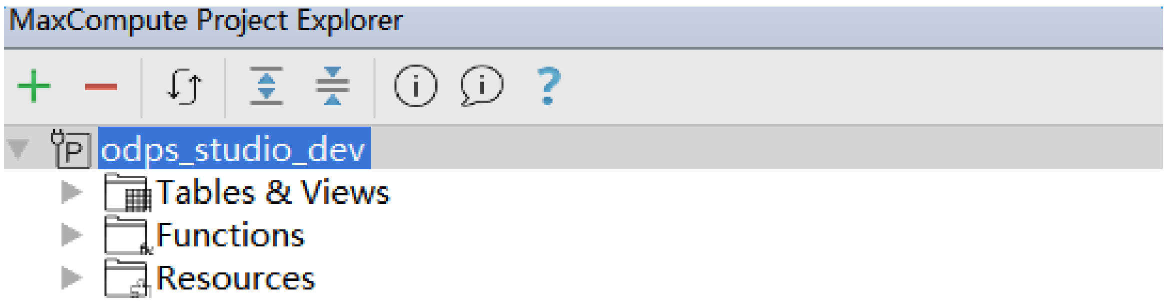
View tables and functionsView tables and functions

In the Project Explorer window, you can view tables, functions, and resources with connections added. For tables and functions to be viewed in the Project Explorer window, the MaxCompute project connections must be added, for more information, see *Add MaxCompute project connections*.

Browse tables and functions

To browse tables and functions in the project space, follow these steps.

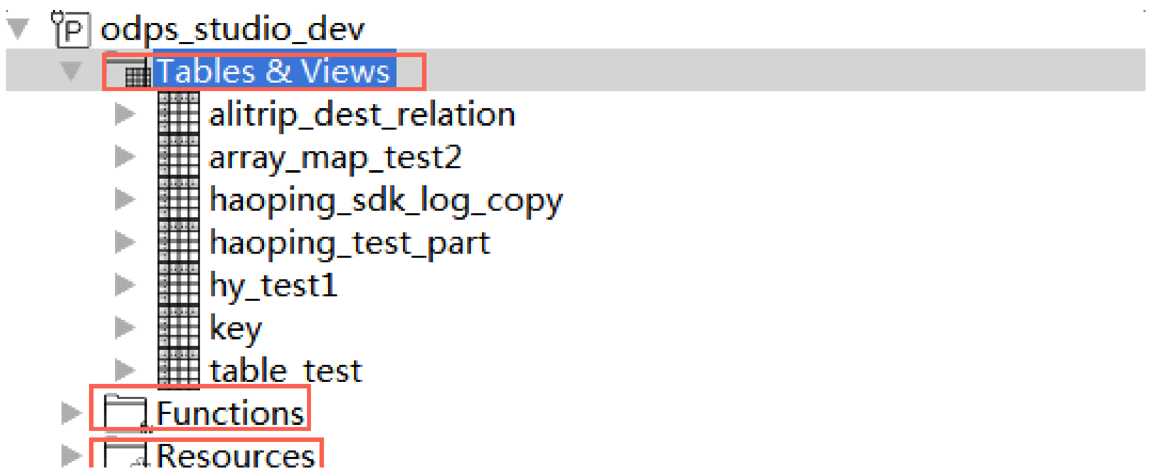
1. Open the Project Explorer window and you can view the added Project node tree.


MaxCompute Project Explorer
odps_studio_dev
Tables & Views
Functions
Resources

The toolbar is displayed at the top of the node tree, and includes:

- Add Project: Adds a connection to the MaxCompute project space.
- Delete Project: Deletes a connection from Project Explorer, which has no impact on the project space on the server end.
- Update Metadata: Updates metadata information from the project space on the server end and updates the locally buffered metadata.
- Expand Node: Expands all tree nodes.
- Fold Node: Folds all tree nodes.
- User Feedback: Submits user feedback.
- Online Documentation: Opens online documents.

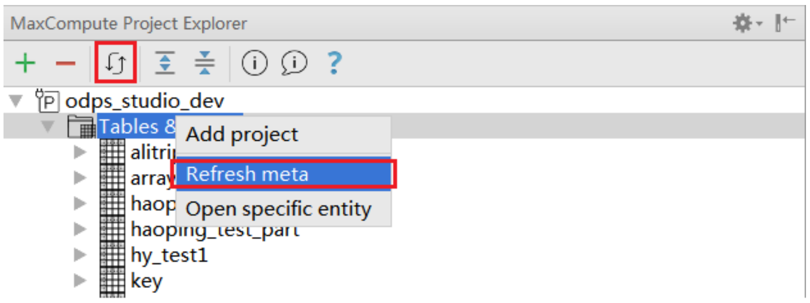
2. Double-click the Tables node or click the drop-down arrow to expand the Tables node to list all tables in the project (including virtual views). The table name list serves the same purpose as the show tables command. You must have the List Table permission in the project. The methods for the Functions and Resources nodes are similar to that of the Tables node.


odps_studio_dev
Tables & Views
alitrip_dest_relation
array_map_test2
haoping_sdk_log_copy
haoping_test_part
hy_test1
key
table_test
Functions
Resources

3. MaxCompute Studio downloads project metadata on the server to the local device . When metadata on the server end is updated, for example, a new table is added, you must manually trigger a refresh to reload changed metadata to the local device . The refresh can be performed at the Project or Table level. The procedure is as follows:

   a. Select a node.

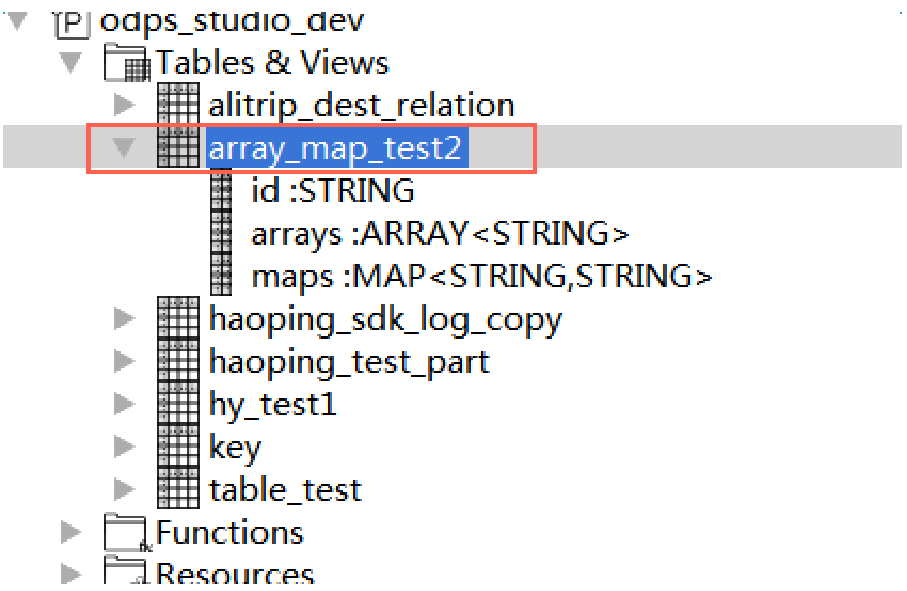
   b. Click the Refresh icon on the toolbar or right-click the node and select Refresh meta.



View table details

You can view data table information in Table Details View of MaxCompute Studio.

1. In the node tree, expand a table node to view the column name and type.



2. Double-click a table or right-click a table and choose Show Table Detail to view the table details. The table details include metadata, such as owner, size, and column, table structure information, and data preview.

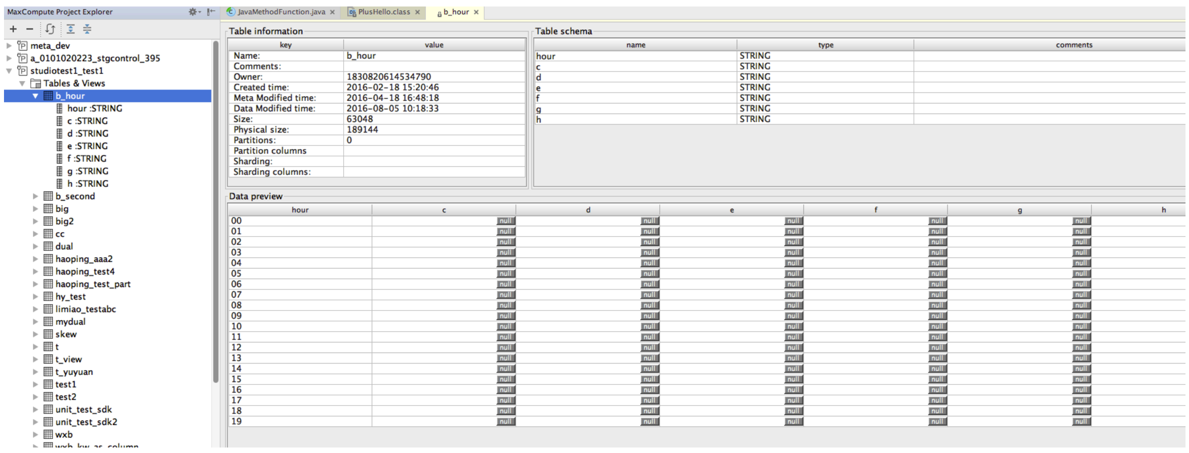


Table information

key	value
Name:	b_hour
Comments:	
Owner:	1830820614534790
Created time:	2016-02-18 15:20:46
Meta Modified time:	2016-04-18 16:48:18
Data Modified time:	2016-08-05 10:18:33
Size:	63048
Physical size:	189144
Partitions:	0
Partition columns	
Sharding:	
Sharding columns:	

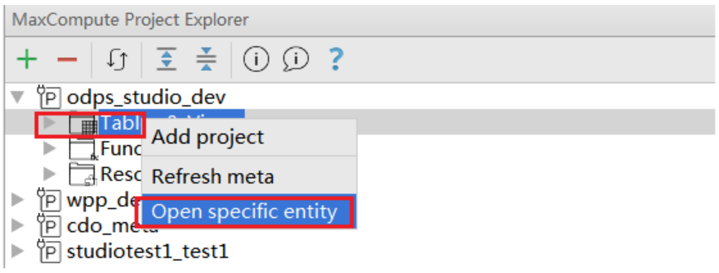
Table schema

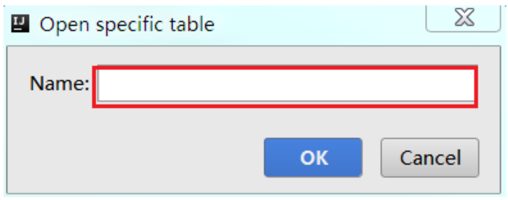
name	type	comments
hour	STRING	
c	STRING	
d	STRING	
e	STRING	
f	STRING	
g	STRING	
h	STRING	

Data preview

hour	c	d	e	f	g	h
00	null	null	null	null	null	
01	null	null	null	null	null	
02	null	null	null	null	null	
03	null	null	null	null	null	
04	null	null	null	null	null	
05	null	null	null	null	null	
06	null	null	null	null	null	
07	null	null	null	null	null	
08	null	null	null	null	null	
09	null	null	null	null	null	
10	null	null	null	null	null	
11	null	null	null	null	null	
12	null	null	null	null	null	
13	null	null	null	null	null	
14	null	null	null	null	null	
15	null	null	null	null	null	
16	null	null	null	null	null	
17	null	null	null	null	null	
18	null	null	null	null	null	
19	null	null	null	null	null	

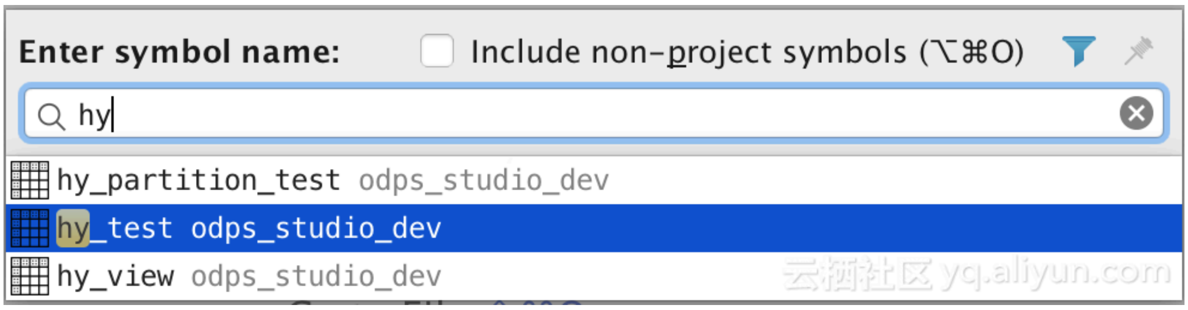
3. Right-click Tables & Views and select Open specific entity to display the details of the specific table. Note that the complete table name must be specified. If you do not have the List permission on the project and only have the permission on a specific table, you can also view details of the table using this method. The methods for the Functions and Resources nodes are similar to that of the Tables node.





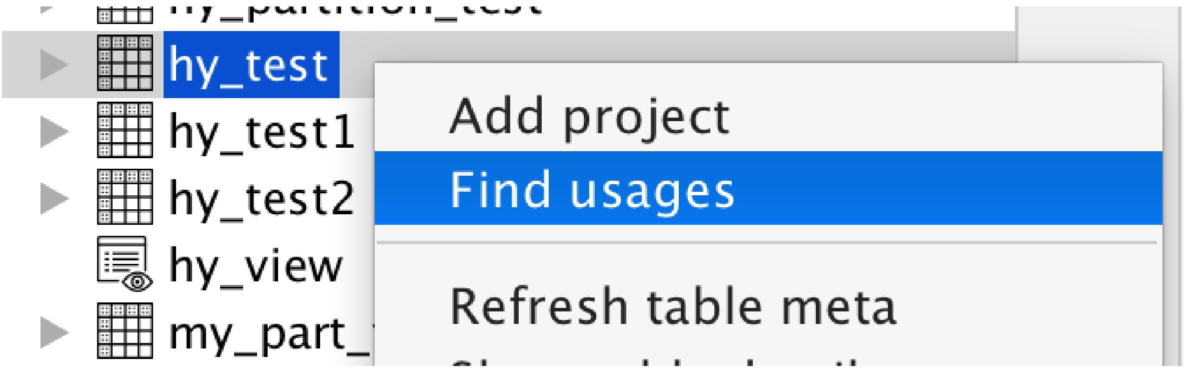
Intellij IDEA supports searching by default. After a table is expanded, you can directly press keys on the keyboard to perform fuzzy match.

4. MaxCompute Studio also supports quick search for the table, you can use the shortcut key (Windows: Ctrl + Alt + Shift + N, macOS: # + ⌘ + O) to call the navigation bar, then enter the name of the table and press Enter.



You can narrow the search by using the pre-keyword (table:, function:, or resource :). For example, to search for the function count, enter function:count.

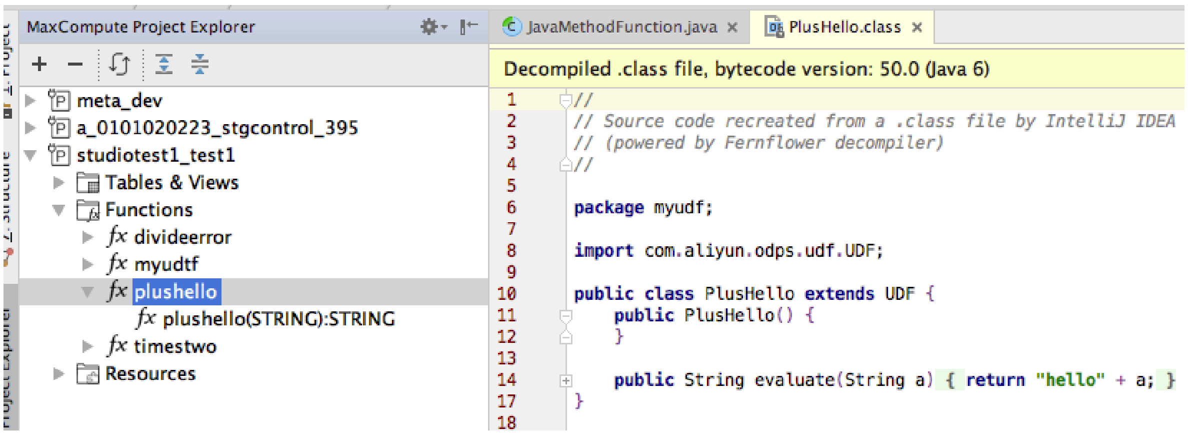
5. To know the scripts in which the table is used, right-click the table and select Find usages.



View function details

1. Expand a function node under the UserDefined node of Functions to display the method signature of this function. Double-click a function node under the Functions node. Alternatively, double-click the source code resource of the

function under the Resources node. In this case, codes of this function are displayed.



Decompiled .class file, bytecode version: 50.0 (Java 6)

```
//
// Source code recreated from a .class file by IntelliJ IDEA
// (powered by Fernflower decompiler)
//

package myudf;

import com.aliyun.odps.udf.UDF;

public class PlusHello extends UDF {
    public PlusHello() {
    }

    public String evaluate(String a) { return "hello" + a; }
}
```

Note:
The Java code is obtained by decompiling JAR, which is not the source code. To enable the Python UDF to parse the signature, install PyODPS (MaxCompute Python SDK) first. Install `pip : sudo / usr / bin / python get - pip . py` (Download `get - pip . py` from Google manually) and then `PyODPS : sudo / usr / bin / python - m pip install PyODPS` . Note that the Mac operating system has Python, which is stored in /usr/bin/python. Install PyODPS in this directory.

2. The classification under the BuiltIn node of Functions shows the built-in functions of the system, expand it to display signature and double-click it to display function document.

2.4.2 Import and export data

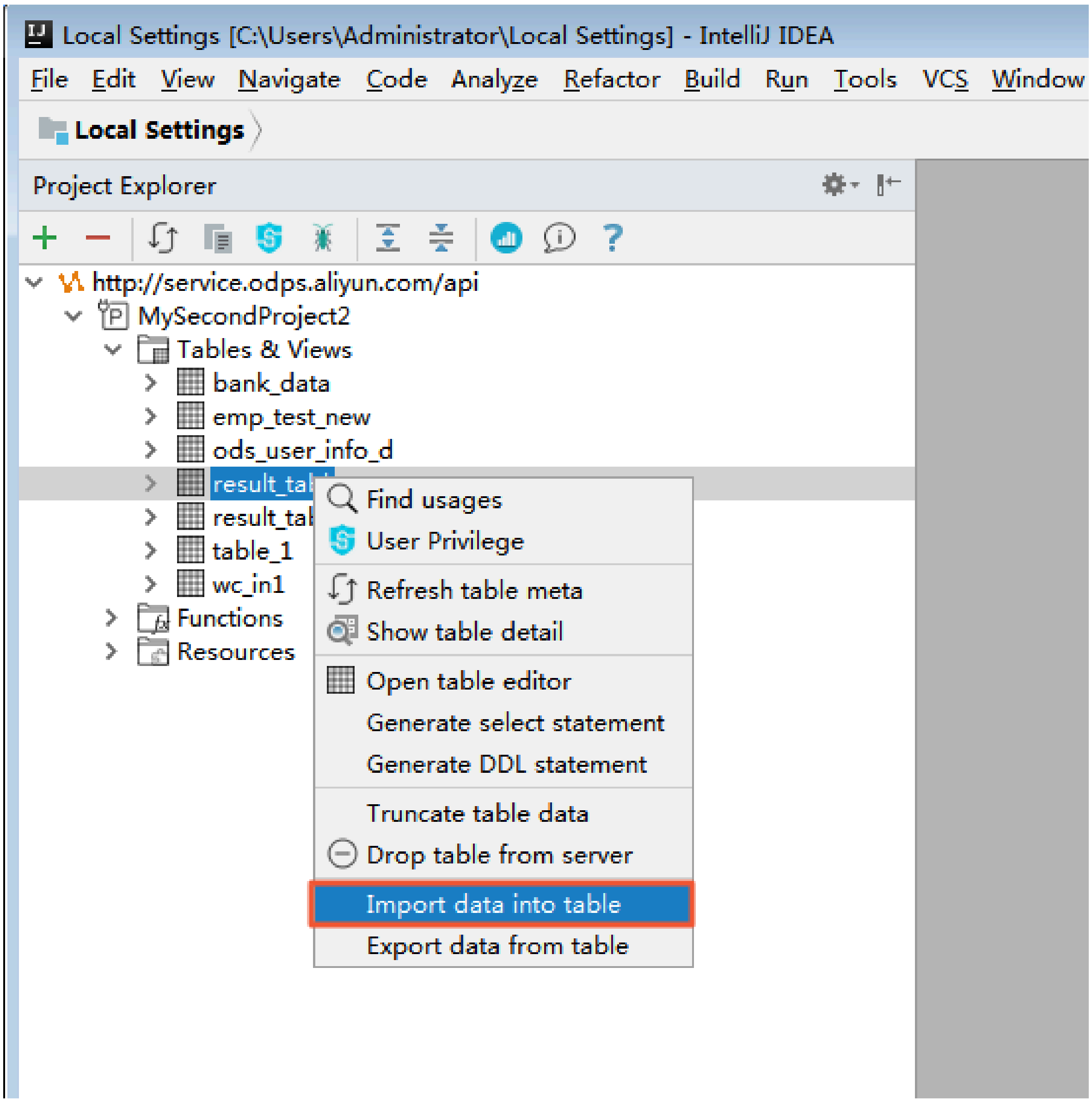
MaxCompute Studio can import local data files in CSV or TSV format to MaxCompute tables and export MaxCompute table data to local files. MaxCompute Studio completes data import and export by using Batch data Tunnel provided by the MaxCompute platform.

Usage instructions

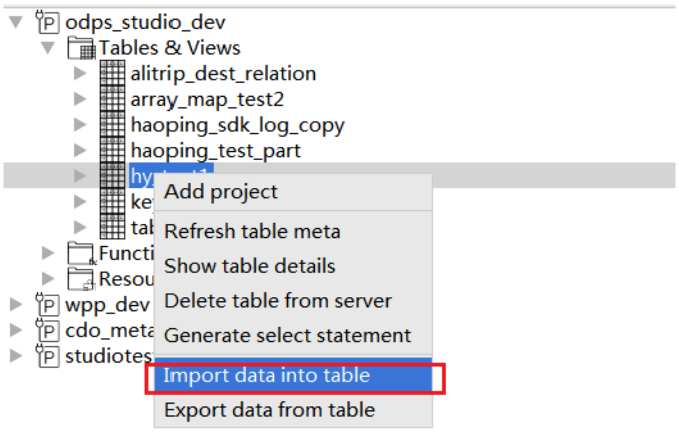
- The MaxCompute Tunnel service must be used for data import and export. Therefore, the MaxCompute project added in Studio must be configured with the Tunnel service.
- Related permissions must be granted for table import and export.

Import data

1. Open the Project Explorer window, right-click a table name or a field attribute in Data preview of Table details and select Import Data Into Table.



2. In the Import Data dialog box that appears, select the path of the imported data file, column separator, size limitation, and number of lines for an error tolerance, and click OK .

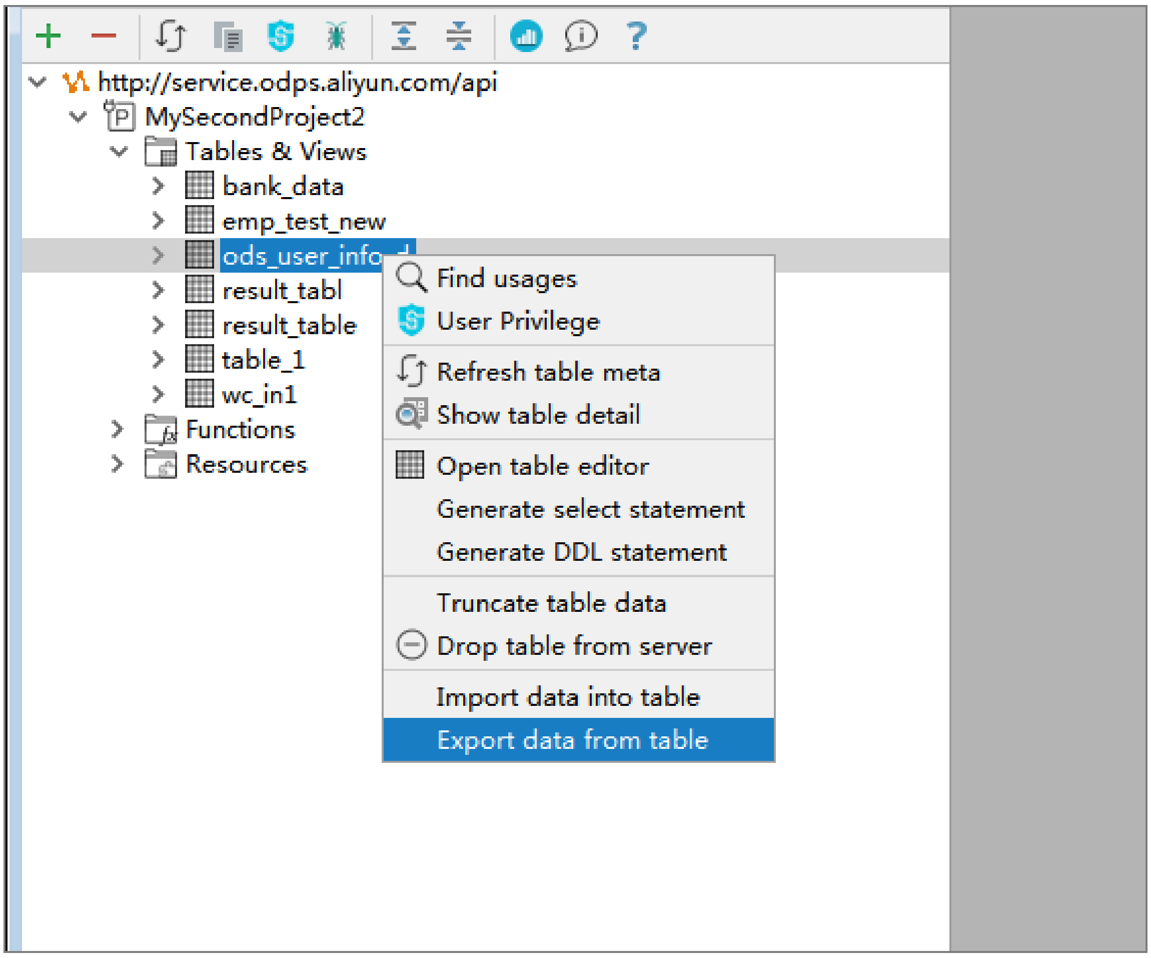


3. If Import Data Success is displayed, data import is successful and imported data can be viewed in the table.

Export data

1. Two methods are provided for table data export.

   - Right-click a table name and select Export Data From Table.



   - Right-click a field attribute in Data Preview of Table details and select Export Data From Table.

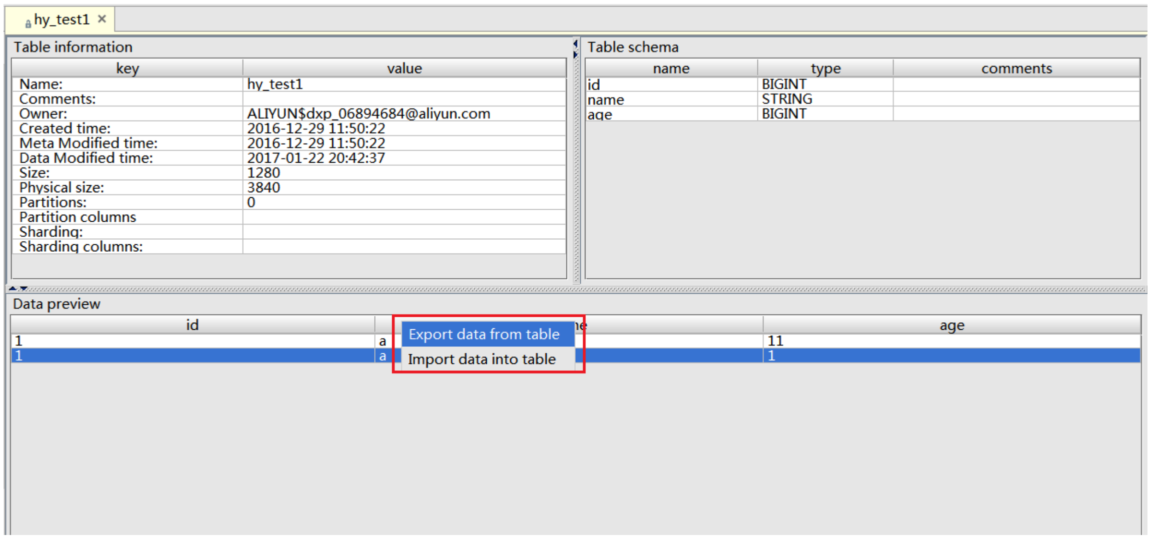


Table information

key	value
Name:	hy_test1
Comments:	
Owner:	ALIYUN$dxp_06894684@aliyun.com
Created time:	2016-12-29 11:50:22
Meta Modified time:	2016-12-29 11:50:22
Data Modified time:	2017-01-22 20:42:37
Size:	1280
Physical size:	3840
Partitions:	0
Partition columns	
Sharding:	
Sharding columns:	

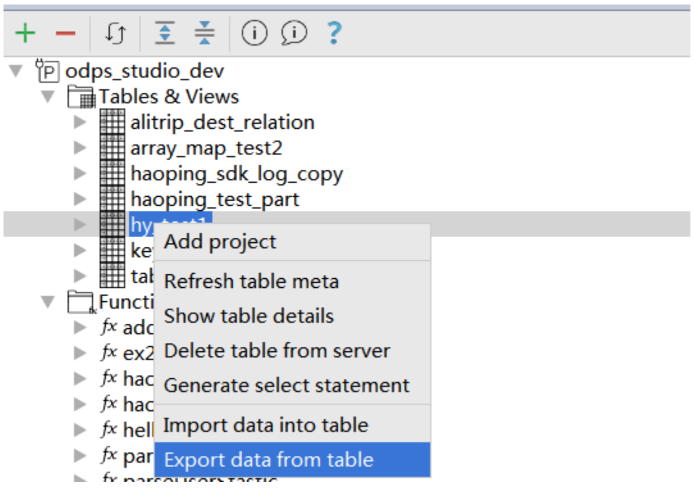
Table schema

name	type	comments
id	BIGINT	
name	STRING	
age	BIGINT	

Data preview

id	ne	age
1	a	11
1	a	1

2. In the Export Data dialog box that appears, select the path for saving the exported data file, column separator, size limitation, and number of lines for an error tolerance, and click OK.



3. If Export Data Success is displayed, data export is successful and exported data can be viewed in the target file.

You can also right-click Data Preview of Table and choose Export Grid Data to export data.

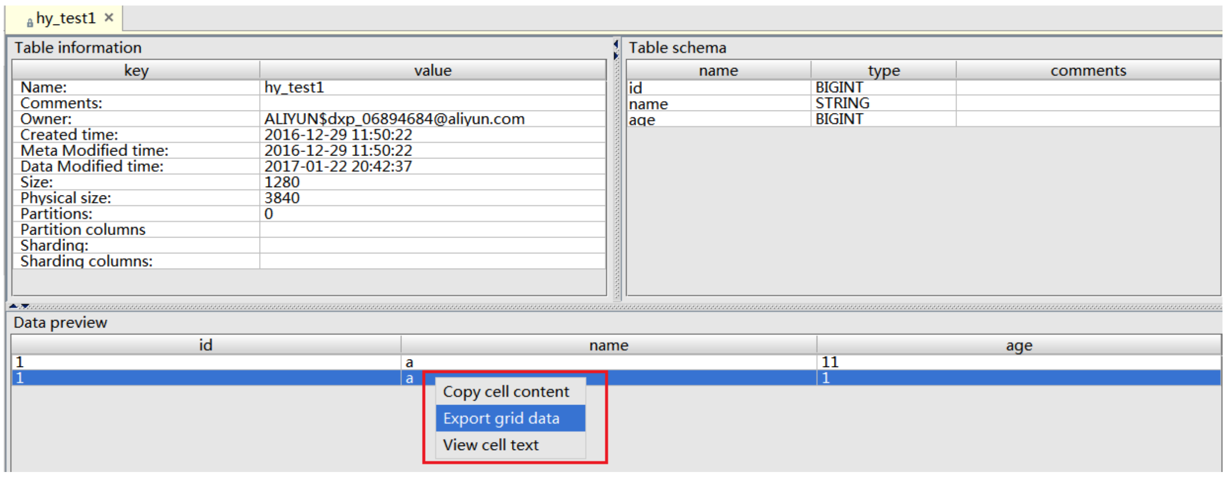

hy_test1

Table information

key	value
Name:	hy_test1
Comments:	
Owner:	ALIYUN$dxp_06894684@aliyun.com
Created time:	2016-12-29 11:50:22
Meta Modified time:	2016-12-29 11:50:22
Data Modified time:	2017-01-22 20:42:37
Size:	1280
Physical size:	3840
Partitions:	0
Partition columns	
Sharding:	
Sharding columns:	

Table schema

name	type	comments
id	BIGINT	
name	STRING	
age	BIGINT	

Data preview

id	name	age
1	a	11
1	a	1

Note:
The Data Preview function in Data preview is used only to export data displayed in Data sample instead of all data in the table.

New Type Import Export

Simply generate text in the agreed format and store it in CSV or TSV format, you can import to table through studio.

Conversion rules for each data type are described in detail below.

Basic Type

1. Tinyint, smallint, Int, bigint is stored directly as an integer string, and the numeric value exceeds the type boundary is reported as wrong
2. Float, double to store fractional strings or floating-point forms, such as: 2.342 1x + 7
3. Varchar is stored directly as a string, which is automatically truncated above the upper limit, and no errors will be reported
4. String is stored directly as a string
5. Decimal string that supports shaping or floating-point
6. Binary needs to encode binary data to base64 string
7. Datetime date time requires that the format specified in import dialog is consistent and that the format mismatches will be reported incorrectly
8. Timestamp timestamp needs to follow yyyy-[m] M-[d] d hh \: mm \: ss [. f...] Format is stored as a string
9. Boolean true or false string?

Composite Type

1. Array needs to be stored as a JSON array. The array elements are converted to strings according to the rules agreed in this article, array elements support any type.
2. Map needs to be stored as a JSON object, and map key, value are converted to a string according to the rules agreed in this article, value supports any type of nesting.
3. Struct needs to be stored as a JSON object, the struct field name is string, and the key converted to a JSON object, strect field values converted to JSON The value of the object, the value of the field that defines the rule transformation in this article.

Example

Array type

For Table structures as follows:

Column name	Column data types
c_1	ARRAY<TINYINT>
c_2	ARRAY<INT>
c_3	ARRAY<FLOAT>

Column name	Column data types
c_4	ARRAY<DATETIME>
c_6	ARRAY<TIMESTAMP>
c_7	ARRAY<STRING>

You can import data in the CSV format shown below:

```
 c_1 , c_2 , c_3 , c_4 , c_6 , c_7
"[" " 1 " "," " 2  "", " 3 " "]", "["  1  "", " 2 ", " 3 ", " 4 "],
 "[" " 1 . 2 " ","  2 . 0  ""] ", "[" " 3 - 00 : 00 : 00 ", " 3 -
 5 - 00 : 00 : 00  "", " 00 : 00 : 00 ", "["  At   00 : 00 : 00 .
 123456789  "", " At   00 : 00 : 00 .  123456789  "", " At   00 : 00 :
 00 .  123456789  ""] "," [" AAA " ","  Steamboat  "", " 4C " "]"
"["" 1 "",""  2 "","" 3 ""]","["" 1 "","" 2 "","" 3 "","" 4 ""]","["" 1
 . 2 "","" 2 . 0 ""]","["" 2017 - 11 - 11   00 : 00 : 00 "","" 2017 -
 11 - 11   00 : 00 : 00 "","" 2017 - 11 - 11   00 : 00 : 00 ""]","[""
 2017 - 11 - 11   00 : 00 : 00 . 123456789 "","" 2017 - 11 - 11   00
 : 00 : 00 . 123456789 "","" 2017 - 11 - 11   00 : 00 : 00 . 123456789
 ""]","["" aaa "","" bbb "","" ccc ""]"
"["" 1 "","" 2 "","" 3 ""]","["" 1 "","" 2 "","" 3 "","" 4 ""]","["" 1
 . 2 "","" 2 . 0 ""]","["" 2017 - 11 - 11   00 : 00 : 00 "","" 2017 -
 11 - 11   00 : 00 : 00 "","" 2017 - 11 - 11   00 : 00 : 00 ""]","[""
 2017 - 11 - 11   00 : 00 : 00 . 123456789 "","" 2017 - 11 - 11   00
 : 00 : 00 . 123456789 "","" 2017 - 11 - 11   00 : 00 : 00 . 123456789
 ""]","["" aaa "","" bbb "","" ccc ""]"
```

Note:

The CSV format needs to escape double quotes, which are expressed by two double quotes, you can refer specifically to the CSV format specification.

Map Type

For Table structures as follows:

Column name	Column data types
c_1	MAP<TINYINT,STRING>
c_2	MAP<STRING,INT>
c_3	MAP<FLOAT,STRING>
c_4	MAP<STRING,DATETIME>
c_5	MAP<STRING,STRING>
c_6	MAP<TIMESTAMP,STRING>

You can import data in the CSV format shown below:

```
 c_1 , c_2 , c_3 , c_4 , c_5 , c_6
"{ 1 :" " 2345 " "}", "{"  123  "": " 2 ", " 3 ": " 4  ""}", "{ 2 .
 0 :" " 223445 " ",  1 . 2 :"  1111  ""}", "{" " AAA " ":" " hub11
   00 : 00 : 00  "", " 4C " ":"  China  " 11   00 : 00 : 00 " ","
 Steamboat  "": " 00 : 00 : 00 "} ","  ckey  "": " cvalue "} "," {""
 hub11   01 : 00 : 00 .  123456789  "": " dddd " "," " hub11   00 :
 00 : 00 .  123456789  "": " AAA " ","  027   11   00 : 01 : 00 .
 123456789  "": " DDD  ""}"
"{ 1 :" " 2345 " "}", "{"  123  "": " 2 ", " 3 ": " 4  ""}", "{ 2 .
 0 :" " 223445 " ",  1 . 2 :"  1111  ""}", "{" " AAA " ":" " hub11
   00 : 00 : 00  "", " 4C " ":"  China  " 11   00 : 00 : 00 " ","
 Steamboat  "": " 00 : 00 : 00 "} ","  ckey  "": " cvalue "} "," {""
 hub11   01 : 00 : 00 .  123456789  "": " dddd " "," " hub11   00 :
 00 : 00 .  123456789  "": " AAA " ","  027   11   00 : 01 : 00 .
 123456789  "": " DDD  ""}"
"{ 1 :"" 2345 ""}","{"" 123 "":"" 2 "","" 3 "":"" 4 ""}","{ 2 . 0 :""
 223445 "", 1 . 2 :"" 1111 ""}","{"" aaa "":"" 2017 - 11 - 11   00 :
 00 : 00 "","" ccc "":"" 2017 - 11 - 11   00 : 00 : 00 "","" bbb "":""
 2017 - 11 - 11   00 : 00 : 00 ""}","{"" ckey "":"" cvalue ""}","{""
 2017 - 11 - 11   01 : 00 : 00 . 123456789 "":"" dddd "","" 2017 - 11
 - 11   00 : 00 : 00 . 123456789 "":"" aaa "","" 2017 - 11 - 11   00 :
 01 : 00 . 123456789 "":"" ddd ""}"
```

Struct Type

For Table structures as follows:

Column name	Column data types
C_struct	<RUCT<x:INT,y:VARCHAR(256),z: STRUCT<a:TINYINT,b:STRING>>

You can import data in the CSV format shown below:

```
 c_struct
"{"" x "":"" 1000 "","" y "":"" varchar_te  st "","" z "":{"" a "":""
 123 "","" b "":"" stringdemo ""}}"
"{"" x "":"" 1000 "","" y "":"" varchar_te  st "","" z "":{"" a "":""
 123 "","" b "":"" stringdemo ""}}"
```

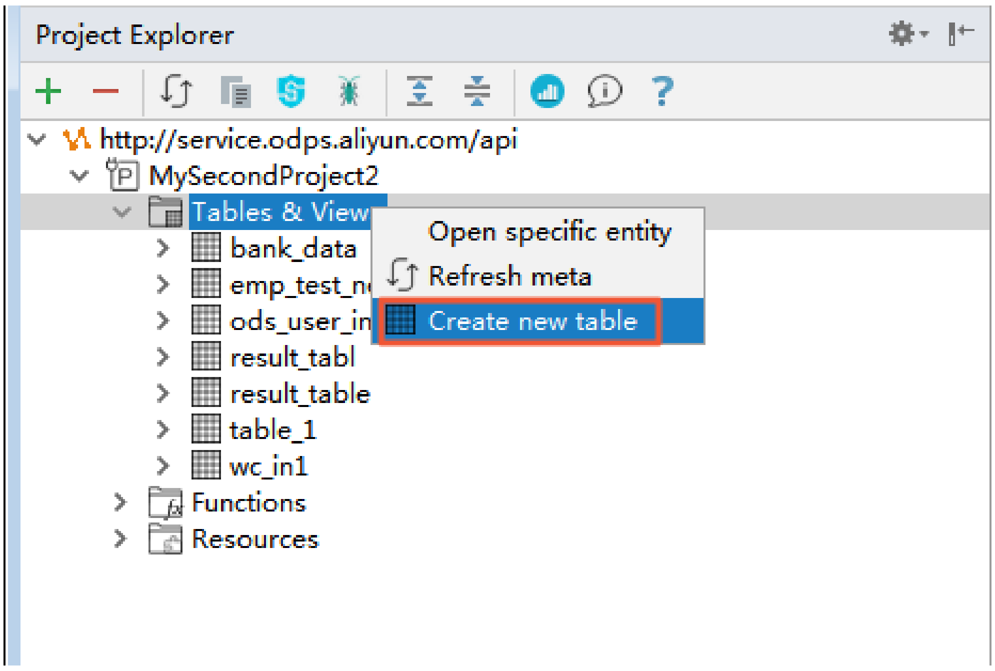
2.4.3 Visualization of operating the tables

The Project Explorer of MaxCompute Studio provides the visualized table structure editor used to create and modify tables.

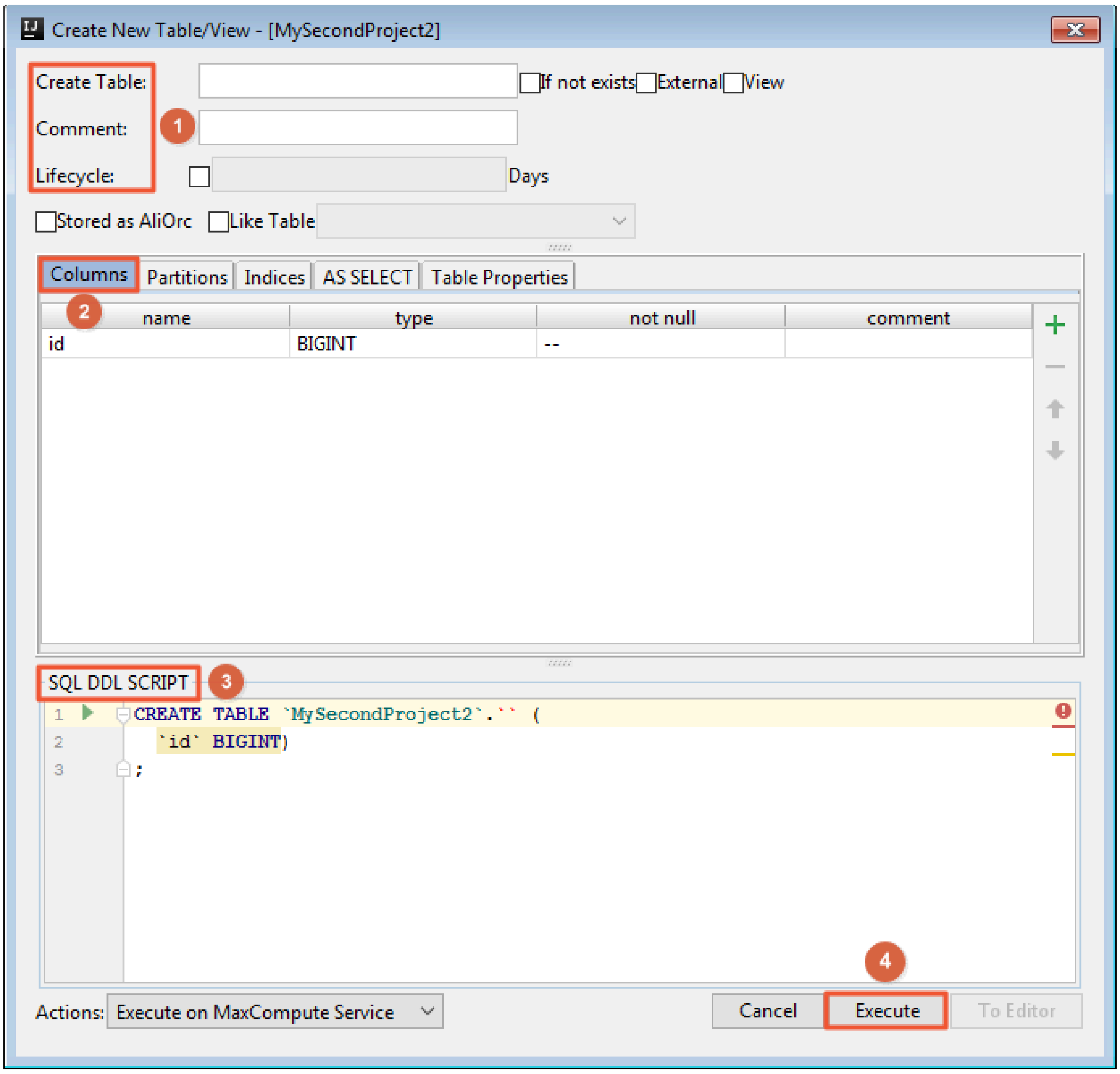
Visualization of creating a table

Procedure

1. Right-click the project that you want to create the table, and select create a new table.



2. In the dialog box that appears, enter a table name and column information. Click Generate CreateTable Statement generates the corresponding pant statement, click Execute to execute the build table.



name	type	not null	comment
id	BIGINT	--	

```
CREATE TABLE `MySecondProject2`.`` (
  `id` BIGINT)
;
```

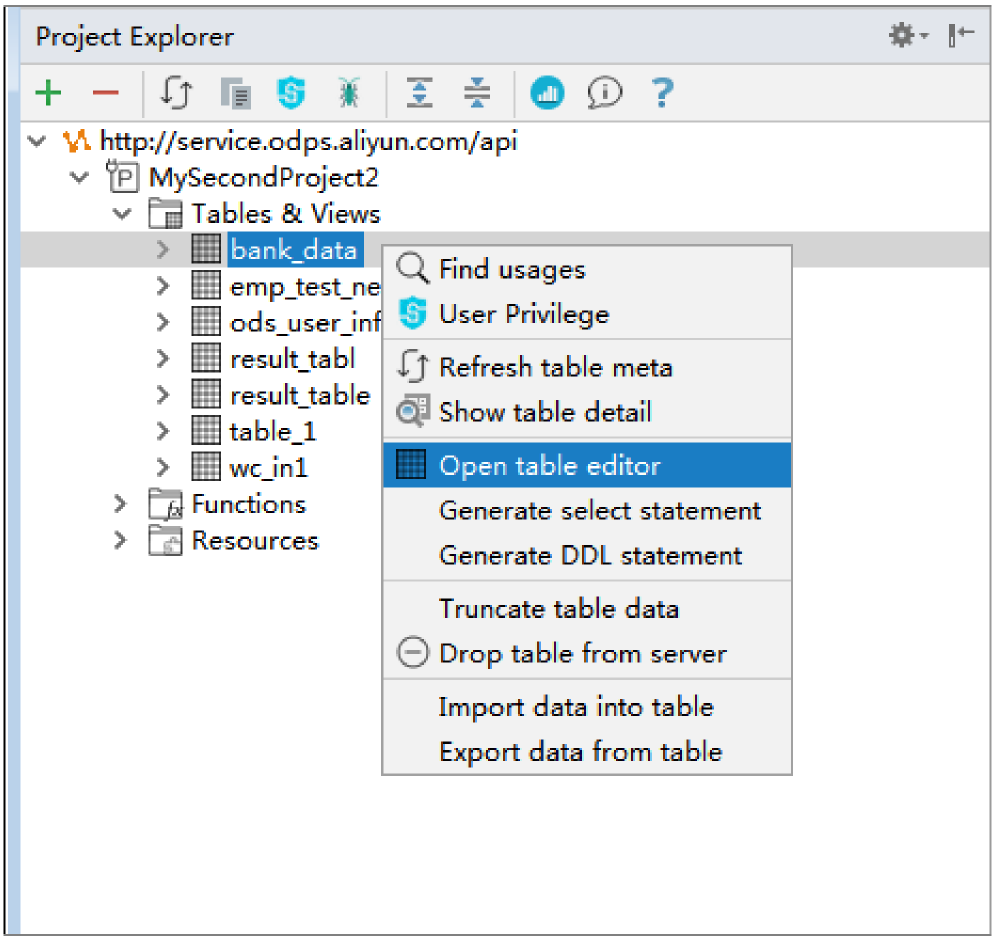
When you set the table name, column name, type, and lifecycle, observe the related requirements of MaxCompute. For more information, see*Table operations*.

3. After the table is created, view the table metadata in table&view of the Project Explorer. If no metadata is displayed, refresh the list.

Visualization of modifying a table

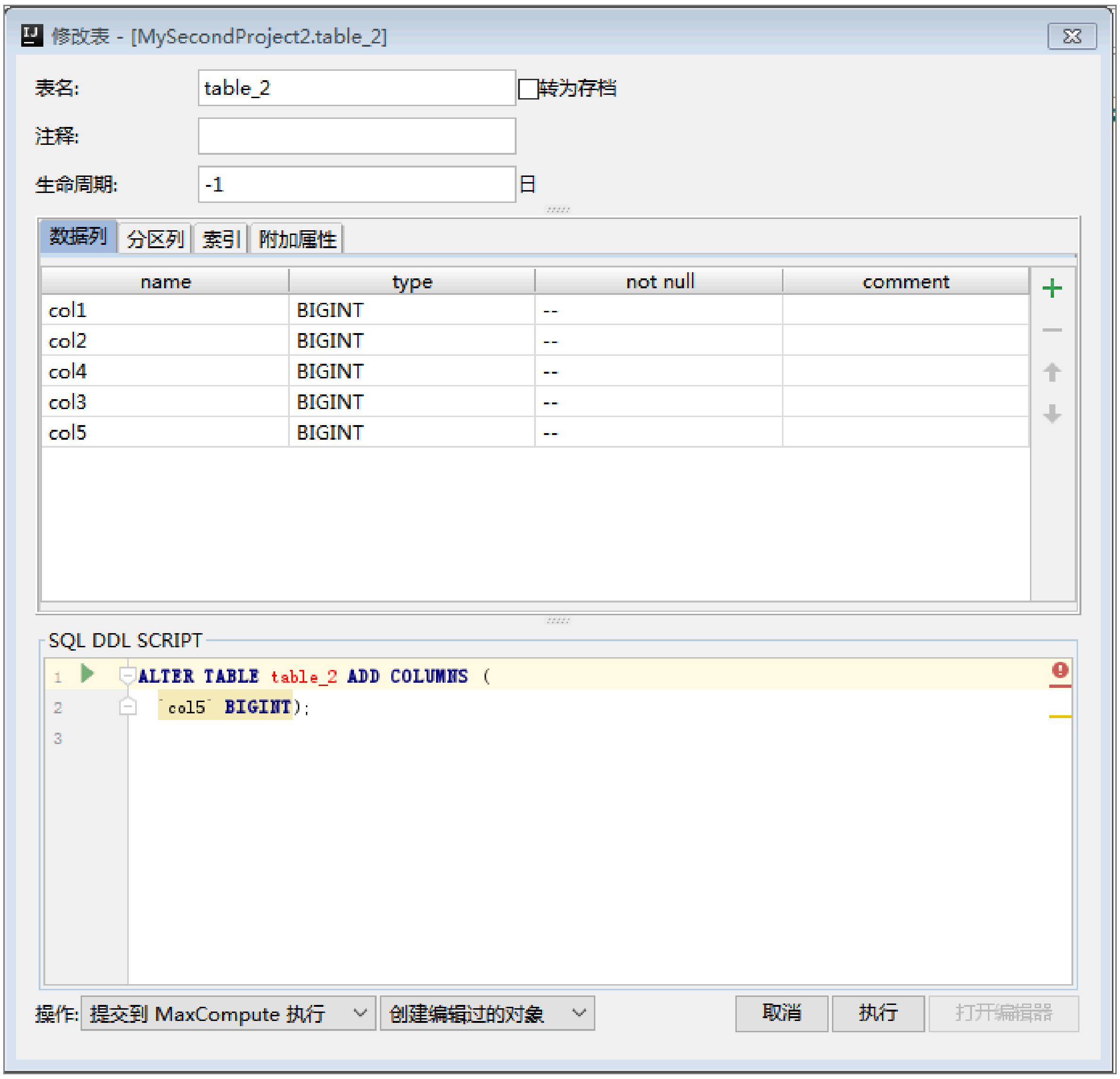
Procedure

1. Intable&view of the Project Explorer, right-click the expected table and select Open table Editor:



2. In the dialog box that appears, edit the table. You can modify the table comments, table lifecycle, column name, and column description, and add columns. Specific

rules follow the MaxCompute table requirements and can be found in*Table operations*.


修改表 - [MySecondProject2.table_2]

表名: table_2 ☐转为存档

注释:

生命周期: -1 日

数据列 | 分区列 | 索引 | 附加属性

name	type	not null	comment
col1	BIGINT	--	
col2	BIGINT	--	
col4	BIGINT	--	
col3	BIGINT	--	
col5	BIGINT	--	

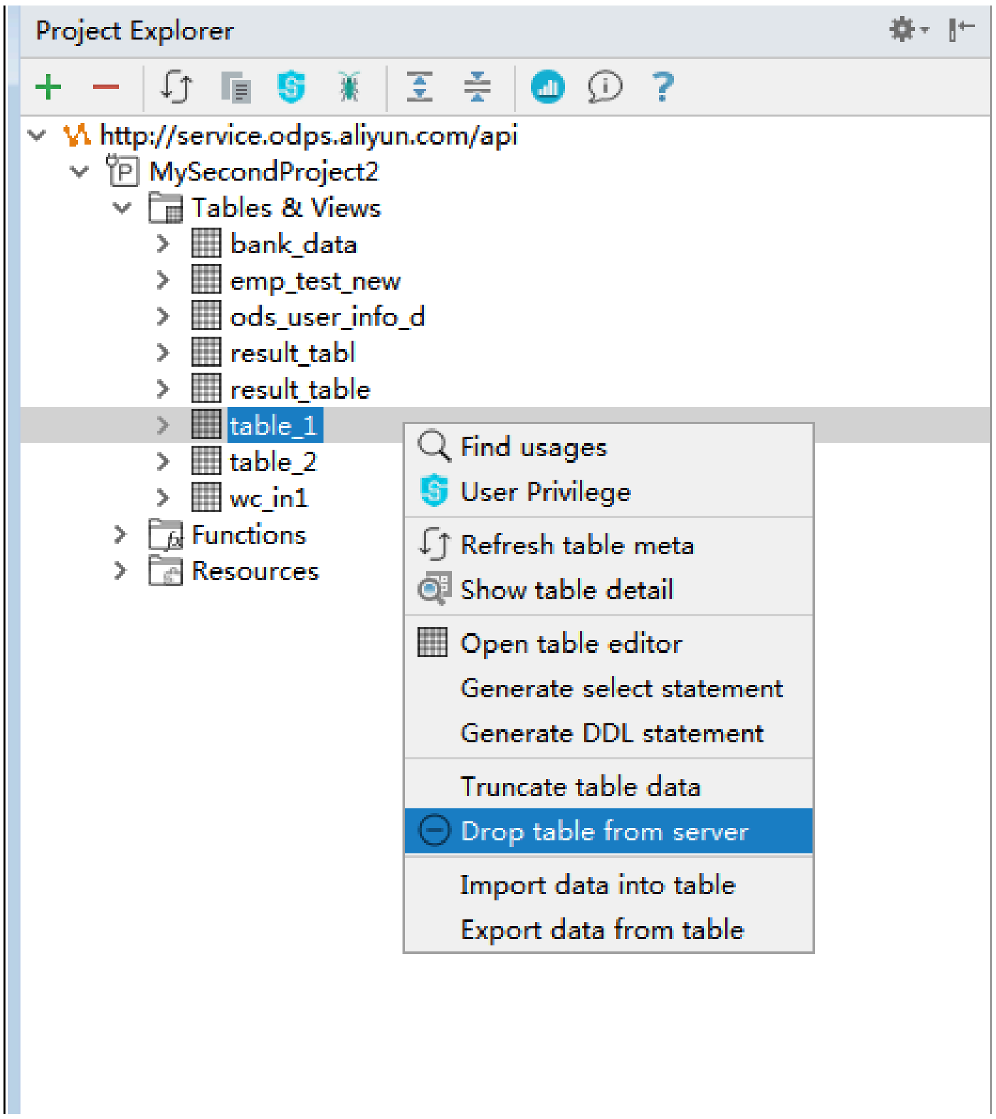
SQL DDL SCRIPT

```
ALTER TABLE table_2 ADD COLUMNS (
  `col5` BIGINT);
```

3. After completing the modifications, click Alter Table Statement Generate a specific alter statement and click Executeto perform the table modify operation. After successful execution, view the table metadata.

Visualization of deleting a table

In table&view of theProject Explorer, right-click the expected table and select Drop table from server:



Select OKin the bullet box to remove the table from the MaxCompute service.

2.5 Develop SQL procedure

2.5.1 Create MaxCompute Script module

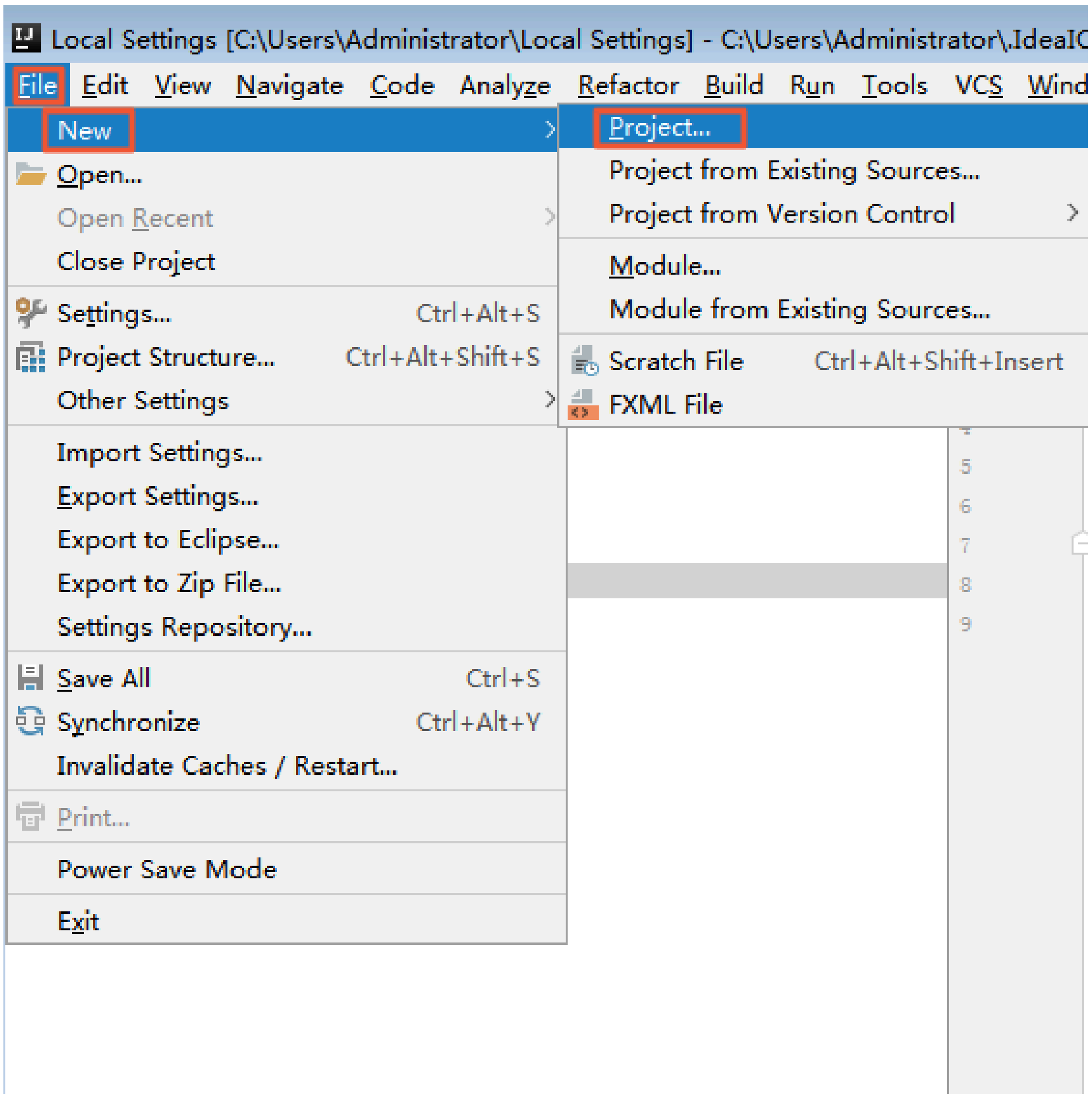
Before developing MaxCompute Script, you need to create a MaxCompute Script module in either of the following scenarios:

No script file exists locally

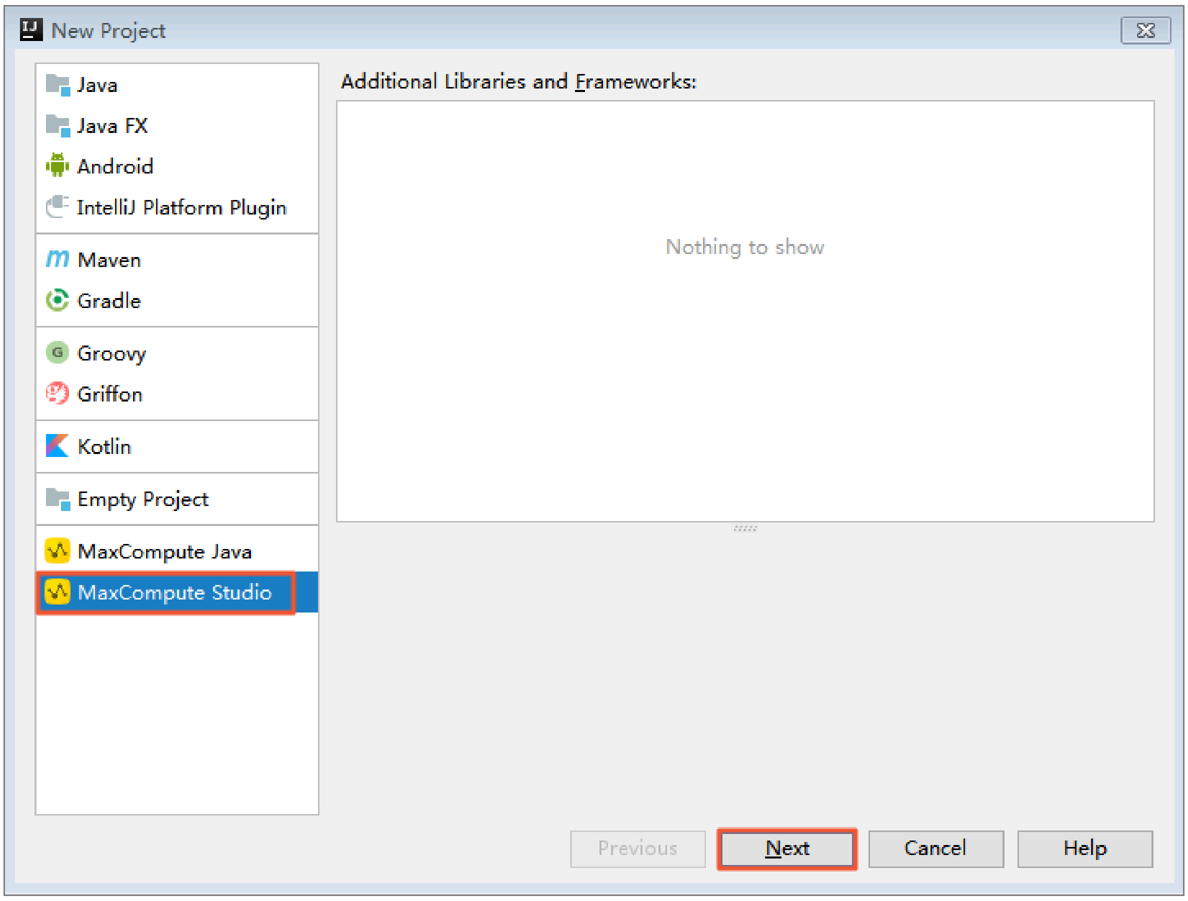
If no script file exists locally, you can use Intellij IDEA to create a new module.

Procedure

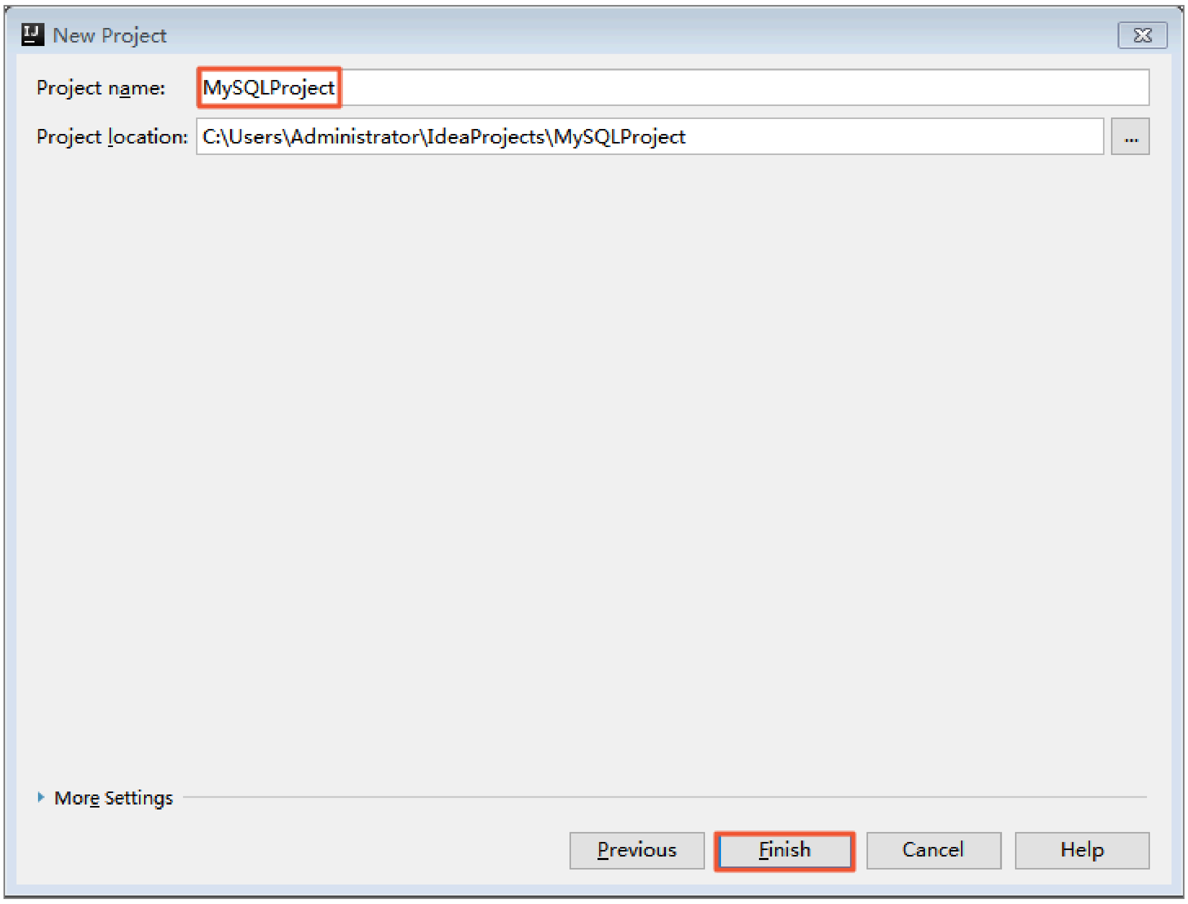
1. Open or create a MaxCompute Studio project. This article uses creating a project as an example. Click File in the menu and select New > Project . as shown in the following figure.



2. Select MaxCompute Studio on the left-side navigation pane, and click Next.



3. Enter the project name, and click Finish.



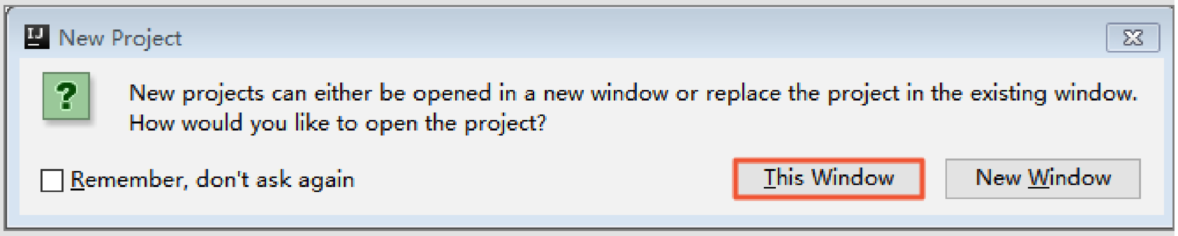
New Project

Project name: MySQLProject

Project location: C:\Users\Administrator\IdeaProjects\MySQLProject

Note:

If a project has been opened before, a dialog box appears, prompting whether to open the new project in the existing window (closing the previous project). Click This Window.

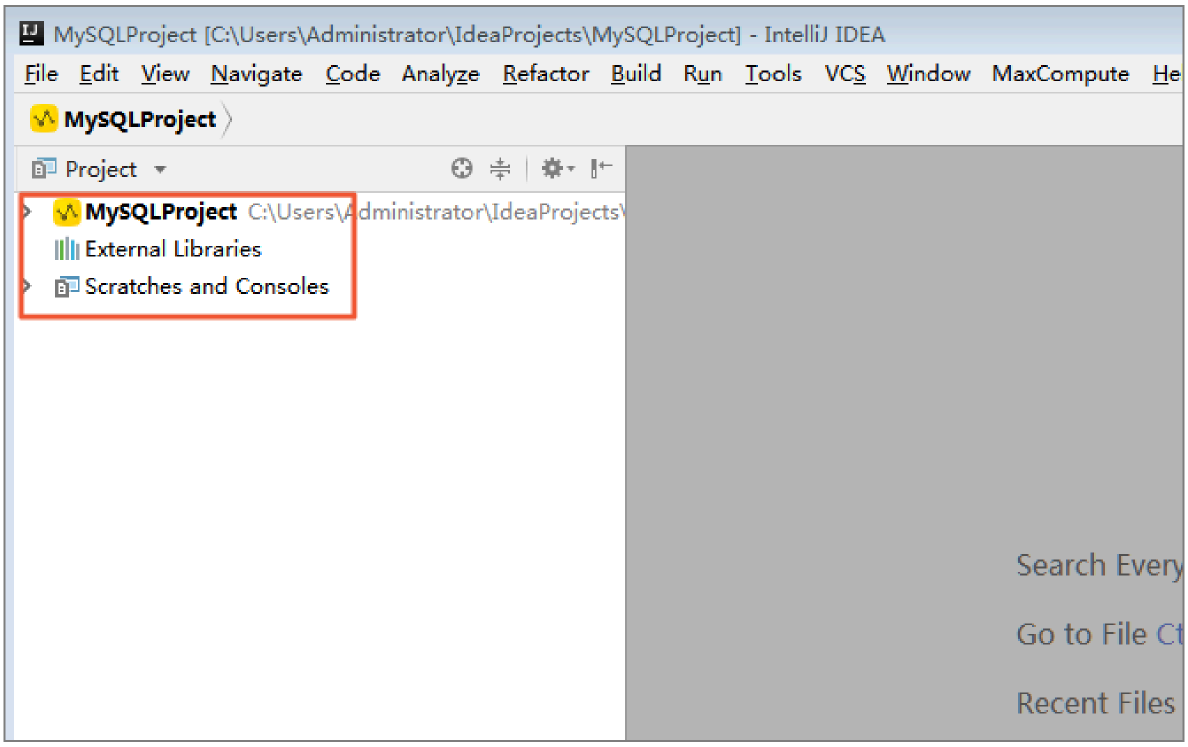


New Project

New projects can either be opened in a new window or replace the project in the existing window. How would you like to open the project?

☐ Remember, don't ask again

4. After the project is created, the page shown in the following figure appears. You can develop SQL scripts in the project.


MySQLProject [C:\Users\Administrator\IdeaProjects\MySQLProject] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window MaxCompute He
MySQLProject
Project
MySQLProject C:\Users\Administrator\IdeaProjects\
External Libraries
Scratches and Consoles
Search Every
Go to File Ct
Recent Files

Script files exist locally

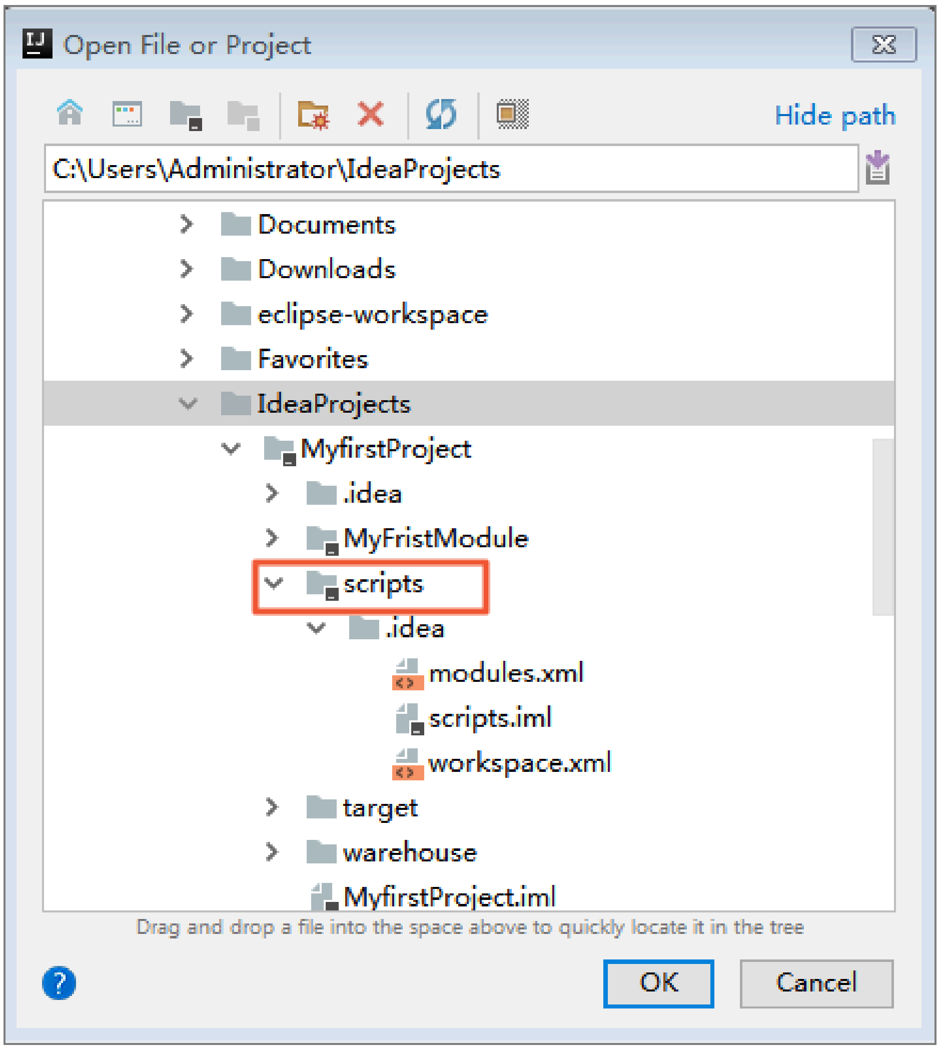
If many scripts have been stored in a local folder, MaxCompute Studio is used to edit the scripts. You can open a module directly.

Procedure

1. Create a connection configuration file *odps_config.ini* for MaxCompute in the scripts folder, and configure authentication information for connecting to MaxCompute.

   - project_name=xxxxxxxx
   - access_id=xxxxxxxxxx
   - access_key=xxxxxxxxx
   - end_point=xxxxxxxxx

2. Open IntelliJ IDEA, select File > > Open, and select the scripts folder.



3. MaxCompute Studio detects whether the odps_config.ini file exists in the folder, captures metadata on the server based on the configuration information in the file, and compiles all scripts in the folder.

2.5.2 Write SQL

Table operations

2.5.3 Submit SQL scripts

MaxCompute Studio directly submits MaxCompute SQL scripts to the server for running, and displays detailed information about the query result and execution plan. Before submission, MaxCompute Studio compiles scripts to effectively prevent compilation errors that are detected after the scripts are submitted to the server.

Prerequisite

- *Create a MaxCompute project connection* and bind it to the target project.

- Create a *MaxCompute Studio module*.
- Before submission, perform setting as required. MaxCompute Studio provides various setting features. You can perform quick setting on the toolbar at the top of the editor page. The following three types of setting can be performed:
  - Compiler Mode: It can be set to Script Mode or Statement Mode.
    - In statement mode, scripts are separated by `;` and submitted to the server one by one.
    - The script mode is newly developed. A whole script can be submitted to the server immediately. The server provides overall optimization, which is more efficient. Therefore, this mode is recommended.
  - Type System: It mainly solves the compatibility problem of SQL statements, which can be set to the following values:
    - Legacy TypeSystem: Indicates the type system of original MaxCompute.
    - MaxCompute TypeSystem: Indicates the new type system introduced by MaxCompute 2.0.
    - Hive Compatible TypeSystem: Indicates the type system in Hive compatibility mode introduced by MaxCompute 2.0.
  - Compiler Version: MaxCompute Studio provides the stable compiler and experimental compiler.
    - Default Version: Indicates the stable version.
    - Flighting Version: Includes the latest features of the compiler.

  Note:
  You can use Global Settings to set the submitted scripts. Select File > > Settings > > MaxCompute , select MaxCompute SQL, and choose Compiler > > Submit to set the preceding attributes.

Submit SQL scripts

The top toolbar of the editor provides the Synchronize and Compile and Submit features.

```
* ** Synchroniz  e **:  Updates   metadata   in   SQL   scripts ,
 including   table   names   and   UDFs .   If   MaxCompute   Studio
   prompts   that   a   table   or   function   cannot   be   found ,
 but   the   table   or   function   obviously   exists   on   the
 server ,  you   can   use   this   function   to   update   metadata
 .
```

```
* ** Compile   and   Submit **:  SQL   scripts   are   compiled
 or   submitted   to   the   server   in   compliance   with   pre -
 released   MaxCompute   SQL   rules .  Details   of   compilatio  n
   errors   are   displayed   in   the  ** MaxCompute   Compiler **
 window .
```

Procedure

1. After SQL statements are compiled, click the green running icon on the toolbar, or right-click Script Editor and select Run MaxCompute SQL Script to submit the SQL statement to the server. If a variable exists in the SQL statement (such as ${bizdate} in the following figure), a dialog box is displayed, prompting you to enter the variable value.
2. The script will be locally compiled (depending on the project metadata you added in the Project Explorer window). If no compilation error exists, the script is submitted to the server for execution. When the SQL script is being executed, the running logs are displayed. If the script is running on the server, the Job Details page is displayed, showing the basic information about job running and the execution diagram.
3. You can view SQL results on the Results page. If there are multiple statements in the single-sentence mode, the result of each statement is displayed. You can select rows or columns in the table, and copy them to the Clipboard.

2.6 Developing Java

2.6.1 Create MaxCompute Java Module

MaxCompute Studio supports Java user-defined function (UDF) and MapReduce development. First, a MaxCompute Java module must be created.

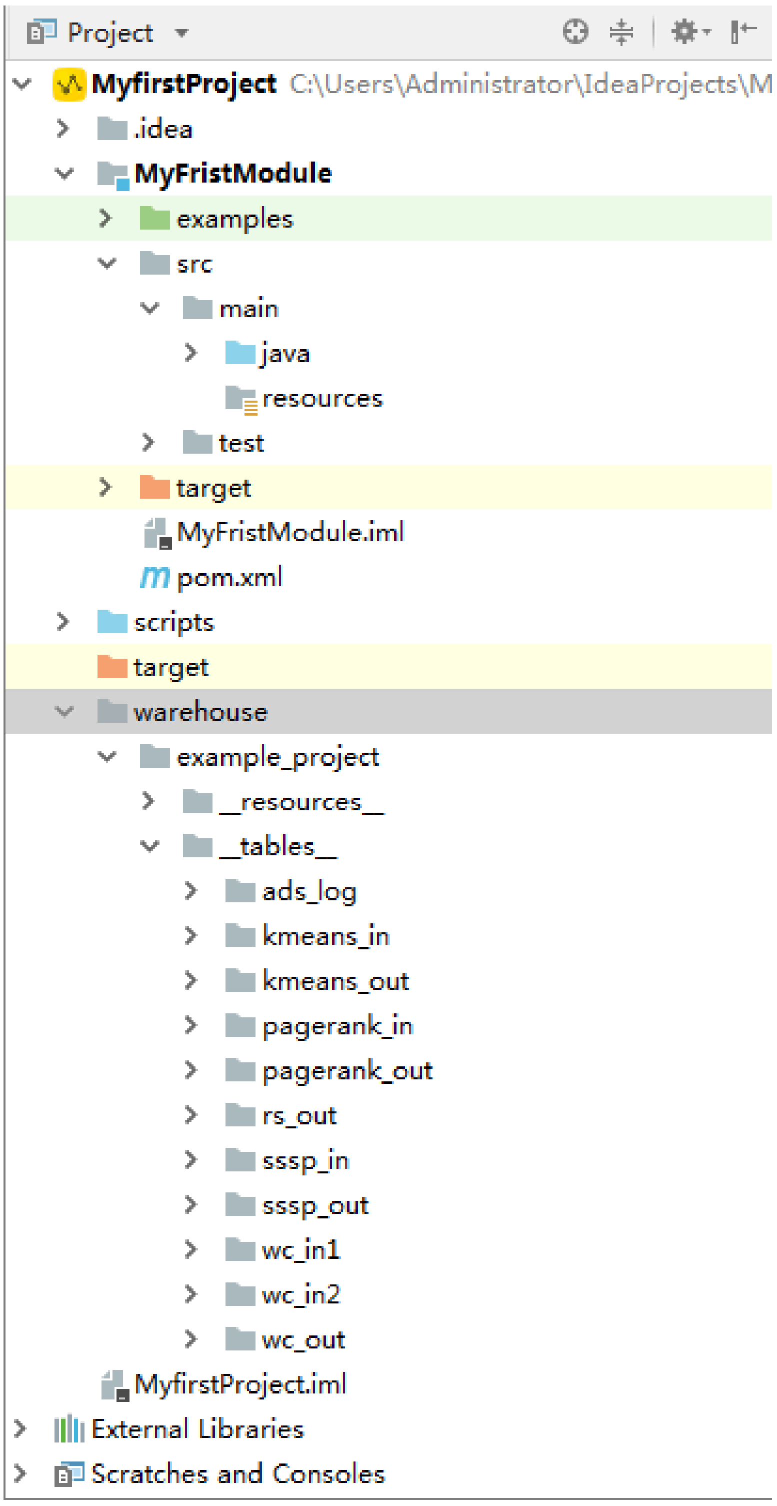
Create a module

ChooseFile > New > Module, set the module type to `MaxCompute   Java`, and configure `Java    JDK`. Click Next, enter a module name, and click Finish. MaxCompute Studio automatically creates a Maven module and introduces MaxCompute dependencies.

Module structure

So far, a module for developing a MaxCompute Java program has been established, that is the mDev shown in the following figure. Its main directories include:

- src/main/java: Source code for Java program development.
- examples: Sample code, including unit test (UT) examples. You can see the examples to develop or compile UT.
- warehouse: Schema and data required for running locally.



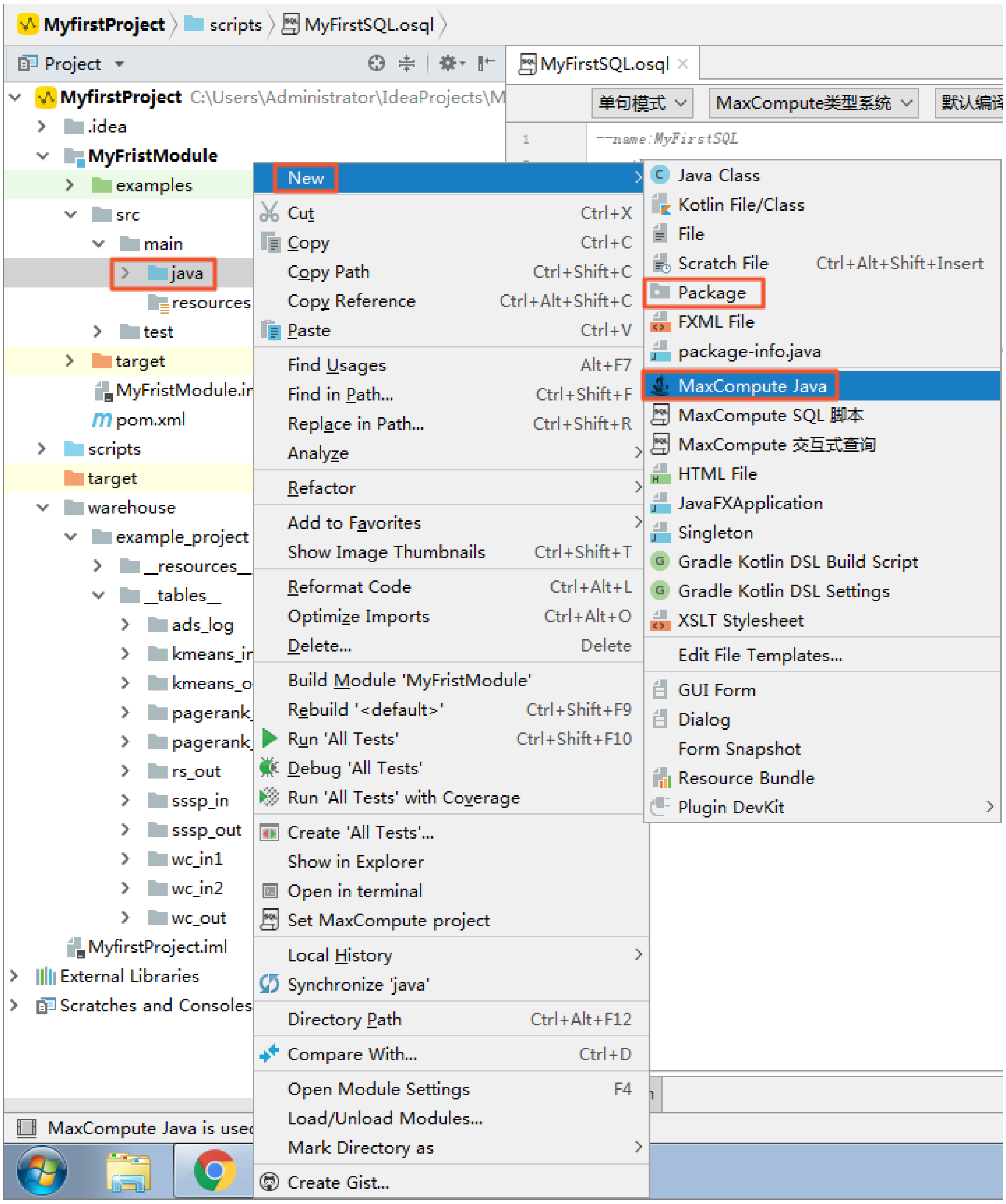
```
Project
MyfirstProject C:\Users\Administrator\IdeaProjects\M
  .idea
  MyFristModule
    examples
    src
      main
        java
        resources
      test
    target
    MyFristModule.iml
    pom.xml
  scripts
  target
  warehouse
    example_project
      __resources__
      __tables__
        ads_log
        kmeans_in
        kmeans_out
        pagerank_in
        pagerank_out
        rs_out
        sssp_in
        sssp_out
        wc_in1
        wc_in2
        wc_out
  MyfirstProject.iml
External Libraries
Scratches and Consoles
```

2.6.2 Develop and debug UDF

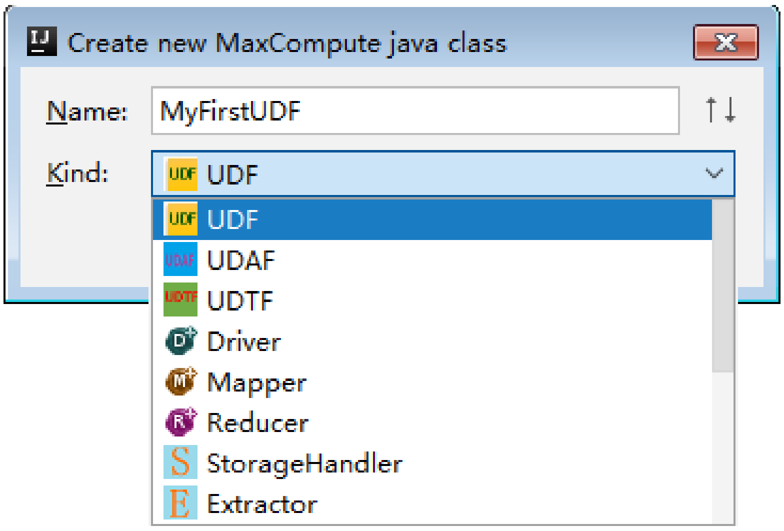
Once the MaxCompute Java Module has been created, udfs can be developed.

Procedure

1. Expand the MaxCompute Java Module Directory that you created and navigate to src > main > java > new, and click MaxCompute Java as shown in the following figure.



2. Set Name and Kind, and click OK. as shown in the following figure.

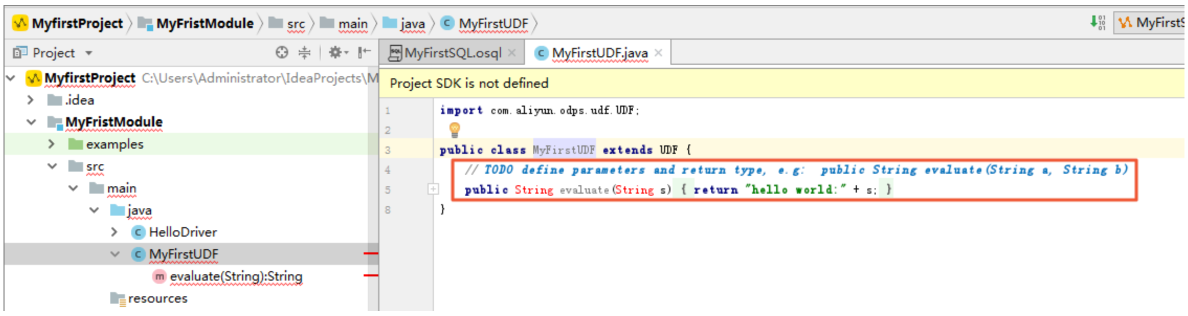


Name: MyFirstUDF

Kind: UDF

- Name: Specifies the name of the MaxCompute Java Class. If you have not created a package, you can enter packagename.classname to automatically create a package.
- Kind: Specifies the type. Supported types include custom functions (UDF/UDAF/UDTF), MapReduce (Driver/Mapper/Reducer), and non-structural development (StorageHandler/Extractor).

3. After the creation is successful, the Java program can be developed, modified, and tested.



Project SDK is not defined

```
import com.aliyun.odps.udf.UDF;

public class MyFirstUDF extends UDF {
    // TODO define parameters and return type, e.g:  public String evaluate(String a, String b)
    public String evaluate(String s) { return "hello world:" + s; }
}
```

Note:
Here's a code template that can be customized in Intellij. You can define it in Preference > Editor > File > Code Templates. Then look for the corresponding template in the Code tab.

For detailed development steps, see*JAVA UDF development*.

Normally, the development of JAVA UDF can be done in the following ways:

- Use MaxCompute Studio to complete the whole process of JAVA UDF development.

- Use *Develop and debug JAVA UDF using the Eclipse plug-in*, export the Jar package, then *Add resources* through commands or DataWorks, and *Register the function*.

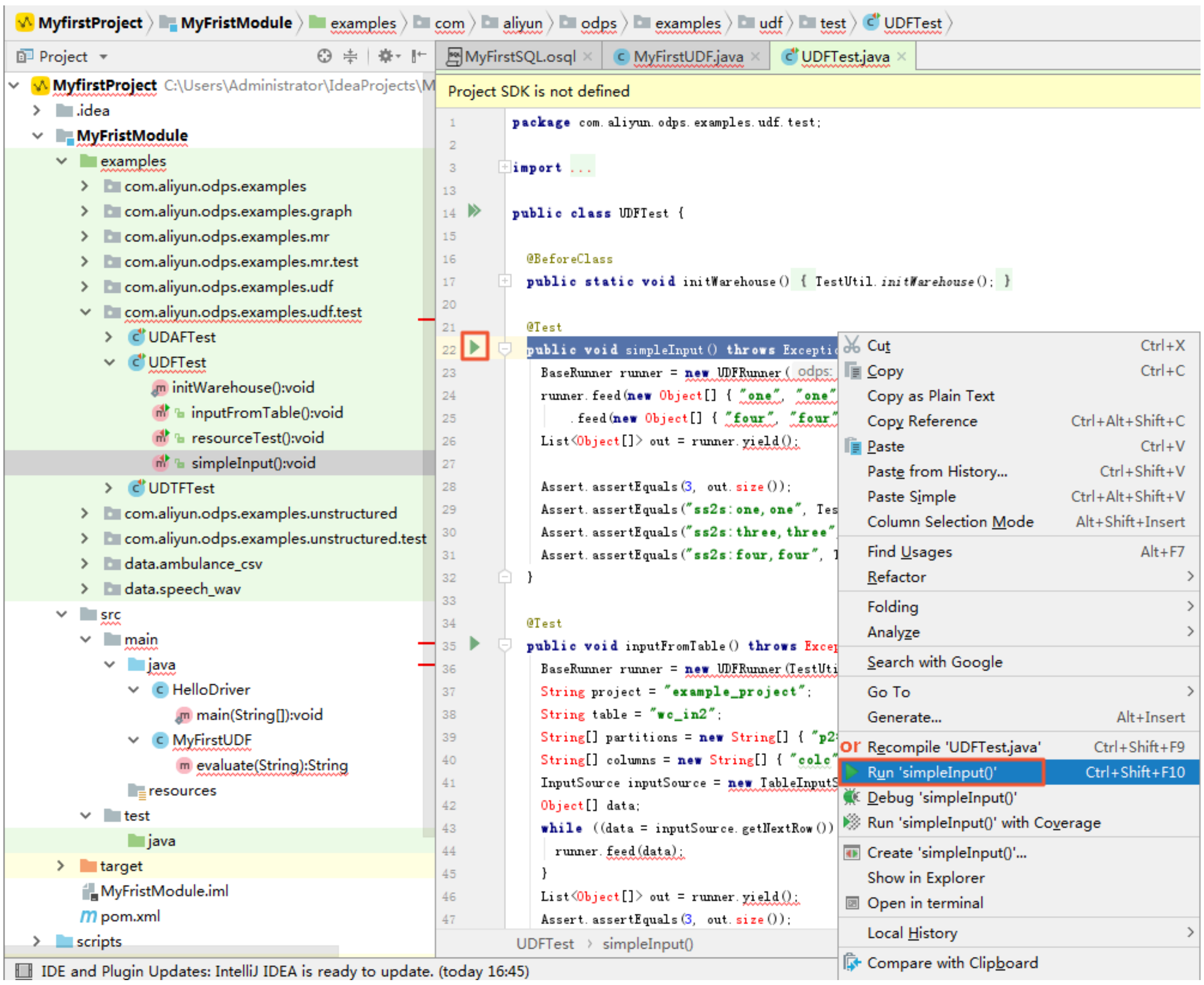
For detailed development steps, see*JAVA UDF Development*.

Debug the UDF program

After the UDF program is developed, it can be tested using unit test (UT) or local running to check whether it meets expectations.

Unit Testing

There are various UT examples in the examples directory and you can refer to them to compile your UT.



```
package com.aliyun.odps.examples.udf.test;

import ...

public class UDFTest {

  @BeforeClass
  public static void initWarehouse() { TestUtil.initWarehouse(); }

  @Test
  public void simpleInput() throws Exceptio
    BaseRunner runner = new UDFRunner( odps:
    runner.feed(new Object[] { "one", "one"
        .feed(new Object[] { "four", "four"
    List<Object[]> out = runner.yield();

    Assert.assertEquals(3, out.size());
    Assert.assertEquals("ss2s:one,one", Tes
    Assert.assertEquals("ss2s:three,three"
    Assert.assertEquals("ss2s:four,four", T
  }

  @Test
  public void inputFromTable() throws Excep
    BaseRunner runner = new UDFRunner(TestUti
    String project = "example_project";
    String table = "wc_in2";
    String[] partitions = new String[] { "p2
    String[] columns = new String[] { "colc"
    InputSource inputSource = new TableInputS
    Object[] data;
    while ((data = inputSource.getNextRow())
      runner.feed(data);
    }
    List<Object[]> out = runner.yield();
    Assert.assertEquals(3, out.size());
```

MyfirstProject
MyFristModule
examples
com
aliyun
odps
udf
test
UDFTest
Project
MyFirstSQL.osql
MyFirstUDF.java
UDFTest.java
Project SDK is not defined
MyfirstProject C:\Users\Administrator\IdeaProjects\M
.idea
com.aliyun.odps.examples
com.aliyun.odps.examples.graph
com.aliyun.odps.examples.mr
com.aliyun.odps.examples.mr.test
com.aliyun.odps.examples.udf
com.aliyun.odps.examples.udf.test
UDAFTest
initWarehouse():void
inputFromTable():void
resourceTest():void
simpleInput():void
UDTFTest
com.aliyun.odps.examples.unstructured
com.aliyun.odps.examples.unstructured.test
data.ambulance_csv
data.speech_wav
src
main
java
HelloDriver
main(String[]):void
MyFirstUDF
evaluate(String):String
resources
target
MyFristModule.iml
pom.xml
scripts
Cut Ctrl+X
Copy Ctrl+C
Copy as Plain Text
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Paste from History... Ctrl+Shift+V
Paste Simple Ctrl+Alt+Shift+V
Column Selection Mode Alt+Shift+Insert
Find Usages Alt+F7
Refactor
Folding
Analyze
Search with Google
Go To
Generate... Alt+Insert
or
Recompile 'UDFTest.java' Ctrl+Shift+F9
Run 'simpleInput()' Ctrl+Shift+F10
Debug 'simpleInput()'
Run 'simpleInput()' with Coverage
Create 'simpleInput()'...
Show in Explorer
Open in terminal
Local History
Compare with Clipboard
UDFTest › simpleInput()
IDE and Plugin Updates: IntelliJ IDEA is ready to update. (today 16:45)

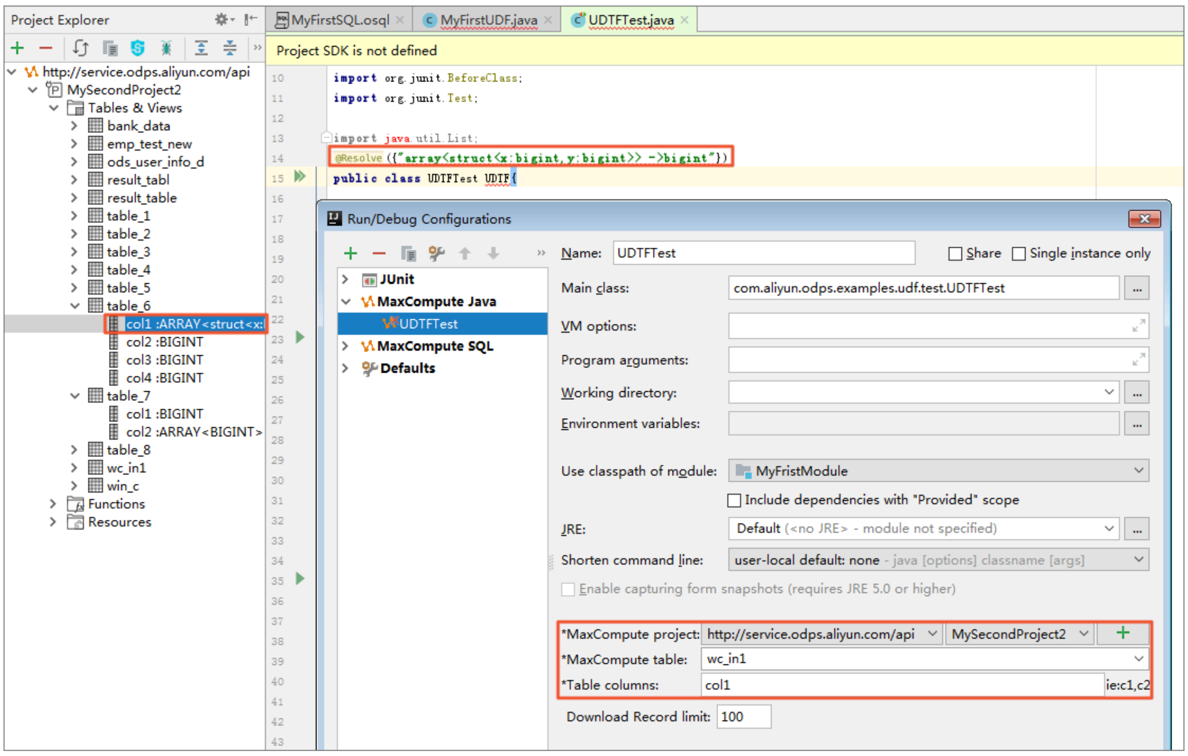
Run locally

During local running of the UDF program, the running data source must be specified. The following two methods are provided to set the test data source:

- MaxCompute Studio uses the Tunnel Service to automatically download table data of a specific project to the warehouse directory.
- The mock project and table data are provided. You can see example_project in warehouse to set it by yourself.

Procedure

1. Right-click UDF Class and select Run UDF class.main(). The Run Configuration dialog box is displayed. In normal cases, UDF/UDAF/UDTF data is used as columns in tables of a select sub-statement. The MaxCompute project, table, and column need to be configured. (The metadata is from the mock project under project explorer and warehouse.) Debugging for complex types is also supported, as shown in the following figure:



```
import org.junit.BeforeClass;
import org.junit.Test;

import java.util.List;
@Resolve({"array<struct<x:bigint,y:bigint>> ->bigint"})
public class UDTFTest UDTF{
```

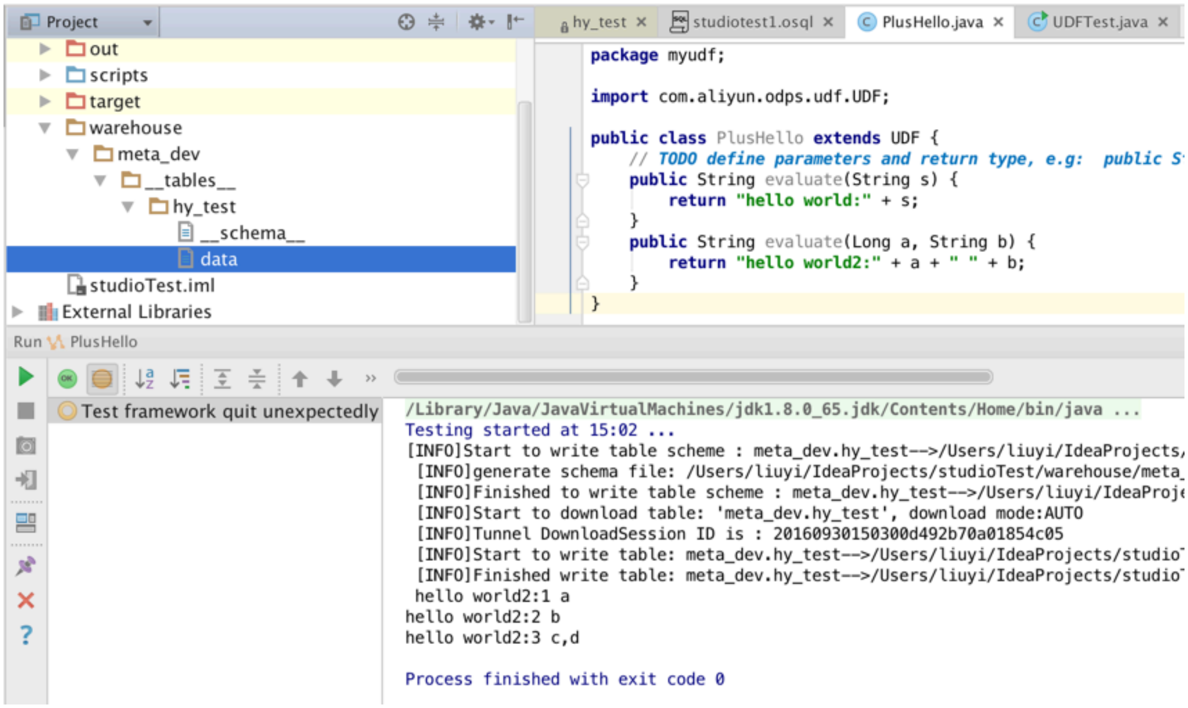
Project SDK is not defined

Project Explorer:
- http://service.odps.aliyun.com/api
  - MySecondProject2
    - Tables & Views
      - bank_data
      - emp_test_new
      - ods_user_info_d
      - result_tabl
      - result_table
      - table_1
      - table_2
      - table_3
      - table_4
      - table_5
      - table_6
        - col1 :ARRAY<struct<x:
        - col2 :BIGINT
        - col3 :BIGINT
        - col4 :BIGINT
      - table_7
        - col1 :BIGINT
        - col2 :ARRAY<BIGINT>
      - table_8
      - wc_in1
      - win_c
    - Functions
    - Resources

Run/Debug Configurations

Name:	UDTFTest
Main class:	com.aliyun.odps.examples.udf.test.UDTFTest
VM options:	
Program arguments:	
Working directory:	
Environment variables:	
Use classpath of module:	MyFristModule
	Include dependencies with "Provided" scope
JRE:	Default (<no JRE> - module not specified)
Shorten command line:	user-local default: none - java [options] classname [args]
	Enable capturing form snapshots (requires JRE 5.0 or higher)
*MaxCompute project:	http://service.odps.aliyun.com/api MySecondProject2
*MaxCompute table:	wc_in1
*Table columns:	col1 ie:c1,c2
Download Record limit:	100

2. Click OK.



```java
package myudf;

import com.aliyun.odps.udf.UDF;

public class PlusHello extends UDF {
    // TODO define parameters and return type, e.g:  public S
    public String evaluate(String s) {
        return "hello world:" + s;
    }
    public String evaluate(Long a, String b) {
        return "hello world2:" + a + " " + b;
    }
}
```

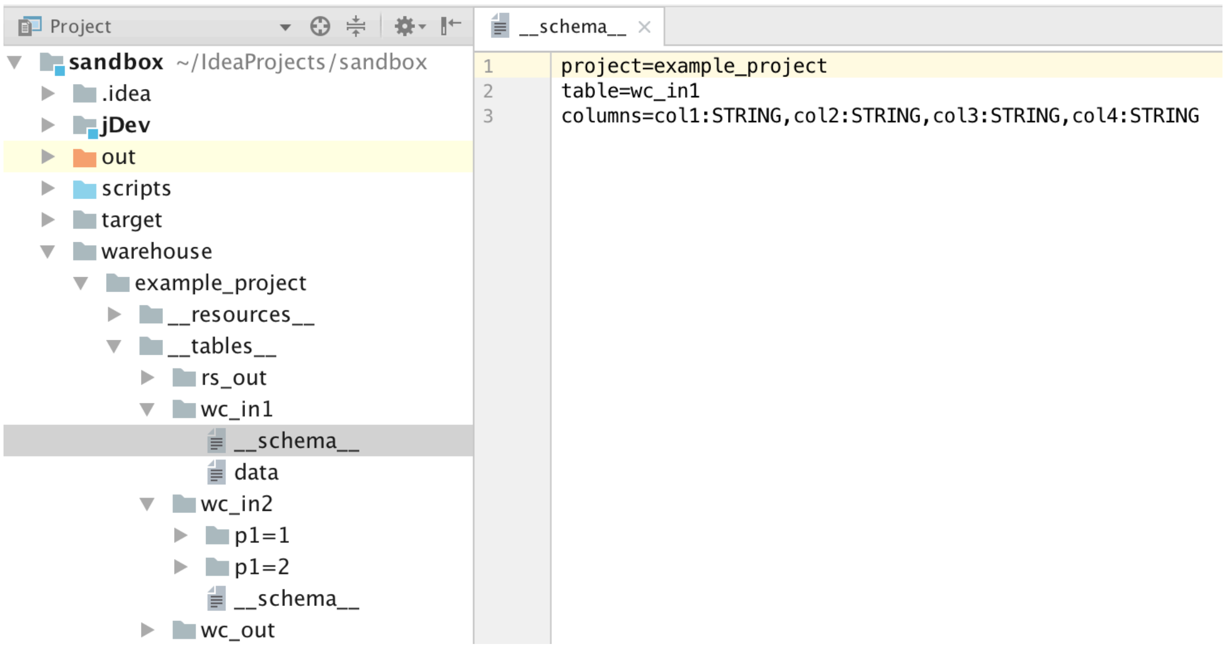
```
/Library/Java/JavaVirtualMachines/jdk1.8.0_65.jdk/Contents/Home/bin/java ...
Testing started at 15:02 ...
[INFO]Start to write table scheme : meta_dev.hy_test-->/Users/liuyi/IdeaProjects,
 [INFO]generate schema file: /Users/liuyi/IdeaProjects/studioTest/warehouse/meta_
 [INFO]Finished to write table scheme : meta_dev.hy_test-->/Users/liuyi/IdeaProje
 [INFO]Start to download table: 'meta_dev.hy_test', download mode:AUTO
 [INFO]Tunnel DownloadSession ID is : 20160930150300d492b70a01854c05
 [INFO]Start to write table: meta_dev.hy_test-->/Users/liuyi/IdeaProjects/studio
 [INFO]Finished write table: meta_dev.hy_test-->/Users/liuyi/IdeaProjects/studio
 hello world2:1 a
hello world2:2 b
hello world2:3 c,d

Process finished with exit code 0
```

Note:

- If the table data under the specified project is not downloaded into glashourse , You need to download the data first, default download 100 entries. If more data is required, use the Tunnel Command of the console or table downloading function of Studio.
- If the mock project is used or the table data is downloaded, directly run the program.
- The UDF local run framework uses data in specific columns in warehouse as the UDF input and run the UDF program locally. You can view log output and result display on the console.

Local warehouse directory

The local warehouse directory is used to store tables (including meta and data) or resources for local UDF or MR running. The following figure shows the warehouse directory.



```
project=example_project
table=wc_in1
columns=col1:STRING,col2:STRING,col3:STRING,col4:STRING
```

Note:

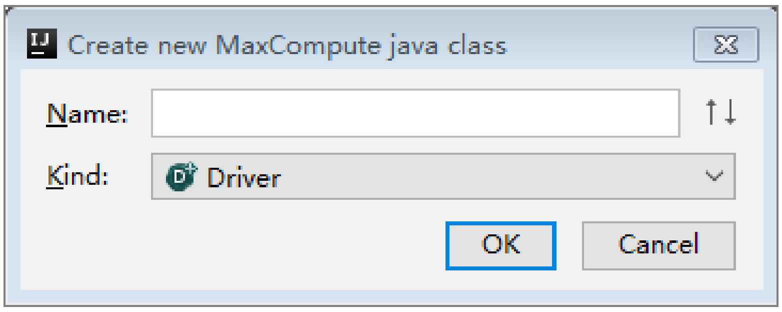
- The project name, tables, table name, table scheme, and sample data are under the warehouse directory in sequence.
- The schema file is configured with the project name, table name, and column name and type (separated using a colon) in sequence. For a partition table, the partition column also needs to be configured. (For a non-partition table, refer to wc_in1. For a partition table, refer to wc_in2).
- The data file uses the standard CSV format to store table sample data.
  - Special characters include comma, double quotation marks, and line feed (\n or \r\n).
  - The column separator is comma and the line separator is \n or \r\n.
  - If the column content includes special characters, double quotation marks (“) must be added before and after the column content. For example, if the column content is 3,No, it is changed to “3, No” .
  - If the column content includes double quotation marks, each double quotation mark is converted to two double quotation marks. For example, if the column content is a” b” c, it is changed to “a” ” b” ” c” .
  - \N indicates that a column is null. If the column content (string type) is \N, it must be converted to “” ” \N” ” ” .
  - The file character code is UTF-8.

2.6.3 Develop MapReduce

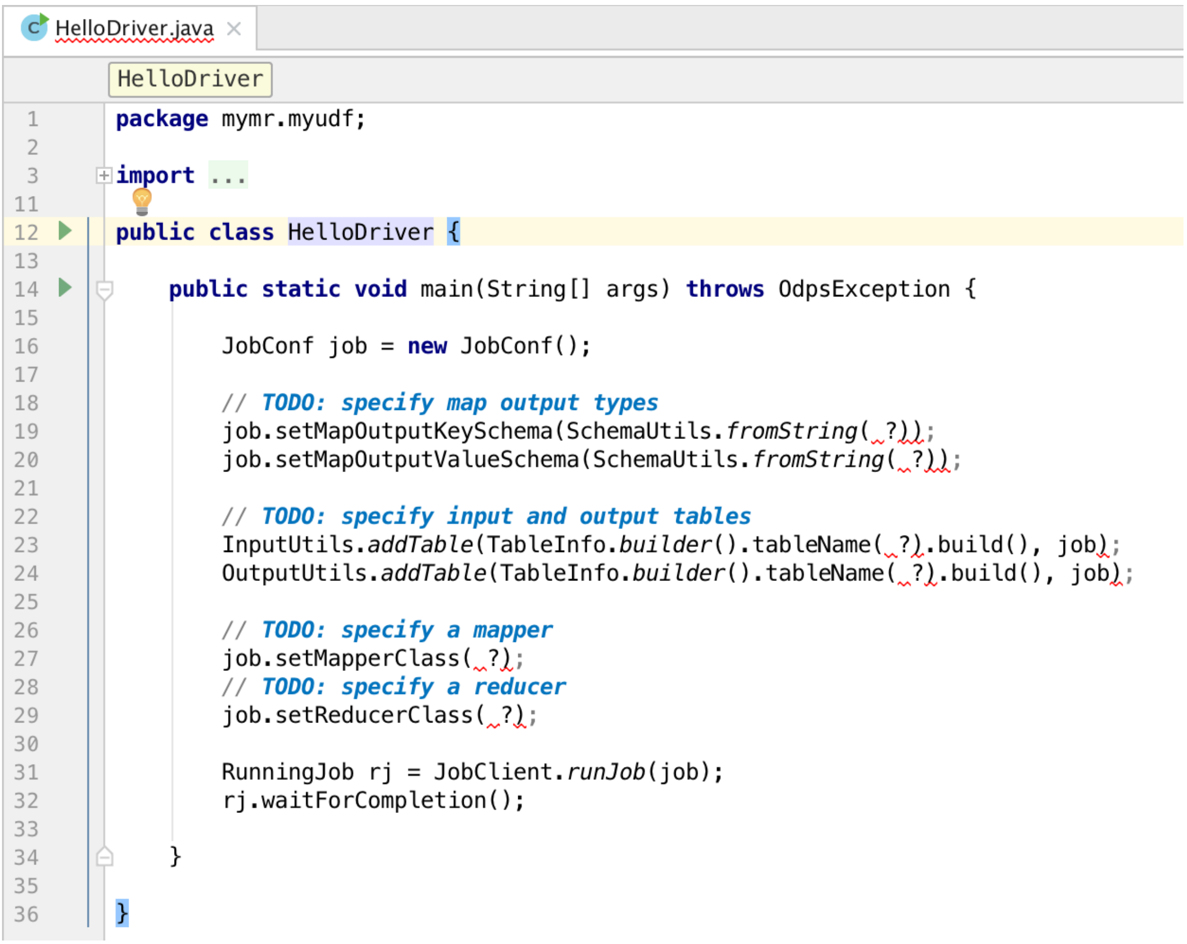
After the *Create MaxCompute Java Module* is created, *MR* can be developed.

Develop the MR program

1. Right-click the module source code directory src > main, select New > java, and select MaxCompute Java.
2. Create Driver, Mapper, and Reducer.



3. Set the input/output table and Mapper/Reducer class. The framework code is automatically filled in the template.



HelloDriver.java

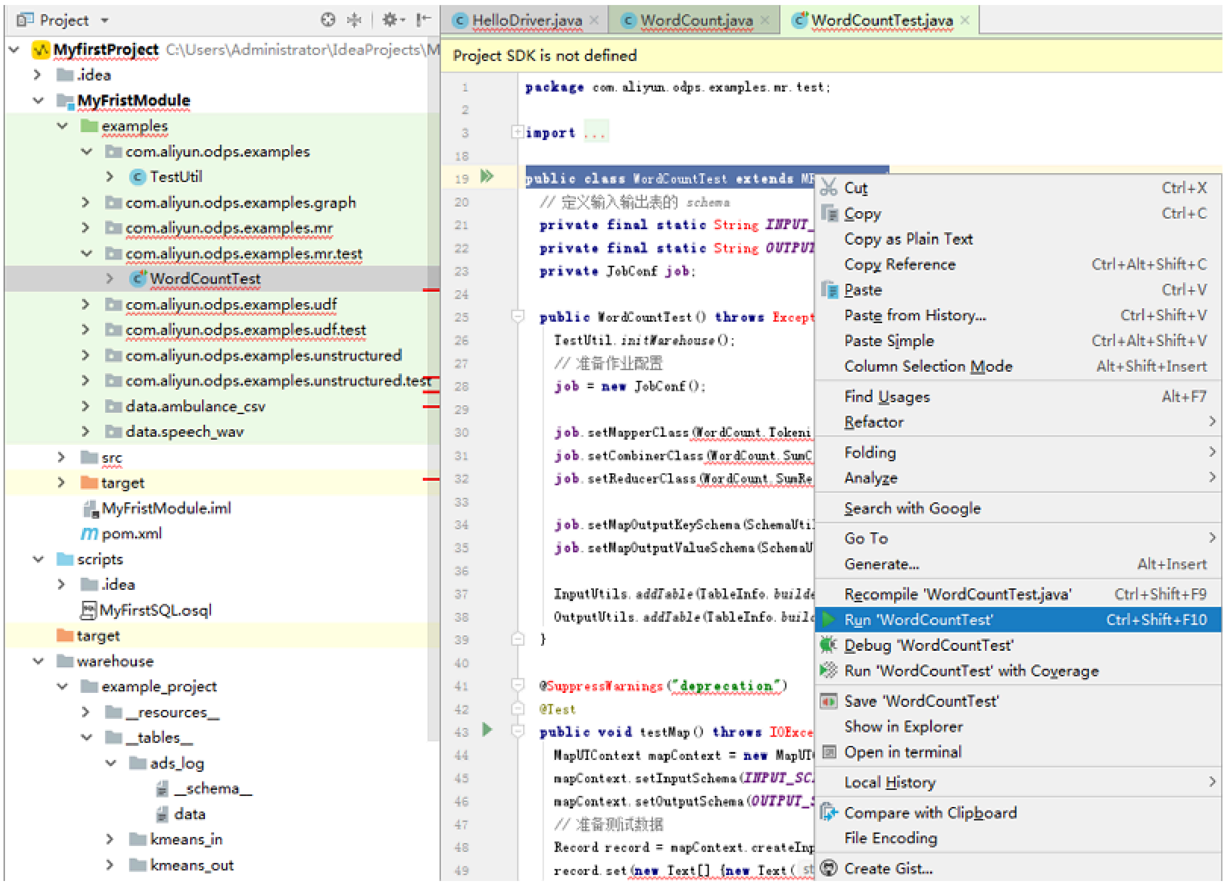
```
package mymr.myudf;

import ...

public class HelloDriver {

    public static void main(String[] args) throws OdpsException {

        JobConf job = new JobConf();

        // TODO: specify map output types
        job.setMapOutputKeySchema(SchemaUtils.fromString( ?));
        job.setMapOutputValueSchema(SchemaUtils.fromString( ?));

        // TODO: specify input and output tables
        InputUtils.addTable(TableInfo.builder().tableName( ?).build(), job);
        OutputUtils.addTable(TableInfo.builder().tableName( ?).build(), job);

        // TODO: specify a mapper
        job.setMapperClass( ?);
        // TODO: specify a reducer
        job.setReducerClass( ?);

        RunningJob rj = JobClient.runJob(job);
        rj.waitForCompletion();

    }

}
```

For details of developing MR, see *To write MapReduce*.

Debug the MR program

After the MR program is developed, test your code and check whether it meets the expectations. The following two methods are supported:

Unit test (UT): There are WordCount UT examples in the examples directory. You can refer to them to compile your UT.

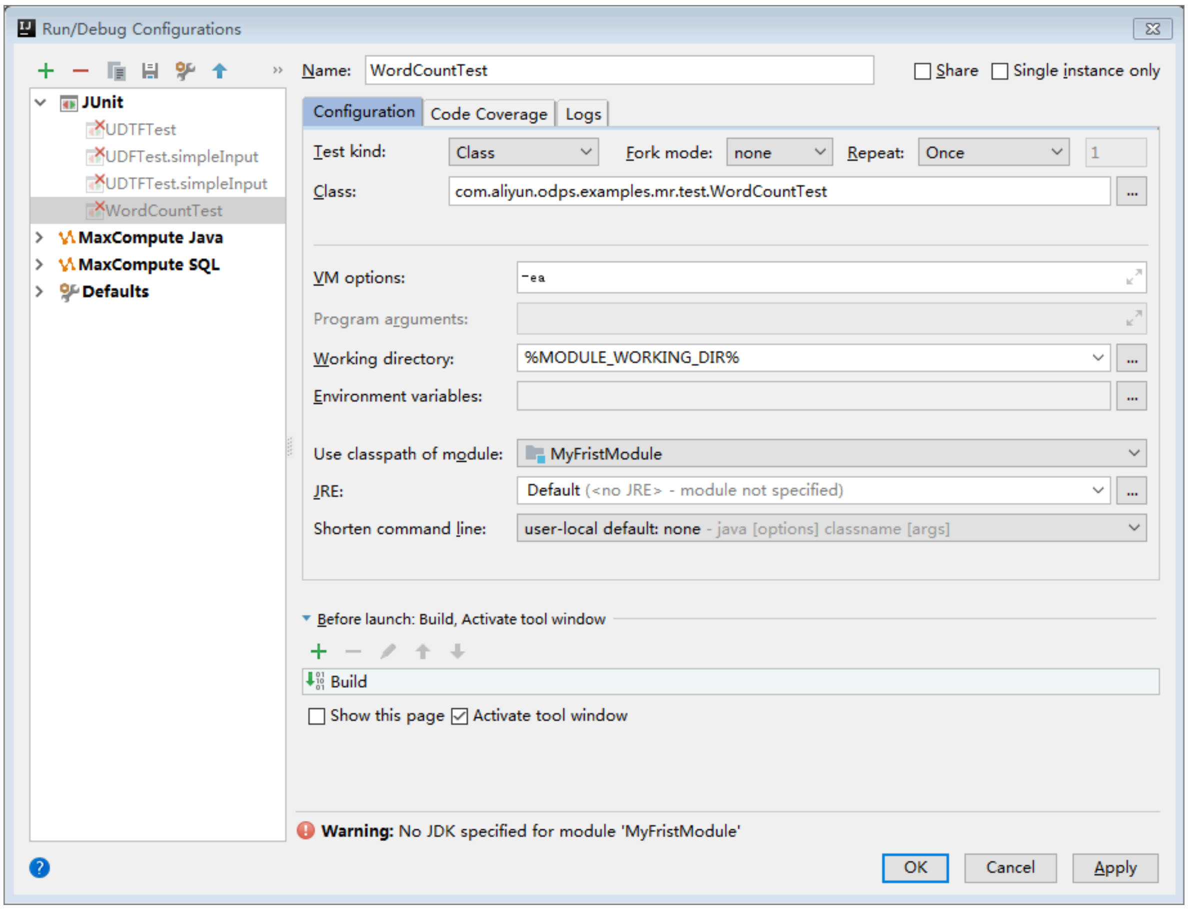


```
package com.aliyun.odps.examples.mr.test;

import ...

public class WordCountTest extends MR
    // 定义输入输出表的 schema
    private final static String INPUT_
    private final static String OUTPUT
    private JobConf job;

    public WordCountTest() throws Except
      TestUtil.initWarehouse();
      // 准备作业配置
      job = new JobConf();

      job.setMapperClass(WordCount.Tokeni
      job.setCombinerClass(WordCount.SumC
      job.setReducerClass(WordCount.SumRe

      job.setMapOutputKeySchema(SchemaUti
      job.setMapOutputValueSchema(SchemaU

      InputUtils.addTable(TableInfo.builde
      OutputUtils.addTable(TableInfo.buil
    }

    @SuppressWarnings("deprecation")
    @Test
    public void testMap() throws IOExce
      MapUTContext mapContext = new MapUT
      mapContext.setInputSchema(INPUT_SC
      mapContext.setOutputSchema(OUTPUT_
      // 准备测试数据
      Record record = mapContext.createInp
      record.set(new Text[] {new Text( st
```

Local MR running: During local running, the running data source must be specified. The following two methods are provided to set the test data source:

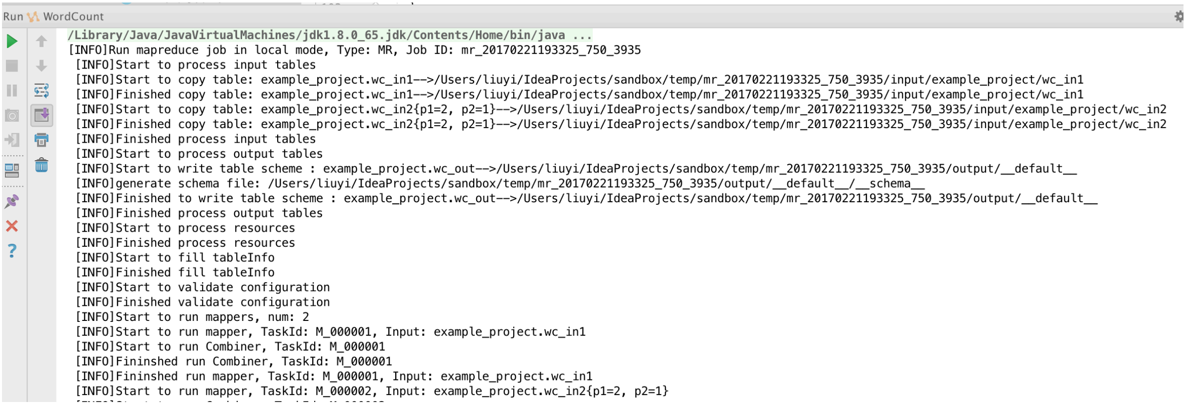
- MaxCompute Studio uses the Tunnel Service to automatically download table data of a specific MaxCompute project to the warehouse directory. By default, 100 data records are downloaded. If more data is required for testing, use the Tunnel Command of the console or table downloading function of MaxCompute Studio.
- Provide the mock project (example_project) and table data. You can see example_project in warehouse to set it by yourself.

1. Run the MR program. Right-click the Driver class and select Run. In the displayed Run Configuration dialog box, configure the MaxCompute project on which the MR program runs.



Name: WordCountTest
Test kind: Class
Fork mode: none
Repeat: Once
Class: com.aliyun.odps.examples.mr.test.WordCountTest
VM options: -ea
Working directory: %MODULE_WORKING_DIR%
Use classpath of module: MyFristModule
JRE: Default (<no JRE> - module not specified)
Shorten command line: user-local default: none - java [options] classname [args]
Before launch: Build, Activate tool window
Build
Show this page ☐ Activate tool window ☑
Warning: No JDK specified for module 'MyFristModule'

2. Click OK. If table data of the specified MaxCompute project is not downloaded to warehouse, download data first. If a mock project is used or the MaxCompute project table data is downloaded, skip this step. Then, the MR local run framework reads specified table data in warehouse as the MR input and runs the MR program locally. You can view log output and result display on the console.



```
/Library/Java/JavaVirtualMachines/jdk1.8.0_65.jdk/Contents/Home/bin/java ...
[INFO]Run mapreduce job in local mode, Type: MR, Job ID: mr_20170221193325_750_3935
 [INFO]Start to process input tables
 [INFO]Start to copy table: example_project.wc_in1-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/input/example_project/wc_in1
 [INFO]Finished copy table: example_project.wc_in1-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/input/example_project/wc_in1
 [INFO]Start to copy table: example_project.wc_in2{p1=2, p2=1}-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/input/example_project/wc_in2
 [INFO]Finished copy table: example_project.wc_in2{p1=2, p2=1}-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/input/example_project/wc_in2
 [INFO]Finished process input tables
 [INFO]Start to process output tables
 [INFO]Start to write table scheme : example_project.wc_out-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/output/__default__
 [INFO]generate schema file: /Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/output/__default__/__schema__
 [INFO]Finished to write table scheme : example_project.wc_out-->/Users/liuyi/IdeaProjects/sandbox/temp/mr_20170221193325_750_3935/output/__default__
 [INFO]Finished process output tables
 [INFO]Start to process resources
 [INFO]Finished process resources
 [INFO]Start to fill tableInfo
 [INFO]Finished fill tableInfo
 [INFO]Start to validate configuration
 [INFO]Finished validate configuration
 [INFO]Start to run mappers, num: 2
 [INFO]Start to run mapper, TaskId: M_000001, Input: example_project.wc_in1
 [INFO]Start to run Combiner, TaskId: M_000001
 [INFO]Fininshed run Combiner, TaskId: M_000001
 [INFO]Fininshed run mapper, TaskId: M_000001, Input: example_project.wc_in1
 [INFO]Start to run mapper, TaskId: M_000002, Input: example_project.wc_in2{p1=2, p2=1}
```

Run the MR program in the production environment

After local debugging is complete, release the MR program to the server and run it in the MaxCompute distributed environment.

1. Package the MR program to a JAR package and release it to the server. For more information, see *Package, Upload, and Register*.
2. Use the MaxCompute console integrated with MaxCompute Studio in seamless mode, that is, in the Project Explorer window, right-click Project and select Open in Console, and input the commands similar to the following *JAR command* in the console command line:

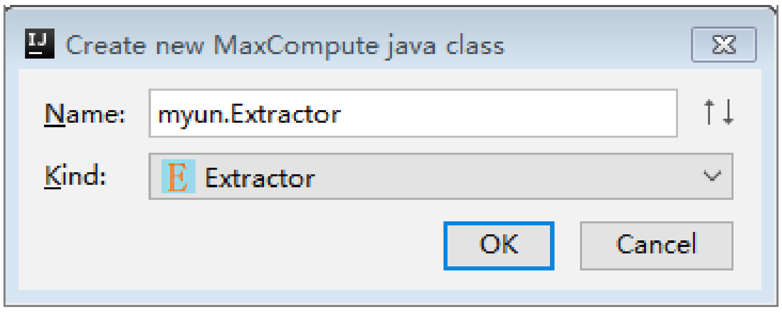
```
jar  - libjars  wordcount . jar  - classpath  D :\ odps \ clt \
wordcount . jar  com . aliyun . odps . examples . mr . WordCount
wc_in  wc_out ;
```

2.6.4 Unstructured development

An *unstructured data processing framework* is added for MaxCompute 2.0, supporting access to the OSS and Table Store using external tables. Studio provides some code templates for the framework, facilitating users' fast development.

Compile StorageHandler/Extractor/Outputter

1. After the *MaxCompute Java Module* (Sample code is provided in the unstructured folder of the examples directory for your reference).
2. Right-click the module source code directory src > > main, select new, and select MaxCompute Java.
3. Specify Name and Kind. For example, set Name to myun.MyExtractor and Kind to Extractor. Click OK.



Create new MaxCompute java class

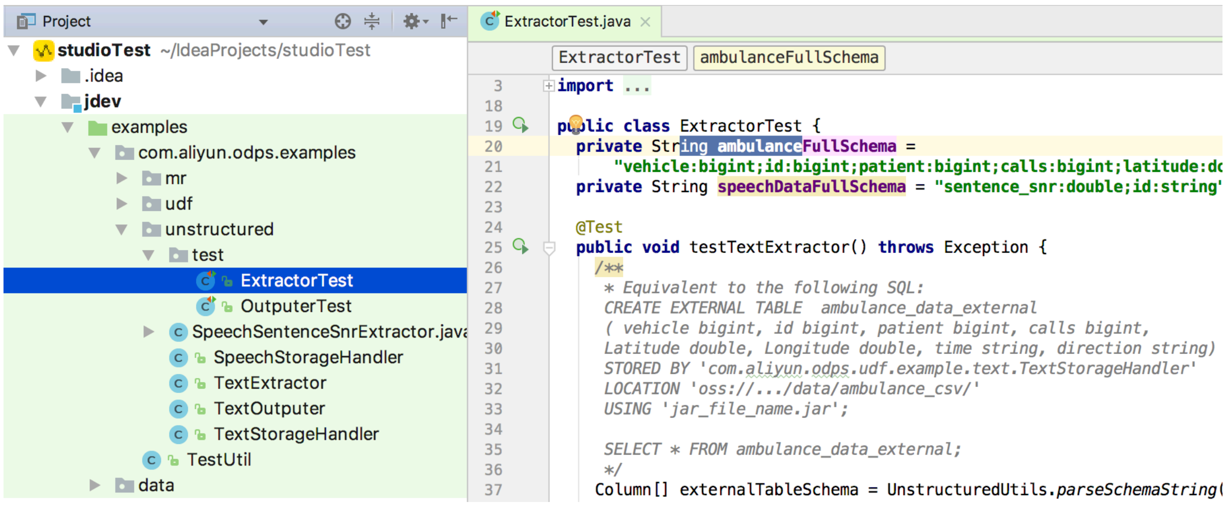
Name: myun.Extractor

Kind: Extractor

4. The framework code has been automatically filled in the template. Compile your logic code.
5. Compile Outputter and StorageHandler by following the preceding steps.

Unit Testing

You can compile the unit test (UT) by following the examples in the examples directory to test your Extractor/Outputter.



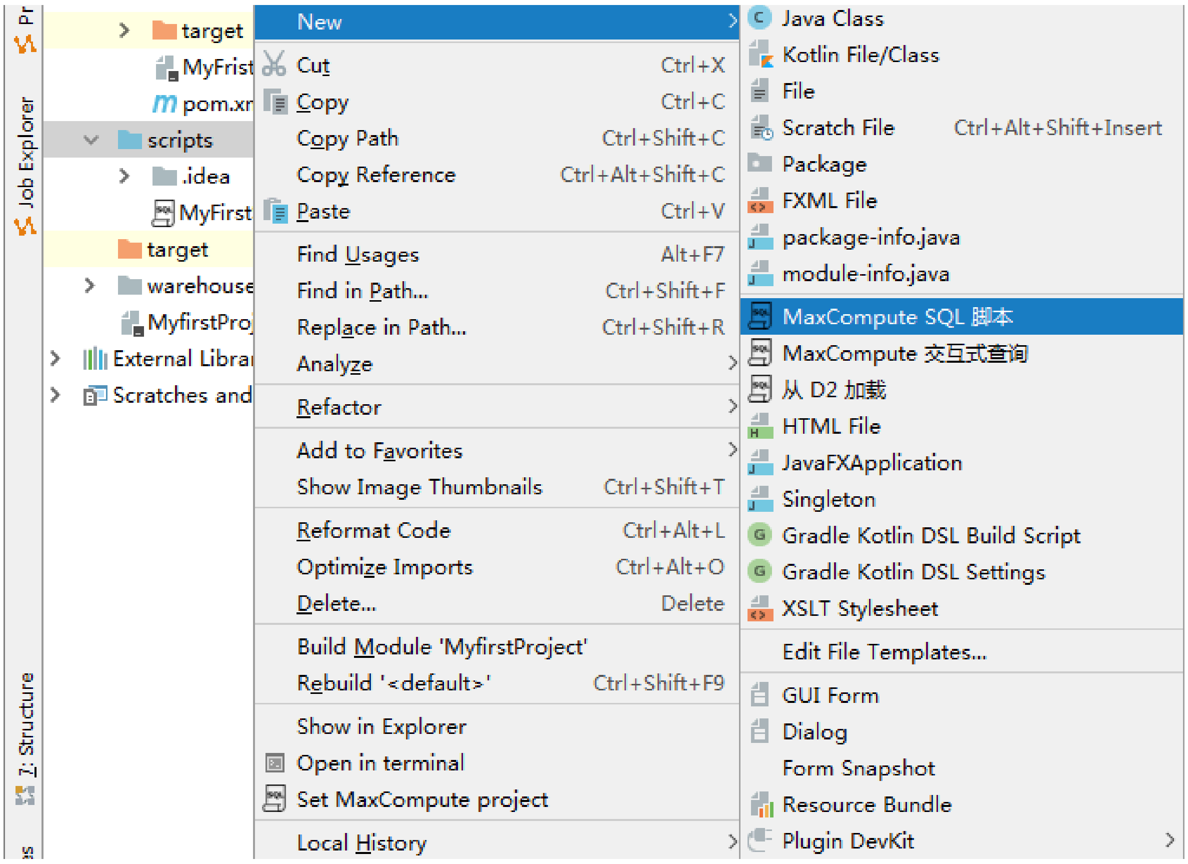
```
import ...

public class ExtractorTest {
  private String ambulanceFullSchema =
      "vehicle:bigint;id:bigint;patient:bigint;calls:bigint;latitude:d
  private String speechDataFullSchema = "sentence_snr:double;id:string'

  @Test
  public void testTextExtractor() throws Exception {
    /**
     * Equivalent to the following SQL:
     CREATE EXTERNAL TABLE  ambulance_data_external
     ( vehicle bigint, id bigint, patient bigint, calls bigint,
     Latitude double, Longitude double, time string, direction string)
     STORED BY 'com.aliyun.odps.udf.example.text.TextStorageHandler'
     LOCATION 'oss://.../data/ambulance_csv/'
     USING 'jar_file_name.jar';

     SELECT * FROM ambulance_data_external;
     */
    Column[] externalTableSchema = UnstructuredUtils.parseSchemaString(
```

Package and upload

After StorageHandler/Extractor/Outputter is compiled, compress the completed Java program to a JAR package, and upload the package as a resource to the server, see *Package and release*.

Create External Table

1. Right-click scripts and select new > MaxCompute Script.



2. Enter the SQL script name. Select the MaxCompute project in which the script is to be executed for Target Project and click OK.
3. Select create external table live template in the editor to rapidly insert the script template for creating an external table.

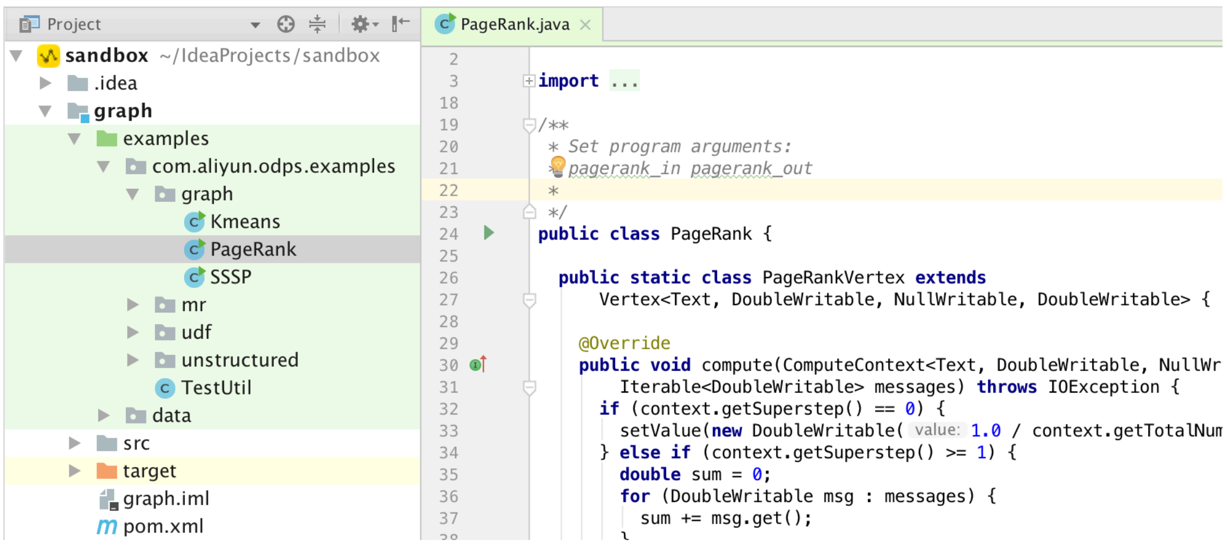
   Modify the external table name, column, type, StorageHanlder class path, configuration parameter, external path, and JAR name. Click Run MaxCompute SQL Script to create the external table.
4. Query the created external table.

2.6.5 Develop Graph

After the *MaxCompute Java module* is created, *Overview of Graph models* can be developed.

Sample Code

There are some code examples of Graph in the examples directory, and you can refer to the example to get familiar with the structure of the Graph program.

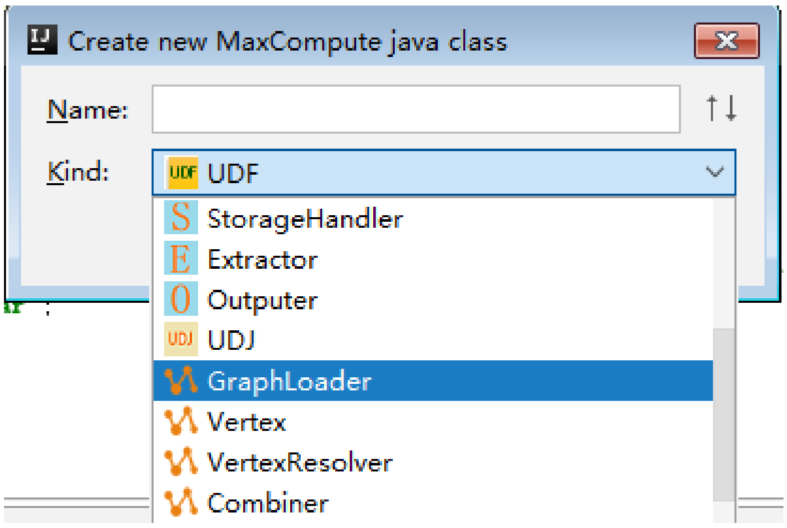


```
Project
sandbox ~/IdeaProjects/sandbox
  .idea
  graph
    examples
      com.aliyun.odps.examples
        graph
          Kmeans
          PageRank
          SSSP
        mr
        udf
        unstructured
        TestUtil
      data
    src
    target
    graph.iml
    pom.xml
```

```
PageRank.java

import ...

/**
 * Set program arguments:
 * pagerank_in pagerank_out
 *
 */
public class PageRank {

  public static class PageRankVertex extends
      Vertex<Text, DoubleWritable, NullWritable, DoubleWritable> {

    @Override
    public void compute(ComputeContext<Text, DoubleWritable, NullWr
        Iterable<DoubleWritable> messages) throws IOException {
      if (context.getSuperstep() == 0) {
        setValue(new DoubleWritable( value: 1.0 / context.getTotalNum
      } else if (context.getSuperstep() >= 1) {
        double sum = 0;
        for (DoubleWritable msg : messages) {
          sum += msg.get();
```

Develop a Graph Program

1. Right-click the module source code directory src > > main, select new, and select MaxCompute Java.
2. Select the GraphLoader/Vertex type and enter the class name (package name is supported) in the Name text box. Click OK, and the frame code will be automatically filled in by the template, you can continue to modify.



Create new MaxCompute java class

Name:

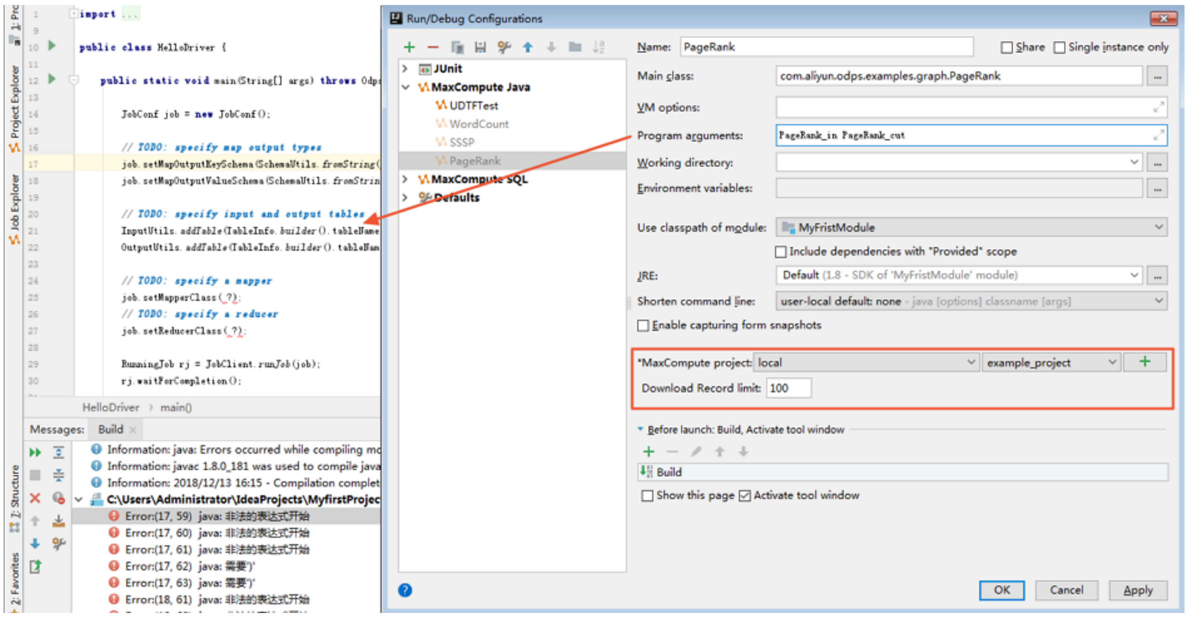
Kind: UDF

- StorageHandler
- Extractor
- Outputer
- UDJ
- GraphLoader
- Vertex
- VertexResolver
- Combiner

Debug Graph Locally

After the Graph program is developed, test your code and check whether it meets the expectations. You can run the Graph code locally.

1. Run the Graph program: Right-click the Driver class and select Run. In the displayed Run Configuration dialog box, configure the MaxCompute project on which the Graph program runs.



```
import ...

public class HelloDriver {

    public static void main(String[] args) throws Odps

        JobConf job = new JobConf();

        // TODO: specify map output types
        job.setMapOutputKeySchema(SchemaUtils.fromString(
        job.setMapOutputValueSchema(SchemaUtils.fromStrin

        // TODO: specify input and output tables
        InputUtils.addTable(TableInfo.builder().tableName
        OutputUtils.addTable(TableInfo.builder().tableNam

        // TODO: specify a mapper
        job.setMapperClass(?);
        // TODO: specify a reducer
        job.setReducerClass(?);

        RunningJob rj = JobClient.runJob(job);
        rj.waitForCompletion();
```

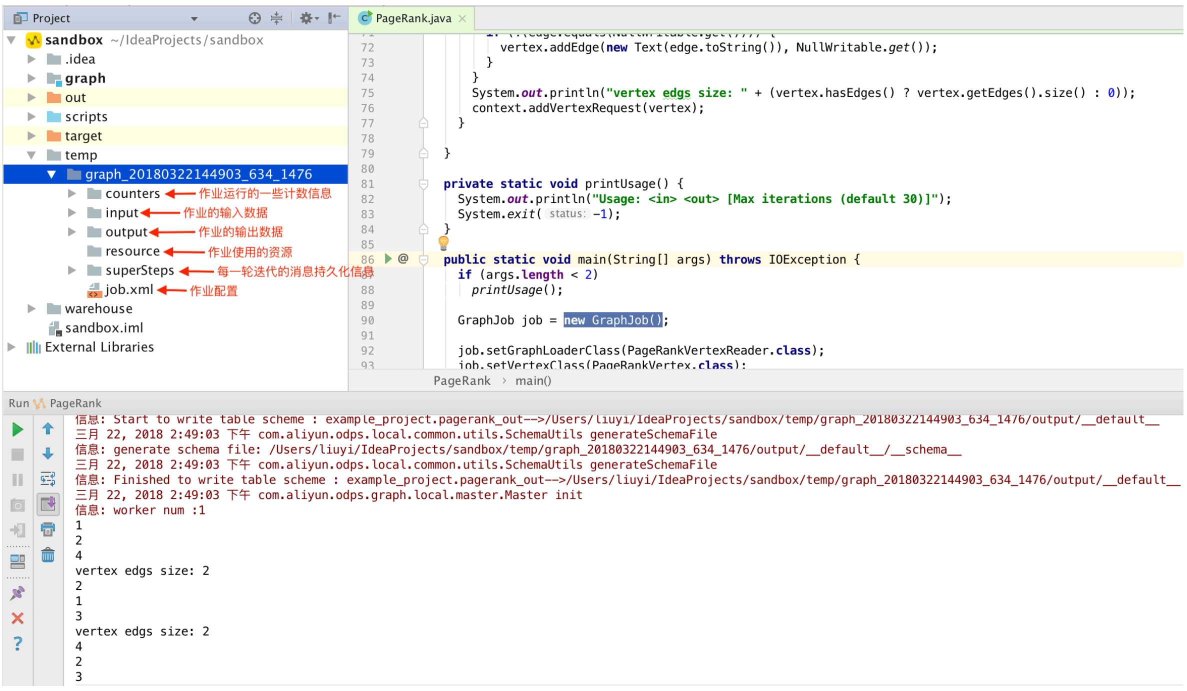
```
Information: java: Errors occurred while compiling mo
Information: javac 1.8.0_181 was used to compile java
Information: 2018/12/13 16:15 - Compilation complet
C:\Users\Administrator\IdeaProjects\MyfirstProjec
    Error:(17, 59)  java: 非法的表达式开始
    Error:(17, 60)  java: 非法的表达式开始
    Error:(17, 61)  java: 非法的表达式开始
    Error:(17, 62)  java: 需要')'
    Error:(17, 63)  java: 需要')'
    Error:(18, 61)  java: 非法的表达式开始
```

Run/Debug Configurations

Name: PageRank
Main class: com.aliyun.odps.examples.graph.PageRank
VM options:
Program arguments: PageRank_in PageRank_out
Working directory:
Environment variables:
Use classpath of module: MyFristModule
Include dependencies with "Provided" scope
JRE: Default (1.8 - SDK of 'MyFristModule' module)
Shorten command line: user-local default: none - java [options] classname [args]
Enable capturing form snapshots
*MaxCompute project: local / example_project
Download Record limit: 100
Before launch: Build, Activate tool window
Build
Show this page [✓ Activate tool window]

2. Click OK. If table data of the specified MaxCompute project is not downloaded to warehouse, download data first. If a mock project is used or the MaxCompute project table data is downloaded, skip this step. Then, the graph local run framework reads specified table data in warehouse as the Graph input and runs the Graph program locally. You can view log output and result display on the console.

Each time you debug locally， a new temporary directory is created under the Intellij directory, as shown in the following figure:



```java
            vertex.addEdge(new Text(edge.toString()), NullWritable.get());
          }
        }
        System.out.println("vertex edgs size: " + (vertex.hasEdges() ? vertex.getEdges().size() : 0));
        context.addVertexRequest(vertex);
      }

    }

    private static void printUsage() {
      System.out.println("Usage: <in> <out> [Max iterations (default 30)]");
      System.exit( status: -1);
    }

    public static void main(String[] args) throws IOException {
      if (args.length < 2)
        printUsage();

      GraphJob job = new GraphJob();

      job.setGraphLoaderClass(PageRankVertexReader.class);
      job.setVertexClass(PageRankVertex.class);
```

```
信息: Start to write table scheme : example_project.pagerank_out-->/Users/liuyi/IdeaProjects/sandbox/temp/graph_20180322144903_634_1476/output/__default__
三月 22, 2018 2:49:03 下午 com.aliyun.odps.local.common.utils.SchemaUtils generateSchemaFile
信息: generate schema file: /Users/liuyi/IdeaProjects/sandbox/temp/graph_20180322144903_634_1476/output/__default__/__schema__
三月 22, 2018 2:49:03 下午 com.aliyun.odps.local.common.utils.SchemaUtils generateSchemaFile
信息: Finished to write table scheme : example_project.pagerank_out-->/Users/liuyi/IdeaProjects/sandbox/temp/graph_20180322144903_634_1476/output/__default__
三月 22, 2018 2:49:03 下午 com.aliyun.odps.graph.local.master.Master init
信息: worker num :1
1
2
4
vertex edgs size: 2
2
1
3
vertex edgs size: 2
4
2
3
```

Note:

For a detailed introduction to warehouse, see Local Warehouse Directory section in *Develop UDF*.

Run the Graph Program in the Production Environment

After local debugging is complete, release the Graph program to the server and run it in the MaxCompute distributed environment.

1. Package the Graph program to a JAR package and release it to the server. For more information, please see *How to package and release Graph*.
2. Use the MaxCompute console integrated with MaxCompute Studio in seamless mode, that is, in the Project Explorer Window, right-click Project and select Open in Console, and input the commands similar to the following *JAR command* in the console command line:

```
jar  - libjars   xxx . jar  - classpath  / Users / home / xxx . jar
  com . aliyun . odps . graph . examples . PageRank   pagerank_i  n
  pagerank_o  ut ;
```

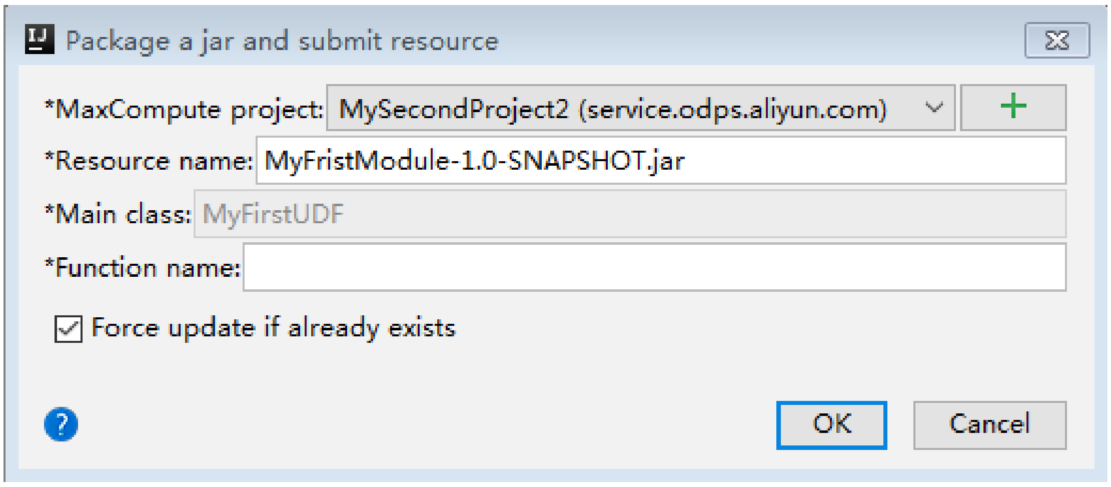
For more imformation about Graph development, please see *Graph*.

2.6.6 Package、Upload and Register

After a *user-defined function* or *MapReduce* is developed, you must package and release it to the MaxCompute system.

Package a UDF or MapReduce

To release a UDF or MapReduce to the MaxCompute server for production use, you must complete packaging > uploading > registration in sequence. You can use the one-click release function to complete these procedures. MaxCompute Studio runs the mvn clean package command, uploads a JAR package, and registers the UDF in one stop. To use this function, right-click the UDF or MapReduce and select Deploy to server···. Make sure that the target class is in the src > main > java subdirectory and is successfully compiled on the Maven module. The dialog box shown in the following figure appears. Select the MaxCompute project to be deployed and enter a resource name and a function name. Click OK and wait until the operation in the background is complete.


Package a jar and submit resource

*MaxCompute project: MySecondProject2 (service.odps.aliyun.com)
*Resource name: MyFristModule-1.0-SNAPSHOT.jar
*Main class: MyFirstUDF
*Function name:
[✓] Force update if already exists

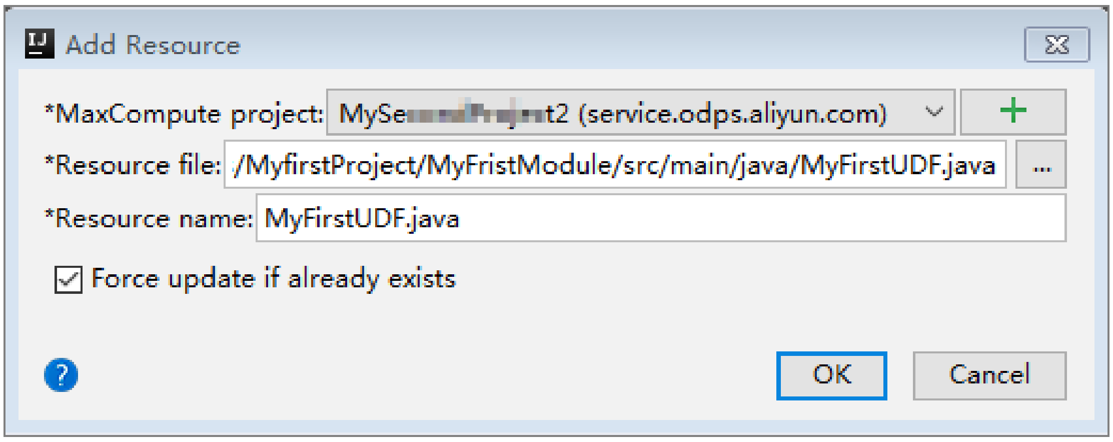
Note:
If you require special packaging, you can modify relevant settings in the pom.xml file. After packaging, follow these steps to upload the JAR package and register the UDF.

Upload the JAR package

After the JAR package is prepared, upload it to the MaxCompute server.

1. Select Add Resource from the MaxCompute menu.

2. Select the MaxCompute project you want to upload the resource to, the JAR file path, and the resource name you want to register. Determine whether to force update when the resource or function already exists. Then click OK.



Add Resource

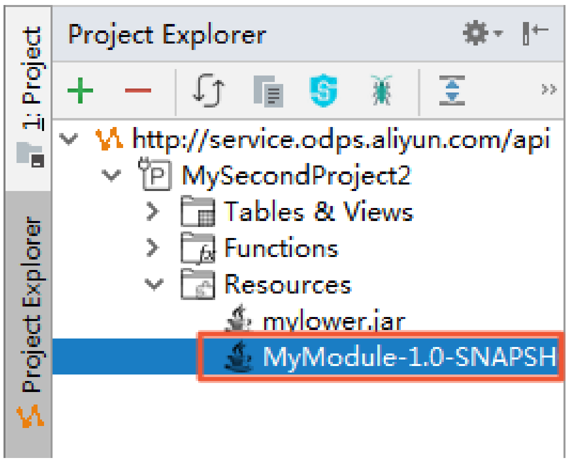
*MaxCompute project: MySe[illegible]t2 (service.odps.aliyun.com)

*Resource file: /MyfirstProject/MyFristModule/src/main/java/MyFirstUDF.java

*Resource name: MyFirstUDF.java

☑ Force update if already exists

3. After uploading is successful, you can view the resource under the Resources node of the Project Explorer window.



Project Explorer

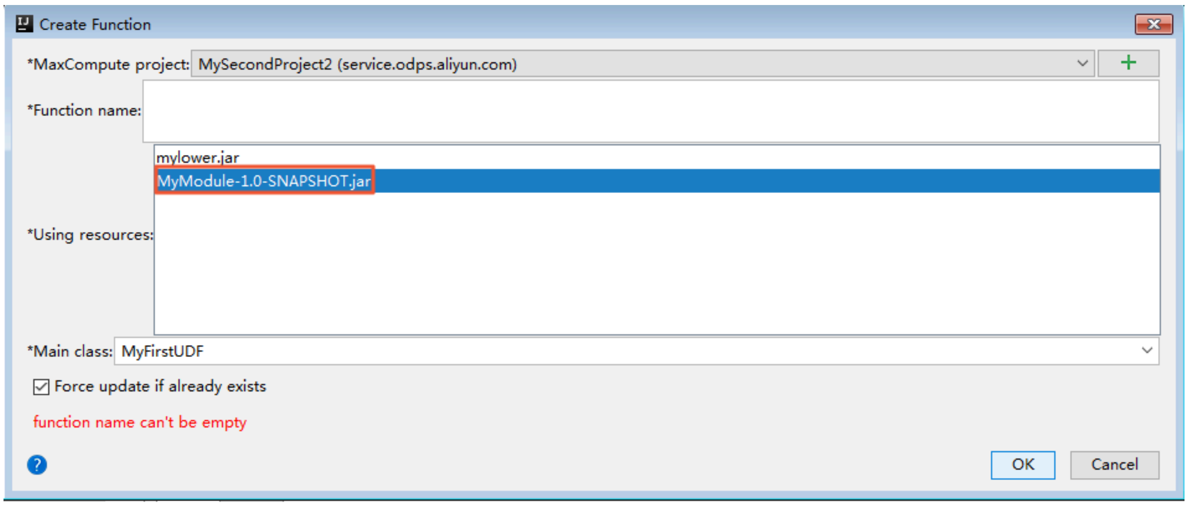
```
http://service.odps.aliyun.com/api
  MySecondProject2
    Tables & Views
    Functions
    Resources
      mylower.jar
      MyModule-1.0-SNAPSH
```

Register the UDF

After the JAR package is uploaded, register the UDF.

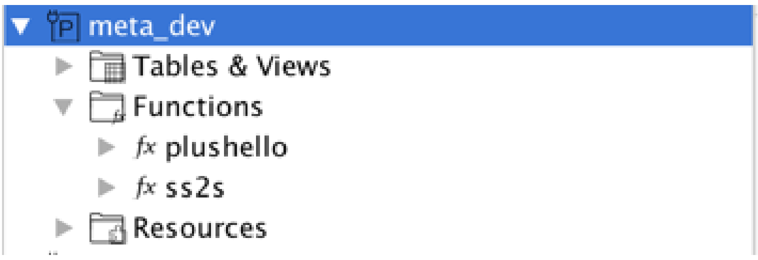
1. Select Create Function from the MaxCompute menu.

2. Select the required resource JAR and JAR main class, and enter the function name. Click OK.



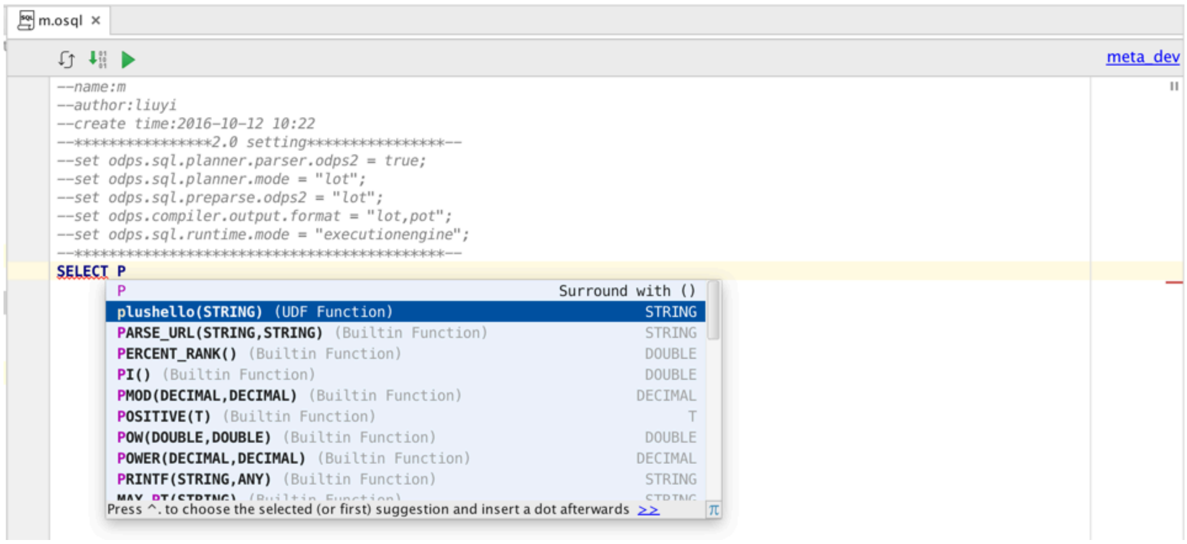
*MaxCompute project: MySecondProject2 (service.odps.aliyun.com)
*Function name:
*Using resources: mylower.jar, MyModule-1.0-SNAPSHOT.jar
*Main class: MyFirstUDF
☑ Force update if already exists
function name can't be empty

3. After the registration is successful, you can view the function under the Functions node of the Project Explorer window.



Apply the UDF

- Apply the UDF in SQL to complete subsequent development.



```
--name:m
--author:liuyi
--create time:2016-10-12 10:22
--****************2.0 setting****************--
--set odps.sql.planner.parser.odps2 = true;
--set odps.sql.planner.mode = "lot";
--set odps.sql.preparse.odps2 = "lot";
--set odps.compiler.output.format = "lot,pot";
--set odps.sql.runtime.mode = "executionengine";
--*******************************************--
SELECT P
```

```
P                                                Surround with ()
plushello(STRING) (UDF Function)                           STRING
PARSE_URL(STRING,STRING) (Builtin Function)                STRING
PERCENT_RANK() (Builtin Function)                          DOUBLE
PI() (Builtin Function)                                    DOUBLE
PMOD(DECIMAL,DECIMAL) (Builtin Function)                  DECIMAL
POSITIVE(T) (Builtin Function)                                  T
POW(DOUBLE,DOUBLE) (Builtin Function)                      DOUBLE
POWER(DECIMAL,DECIMAL) (Builtin Function)                 DECIMAL
PRINTF(STRING,ANY) (Builtin Function)                      STRING
Press ^. to choose the selected (or first) suggestion and insert a dot afterwards >>
```

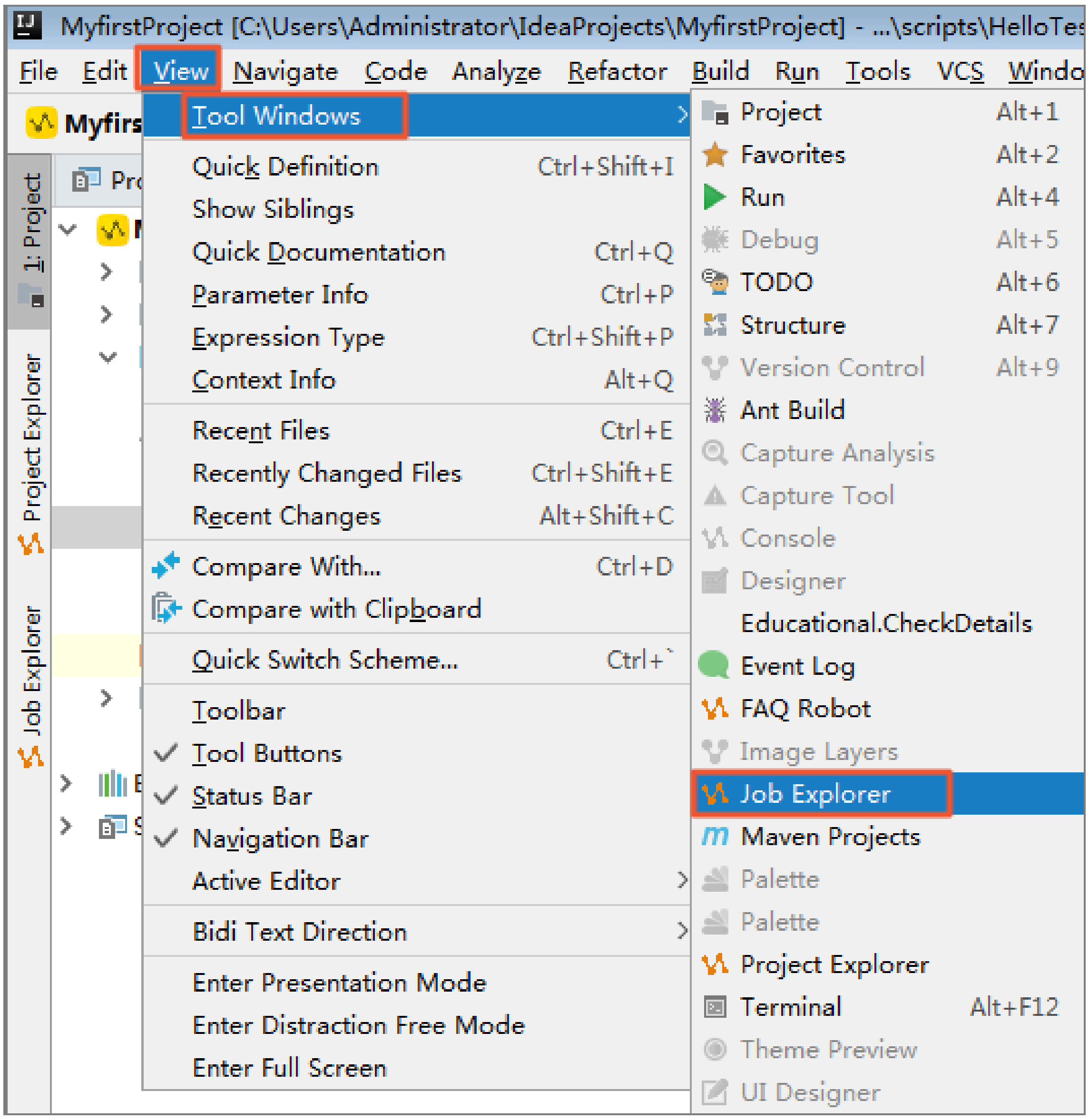
2.7 Manage MaxCompute jobs

2.7.1 Job viewing

MaxCompute Studio supports viewing information of MaxCompute running instances submitted by the current user, including the running status, job type, and start and stop time.

Open Job Explorer

If Job Explorer View is not displayed on Dock on the left, open Job Explorer by choosing View > > Tool > Windows > > MaxCompute Job Explorer.



View all job instances in a project

Job Explorer allows you to query submitted job lists by status.

Click the date drop-down box to select another date.

Click Refresh to obtain the job list.

Note:
By default, only the first 1,000 jobs that meet the conditions are displayed. If more than 1,000 jobs meet the conditions, update the filtering conditions.

Sort the job list

You can click the column name in the job list to sort the jobs.

Job queue

If a job in running status is waiting for scheduling in a queue, the job’ s location in the queue and global priority is displayed in the job list.

Note:
The job status and queue location on the Running Instances tab are automatically updated. After a job finishes, it is removed from the list.

Save job logs

Currently, Logview logs of a job are saved for seven days by default. If you want to save some important Logview logs for a longer period and view them in the future, you can save them locally.

Double-click a job in the list to display the job details on the right. Click Save on the toolbar to save the logs to your local host.

You can set the path for saving the log file on the Setting tab of MaxCompute Studio.

2.7.2 Job instance

View a job instance

Studio supports the following two ways to view MaxCompute job instances:

1. Open the details of a job in read-only mode using a Logview URL or local offline Logview file.

   Users of MaxCompute will be familiar with using Logview to view the details of a job. Using Logview, you can also view the status of tasks submitted by other users in other projects. You can also view the details of any job by entering a valid Logview URL in Studio.

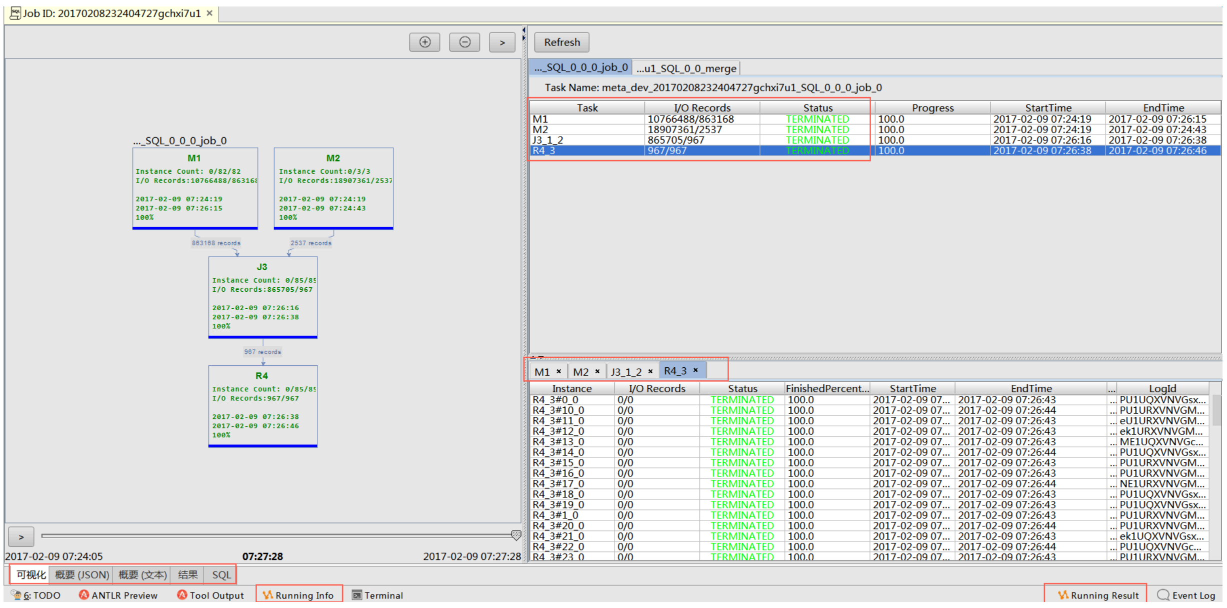
In the menu bar, select MaxCompute > Open Logview. Valid Logview URLs in the Clipboard are automatically copied to the dialog box displayed. Alternatively, you can select to export the local offline Logview file.

2. In *Job Explorer*, double-click a MaxCompute instance to view its details. You can also right-click the instance and select Open.

Job details view

The job details view page comprises a toolbar at the top, a properties bar on the left, and a detailed view on the right. The detailed view consists of seven views:

- Execution view: Displays the overall information of a job in the form of a DAG. You can view the dependencies and detailed execution plans for each subtask.
- Timeline view: Displays the execution timeline of a job, allows you to view this timeline in different granularities, and provides a number of filters.
- Details: Displays the details of a job in table view, including the subtask list, the worker list for each subtask, the volume of data processed by workers, the execution time, and the status.
- Script: Displays the corresponding SQL statement and parameter configuration for when a job is submitted.
- Summary (JSON): Displays the running details of a job in JSON format.
- Result: Displays the running results of a job.
- Analysis: Provides scatter plots, long tail distributions, and skewed data charts to show the results of a job execution.



Task Name: meta_dev_20170208232404727gchxi7u1_SQL_0_0_0_job_0

Task	I/O Records	Status	Progress	StartTime	EndTime
M1	10766488/863168	TERMINATED	100.0	2017-02-09 07:24:19	2017-02-09 07:26:15
M2	18907361/2537	TERMINATED	100.0	2017-02-09 07:24:19	2017-02-09 07:24:43
J3_1_2	865705/967	TERMINATED	100.0	2017-02-09 07:26:16	2017-02-09 07:26:38
R4_3	967/967	TERMINATED	100.0	2017-02-09 07:26:38	2017-02-09 07:26:46

Instance	I/O Records	Status	FinishedPercent...	StartTime	EndTime	...	LogId
R4_3#0_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1UQXVNVGsx...
R4_3#10_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:44	...	PU1URXVNVGM...
R4_3#11_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	eU1URXVNVGM...
R4_3#12_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	ek1URXVNVGM...
R4_3#13_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	ME1UQXVNVGc...
R4_3#14_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:44	...	PU1UQXVNVGsx...
R4_3#15_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1URXVNVGM...
R4_3#16_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1URXVNVGM...
R4_3#17_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:44	...	NE1URXVNVGM...
R4_3#18_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1UQXVNVGsx...
R4_3#19_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1UQXVNVGsx...
R4_3#1_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1URXVNVGM...
R4_3#20_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:44	...	PU1URXVNVGM...
R4_3#21_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	ek1UQXVNVGsx...
R4_3#22_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:44	...	PU1UQXVNVGc...
R4_3#23_0	0/0	TERMINATED	100.0	2017-02-09 07...	2017-02-09 07:26:43	...	PU1URXVNVGM...

Toolbar

Collapse pages on left/right

‹

›

Used to expand or collapse views on the left and right, allowing you to focus on a particular view.

Stop a job

Used to stop a job that is being executed. You need permission (owner or administrator) to do so.

Refresh details

The basic information of a running job, such as its status and quota, is refreshed automatically, but the detailed view on the right is not. If you want the most up-to-date details, you have to refresh them manually.

Copy Logview

Copies the corresponding Logview of a job to the Clipboard.

Open job details in a browser

Generates a Logview URL and opens it in a browser.

Store job details locally

Stores the job details as a local file.

Auto refresh

With this function enabled, Studio automatically refreshes all details of the executed job on a timed basis.

Basic information page

Displays the basic information of a job, including its ID, Owner, status, start and end time, computing resource usage, input, output, and so on. The basic information of a running job is automatically refreshed on a timed basis. The input and output items list the input and output tables of the job. Double-click a table name to view its details .

Execution view

Execution view is the most commonly used tool to display the dependency between Fuxi Job, Fuxi Task, and Operation. It also provides a series of auxiliary tools such as job playback, progress view, heat map view, and so on. The execution view is useful for troubleshooting problems.

The execution view can display job dependencies in three dimensions: the Fuxi Job layer, Fuxi Task layer, and Operation layer. Click the breadcrumb or the up arrow to switch between dimensions. By default, the dependency in the Fuxi Task layer is displayed.

Note:
For non-SQL jobs, only those in the Fuxi Job and Fuxi Task layers can be displayed. Operation layer jobs cannot be displayed.

- Fuxi Job layer:

  Click the breadcrumb or the up arrow to switch between dimensions. By default, the dependency in the Fuxi Task layer is displayed. Fuxi To open the Fuxi Job layer , click the MaxCompute Job in the breadcrumb or the up arrow in the Fuxi Task layer. Fuxi Job workers include the names, start times, and end times of Fuxi Tasks . To go to the Fuxi Task layer, double-click any Fuxi Job worker.

- Fuxi Task layer:

  In the event there is more than one Fuxi Job, the Fuxi Task layer of the last Fuxi Job is opened by default. This layer can display the dependency, input/output tables, and partitions of Fuxi Tasks. Once the job has ended, click the drop-down box in the toolbar to switch between views, including those for progress, input and output heat maps, and Task time and Instance heat maps. The progress view displays the progress of completion for the worker. The heat map view uses colors to distinguis h between different worker heats. Double-click any Fuxi Task to open its Operation layer. Right-click to open the Operation layers of all Fuxi Tasks.

  Contents of a Fuxi Task worker:

```
1 . Instance   Count :  a / b / c   indicates   that   at   a
  certain   point   in   time ,  the   number   of   running
subtask   instances   is   a ,  the   number   of   completed
task   instances   is   b ,  and   the   total   number   of   task
  instances   is   c .

2 . I / O   records :  Similarly ,  this   displays   the   number
  of   input   records   and   output   records   at   a   certain
  point   in   time .

3 . Percentage   and   orange   progress   bar :  Indicates   the
  running   status   of   the   task .  It   is   obtained   by
analyzing   the   running   subtask   instances .

4 . The   line   connecting   subtasks   shows   the   number   of
  output   records .  The   arrow   indicates   the   data   flow
  direction .
```

- Operation layer

  The Operation layer reveals how Fuxi Tasks run internally. By clicking a worker, all Operation information is displayed.

Job playback

Studio supports the job playback function. The history of a job can be reviewed within 12s, just like playing a media file. This function helps you understand the running status of a MaxCompute instance in different time points, rapidly determine the sub-task-level running sequence and time consumed, master the key path for executing a job, and accordingly optimize sub-tasks that run slowly.

- Click > to start playing the job, and click > again to pause. You can also manually drag the progress bar.

- The start time of the job is displayed on the left side of the progress bar, the playing time in the middle, and the end time on the right.

> **Note:**
> The playback function only estimates the volume of I/O data at a certain point in time by measuring the time, thereby determining the progress of completion. This does not represent the actual volume of I/O data. Jobs with a running status do not support playback.

Timeline view

Displays in the form of a Gantt chart detailed data for the distributed execution of a job. You can adjust the display granularity to show all computing workers in a Gantt chart.

Gantt charts can be used to display the time bottlenecks and long-tail workers of running jobs. Multiple filters are also provided that help select the key paths, the largest data worker, and the longest time worker for job execution.

Job details page

Mainly applies to SQL DML jobs, displaying their Fuxi Task lists and computing worker lists on the compute cluster.

One job typically corresponds to one or more Fuxi Jobs. Each Fuxi Job is divided into multiple Fuxi Tasks (stages), and each Fuxi Task comprises multiple Fuxi instances (workers).

Right-click a Fuxi instance, and the displayed menu allows you to view the standard output, standard errors, and Debug Info.

Analysis page

Displays the long-tail workers and skewed data workers of jobs. Displays worker scatter plots and column charts to help diagnose job execution bottlenecks.

Scatter plots and column charts allow you to call the details view page from their workers and check Fuxi instance details.

Results page

The results page displays different pages based on the job type and the parameters configured at the time the job was submitted.

- Select statement and set odps.sql.select.output.format = HumanReadable. This displays the result in text format
- Select statement but do not set the output format. This displays the result in table format
- For scripts where data is exported to the table, the output table name and the link that redirects to the table details are displayed
- For abnormal jobs, the results page limits the details on abnormalities

2.8 Tool integration

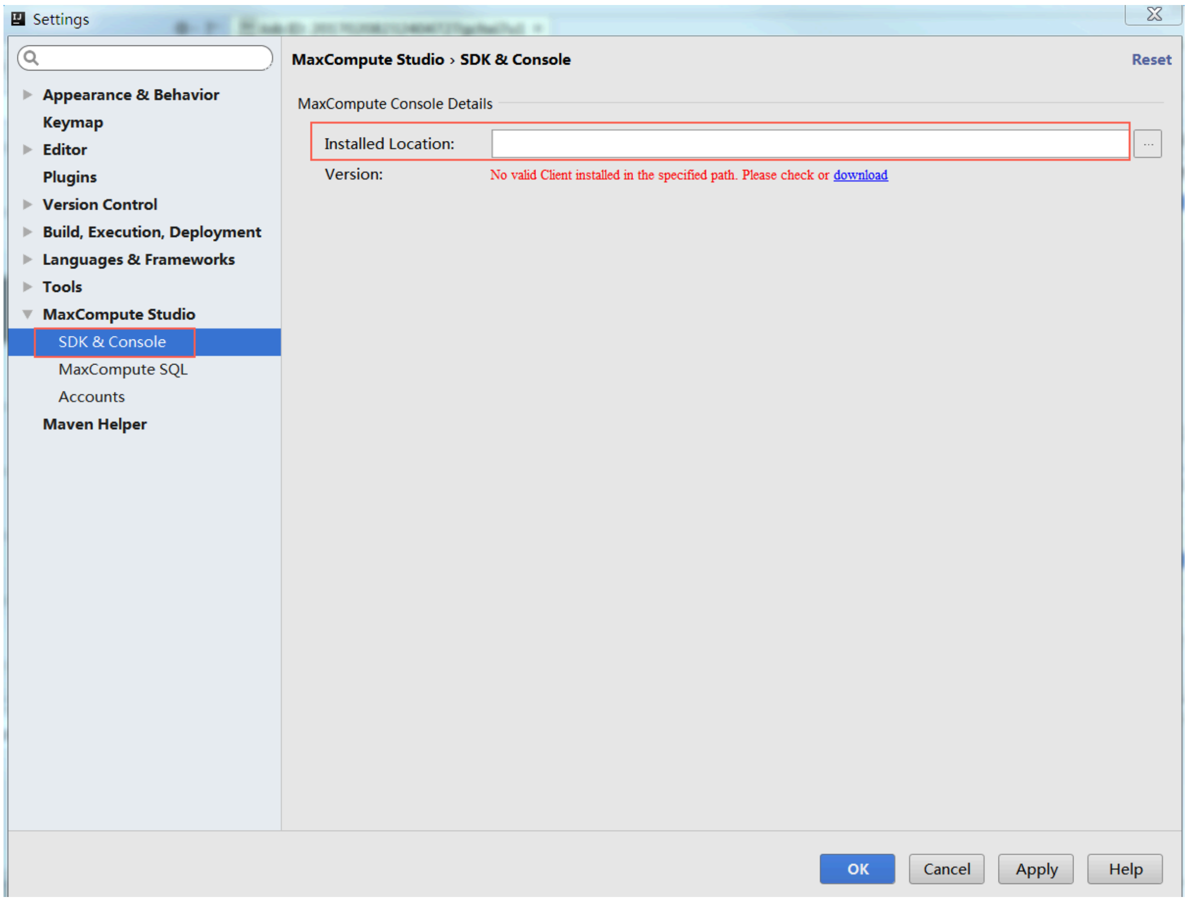
2.8.1 Integrate with MaxCompute client

MaxCompute Studio is integrated with the MaxCompute client program. You can open the client on MaxCompute Studio.

Configure the client installation path

1. MaxCompute Studio contains the MaxCompute client of the latest version, which is specified as the default client. You can also install the client of another version

by selecting Settings > MaxCompute Studio > SDK & Console on IntelliJ IDEA and adding the client program and path. Console download address

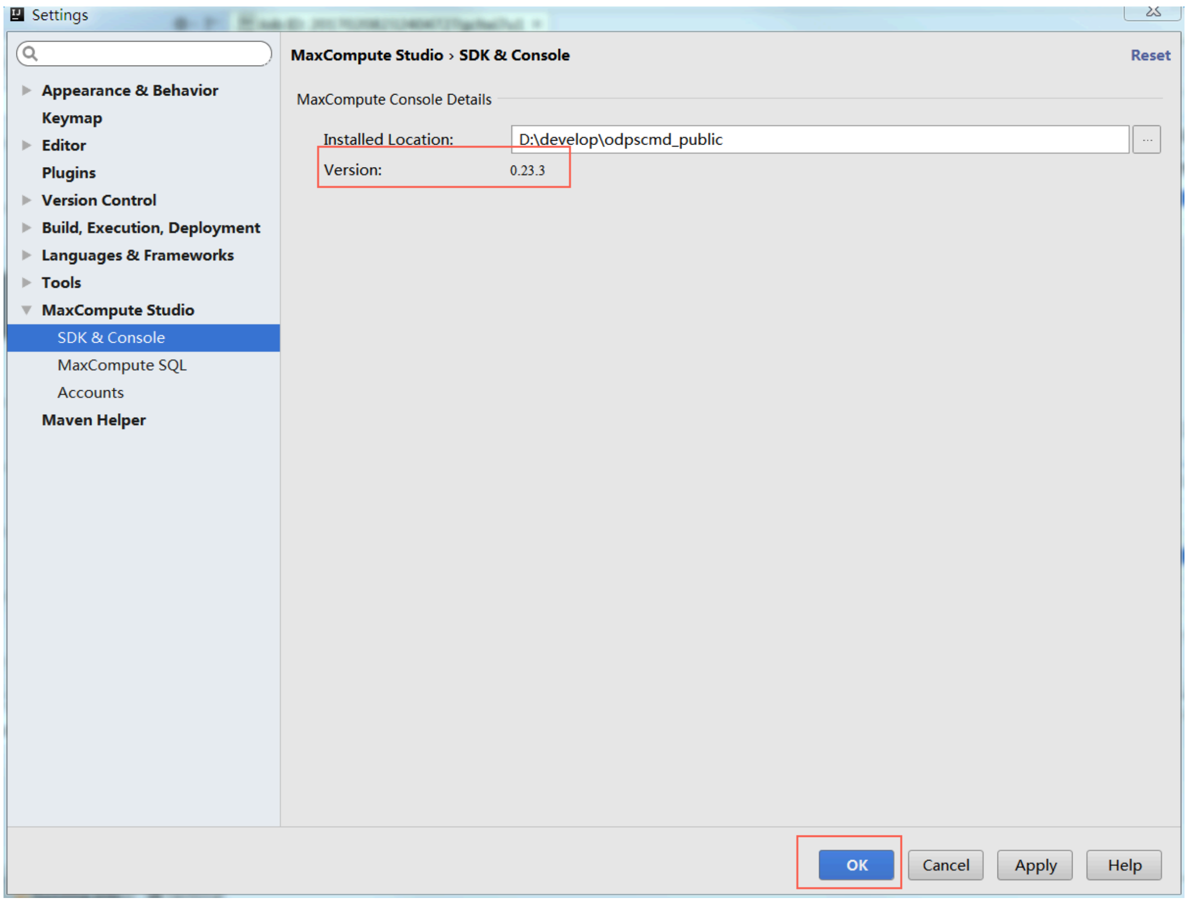


MaxCompute Console Details

Installed Location:

Version: No valid Client installed in the specified path. Please check or download

2. After setting is successful, the MaxCompute client version is displayed.



MaxCompute Console Details

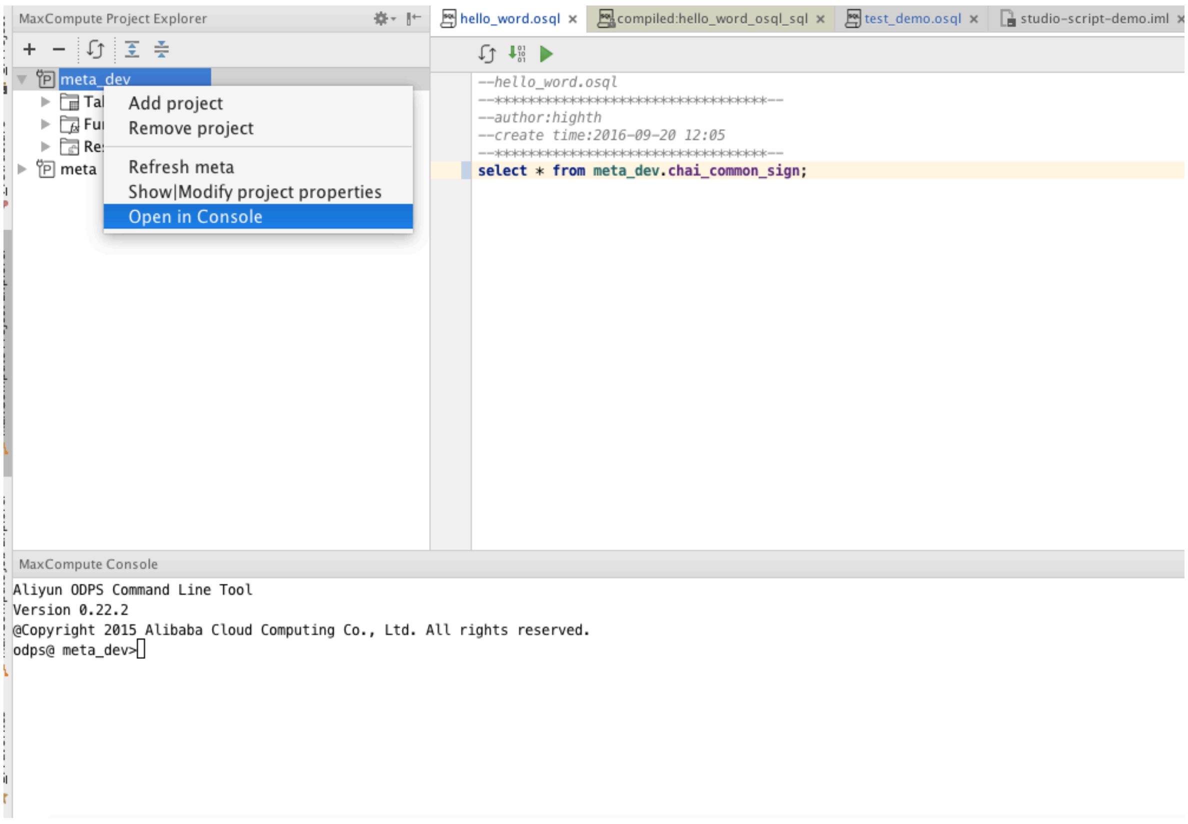
Installed Location: D:\develop\odpscmd_public

Version: 0.23.3

Open the MaxCompute client

After the MaxCompute client installation path is set, you can open the client program on MaxCompute Studio.

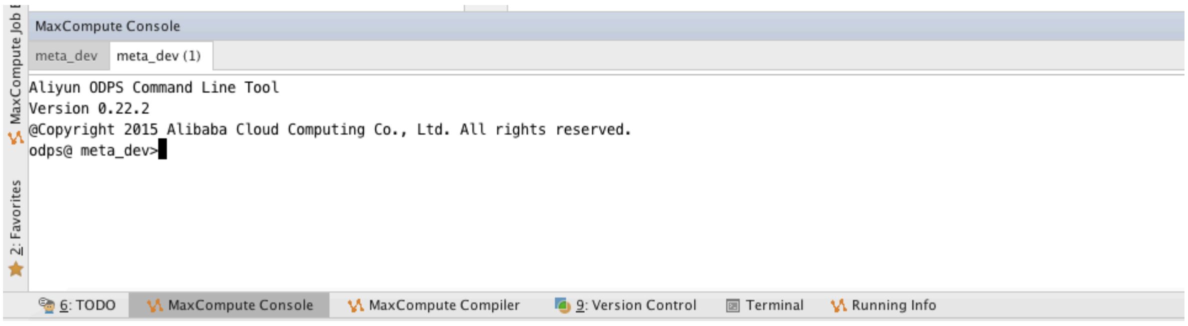
1. In the project browsing list, right-click a project to be opened and select Open in Console.



```
--hello_word.osql
--**********************************--
--author:highth
--create time:2016-09-20 12:05
--**********************************--
select * from meta_dev.chai_common_sign;
```

```
Aliyun ODPS Command Line Tool
Version 0.22.2
@Copyright 2015 Alibaba Cloud Computing Co., Ltd. All rights reserved.
odps@ meta_dev>
```

2. You can open multiple client programs by following the preceding steps.



```
Aliyun ODPS Command Line Tool
Version 0.22.2
@Copyright 2015 Alibaba Cloud Computing Co., Ltd. All rights reserved.
odps@ meta_dev>
```

2.9 Configure options

2.9.1 Configure MaxCompute Studio

After the MaxCompute Studio plug-in is installed, you can find configuration items of MaxCompute Studio on the left bar of the Settings page of IntelliJ IDEA. For more information about how to open the IntelliJ IDEA configuration page, see *IntelliJ IDEA Documentation*.

MaxCompute Studio configuration option page

The MaxCompute Studio configuration option page provides the following configurat ion items:

1. Path for storing the local metadata base

   Specifies the path for locally storing metadata of a MaxCompute project. On MaxCompute Studio, the metadata is stored in the hidden directory `. odps . studio / meta` of the local user directory by default.
2. Version update options

   - You can use the Automatically checks for new version check box to control whether MaxCompute Studio automatically checks for new version updates.
   - You can use the Check new versions button to manually check new versions. After you click this button, if a new version is available, the Install new version button is displayed. You can click this button to install the new version, and restart IntelliJ IDEA after the installation is complete.

SDK and Console configuration option page

The SDK and Console configuration option page provides the following configuration items:

1. Path for installing a MaxCompute client

   Specifies the path for local installation of MaxCompute client. MaxCompute Studio detects the version of the MaxCompute client installed in the path. If detection fails, an error message is prompted.

   Note:
   MaxCompute Studio later than the 2.6.1 version provides the latest MaxCompute client. You do not need to specify the path. If you must use a MaxCompute client of a specific version, you can specify the path.

MaxCompute SQL configuration option page

The MaxCompute SQL configuration option page provides the following configuration items:

1. Enable syntax coloring

   Select Enable syntax coloring to enable the syntax highlighting feature.
2. Enable code completion

   Select Enable code completion to enable the automatic code complementing feature.
3. Enable code formatting

   Select Enable code formatting to enable the code formatting feature.
4. Compiler options

   These are global default compiler options. The following options can be separately set for each file on the toolbar of the SQL compiler.

   - Compiler Mode
     - Statement Mode: In this mode, the compiler compiles and submits a single statement of an SQL file as a unit.
     - Script Mode: In this mode, the compiler compiles and submits an entire SQL file as a unit. *NOTE: Script Mode enables the compiler and optimizer to optimize the execution plan and improve the overall execution efficiency. This mode is in the test phase now.*
   - Type System
     - Legacy TypeSystem: Indicates the type system of original MaxCompute.
     - MaxCompute TypeSystem: Indicates the new type system introduced by MaxCompute 2.0.
     - Hive Compatible TypeSystem: Indicates the type system in Hive compatibility mode introduced by MaxCompute 2.0.
   - Compiler Version
     - Default Version: Indicates the default version of the compiler.
     - Flighting Version: Indicates the experimental version of the compiler, which includes new features of the compiler being tested.

Account configuration option page

You can add or manage accounts used to access MaxCompute on the Account configuration option page. For more information, see *User authentication*.

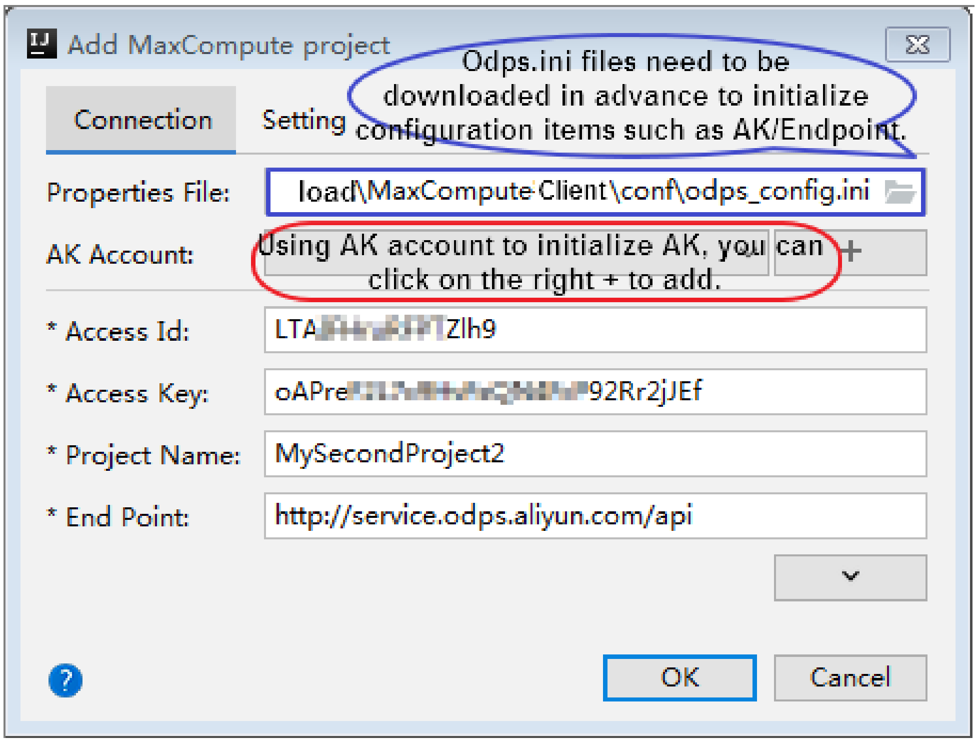
You must specify an account on MaxCompute Studio to access a MaxCompute project and run or submit jobs. MaxCompute Studio currently supports the following account type:

- Alibaba Cloud account (AccessKey)

Add an account

On the Account configuration option page, follow these steps:

1. Click + or press `Ctrl - N`.
2. Select the account type Aliyun Account by AccessKey.
3. In the displayed Add Account window, set the following items:
   - `Account Name`: Indicates the name of the account on MaxCompute Studio.
   - `Using properties file`: Read the AccessKey ID and AccessKey Secret from the configuration file.
     - Select the configuration file `conf/odps_config.ini` after you process *User authentication*.
   - `Using properties`: Manually enter the AccessKey ID and AccessKey Secret.
     - `Access Id`: Enter the AccessKey ID of your Alibaba Cloud account.
     - `Access Key`: Enter the AccessKey Secret of your Alibaba Cloud account.


Add MaxCompute project

Connection | Setting

Odps.ini files need to be downloaded in advance to initialize configuration items such as AK/Endpoint.

Properties File: load\MaxCompute Client\conf\odps_config.ini

AK Account: Using AK account to initialize AK, you can click on the right + to add.

* Access Id: LTA[redacted]Zlh9

* Access Key: oAPre[redacted]92Rr2jJEf

* Project Name: MySecondProject2

* End Point: http://service.odps.aliyun.com/api

OK Cancel

4. Click OK to complete addition. Then, the account will be displayed in the Account list on the Account configuration option page.

Delete an account

On the Account configuration option page, follow these steps: (This operation only deletes the account configuration on Studio configuration, which does not affect your account.)

1. Select the account to be deleted in the account list.
2. Click -.
3. In the displayed dialog box, click OK.

Modify the AccessKey of an account

On the Account configuration option page, follow these steps:

1. Select the account to be deleted in the Account list.
2. Click the pencil icon.
3. In the displayed Edit Account window, modify the account information. The content is similar to that in the preceding section Add Account.

View the opening and connection of MaxCompute Region and the settings of Endpoint, see *Endpoints and Data Centers*.

3 Eclipse Plugins

3.1 Install

To facilitate the development work with *Java SDK of MapReduce* and *UDF*, MaxCompute provides Eclipse Development Plug-in. This plug-in can simulate the running process of MapReduce and UDF to provide local debugging methods and simple template generation.

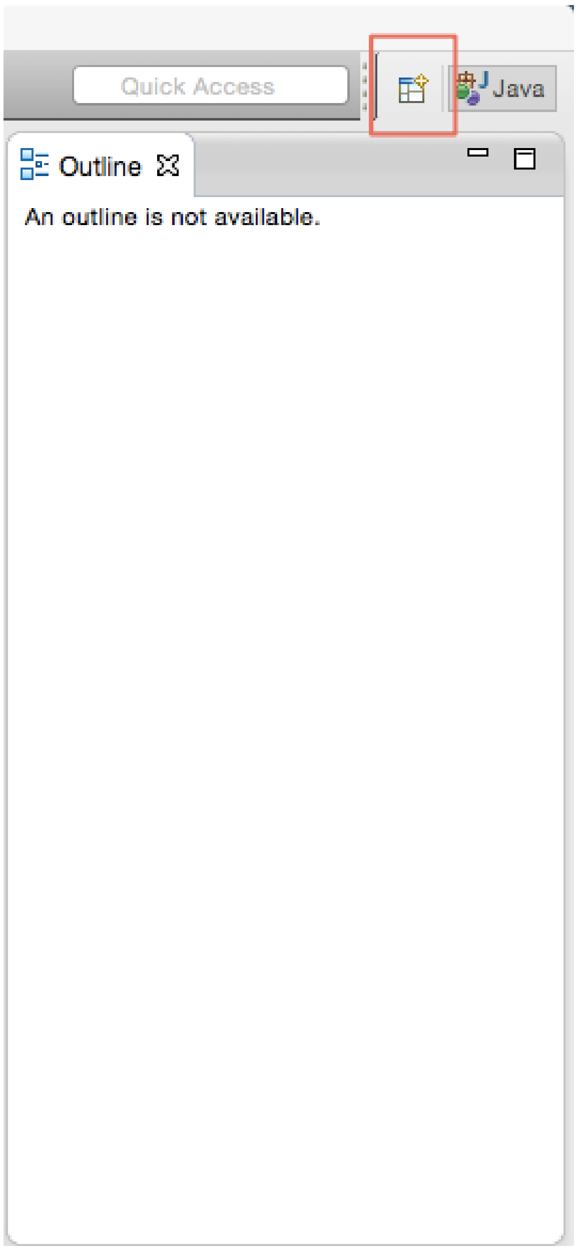
Note:

- Current versions of Eclipse Neon are likely to cause plug-in loading to fail, please use the Eclipse Luna version.
- To download this plug-in, click *Here*.
- Unlike the local running mode provided by MapReduce, Eclipse plug-in cannot synchronize data with MaxCompute. Data must be manually copied to the warehouse directory of Eclipse plug-in.

After downloading the Eclipse plug-in, decompress the software package to find the following jar:

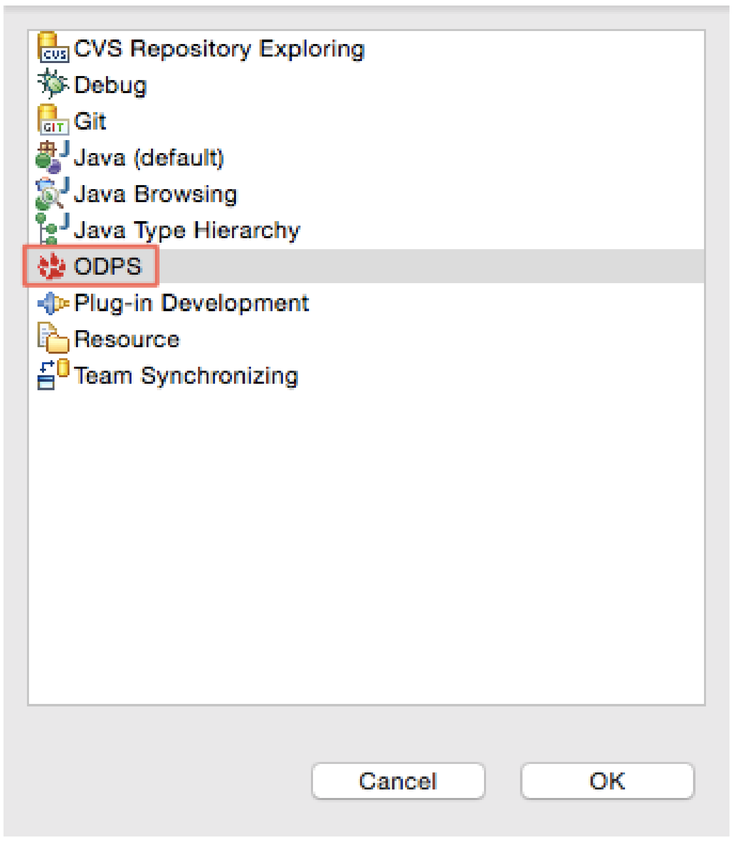
```
odps - eclipse - plugin - bundle - 0 . 16 . 0 . jar
```

Place the plug-in into the subdirectory plugins in Eclipse installation directory. Start the Eclipse plug-in, and click Open Perspective in the upper right corner.As follows.

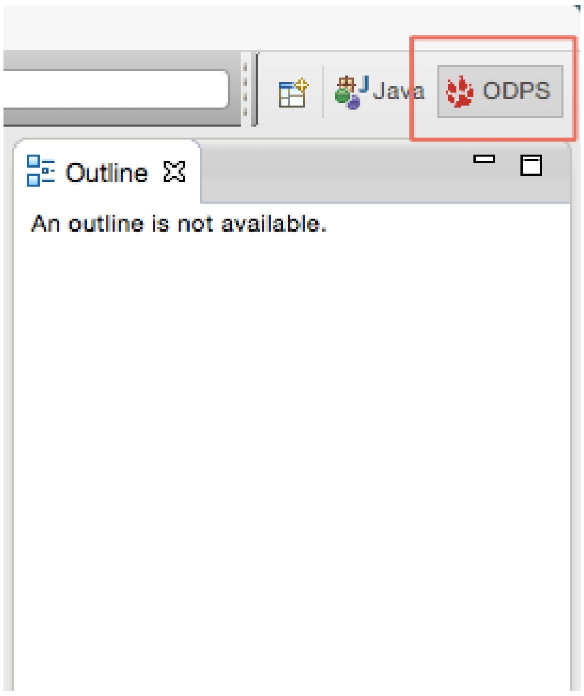


An outline is not available.

After clicking the button, the following dialog box is displayed.As follows.



Select ODPS and click OK. The MaxCompute icon appears in the upper right corner, indicating that the plug-in takes effect.As follows.



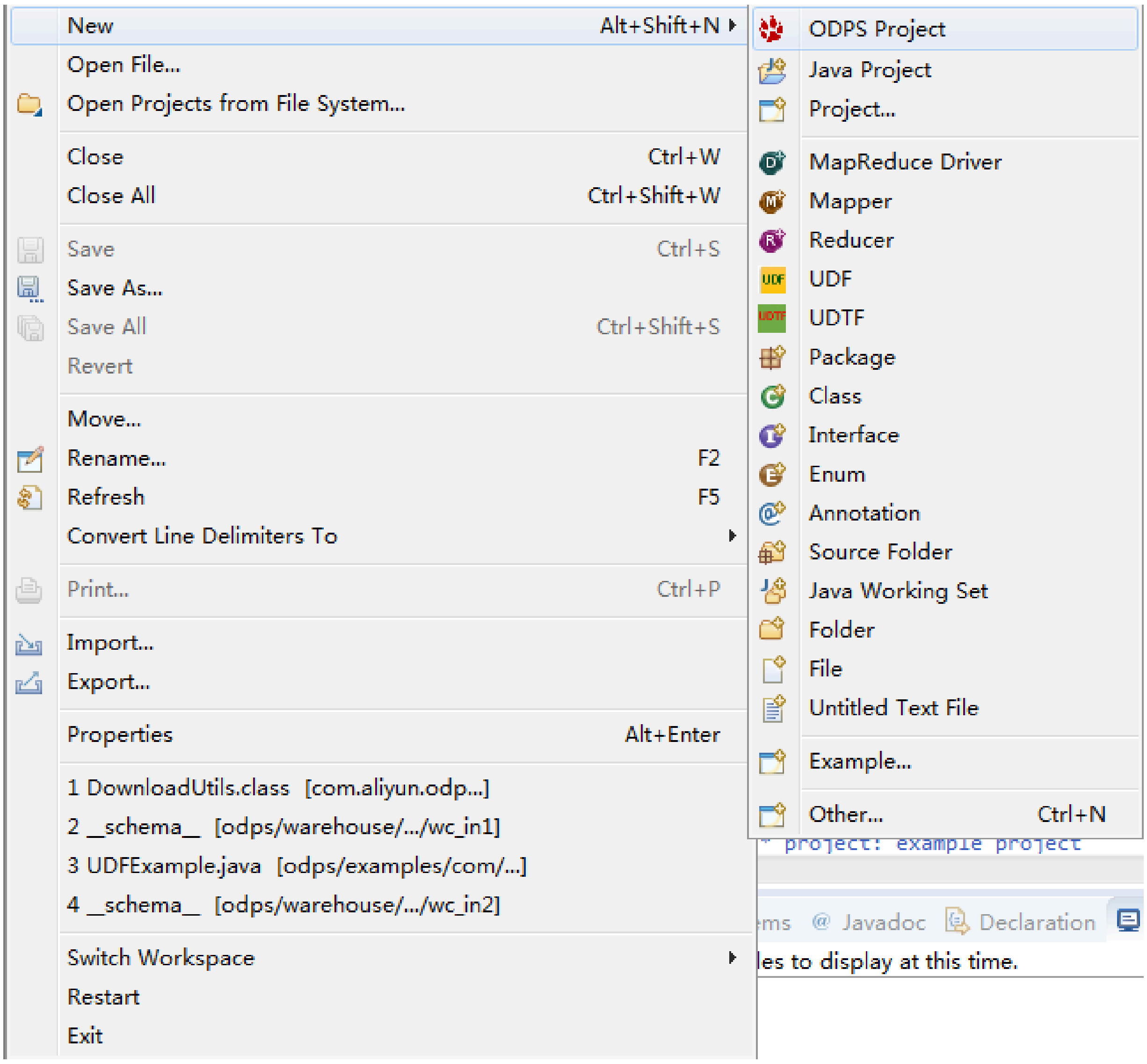
An outline is not available.

3.2 Create a project

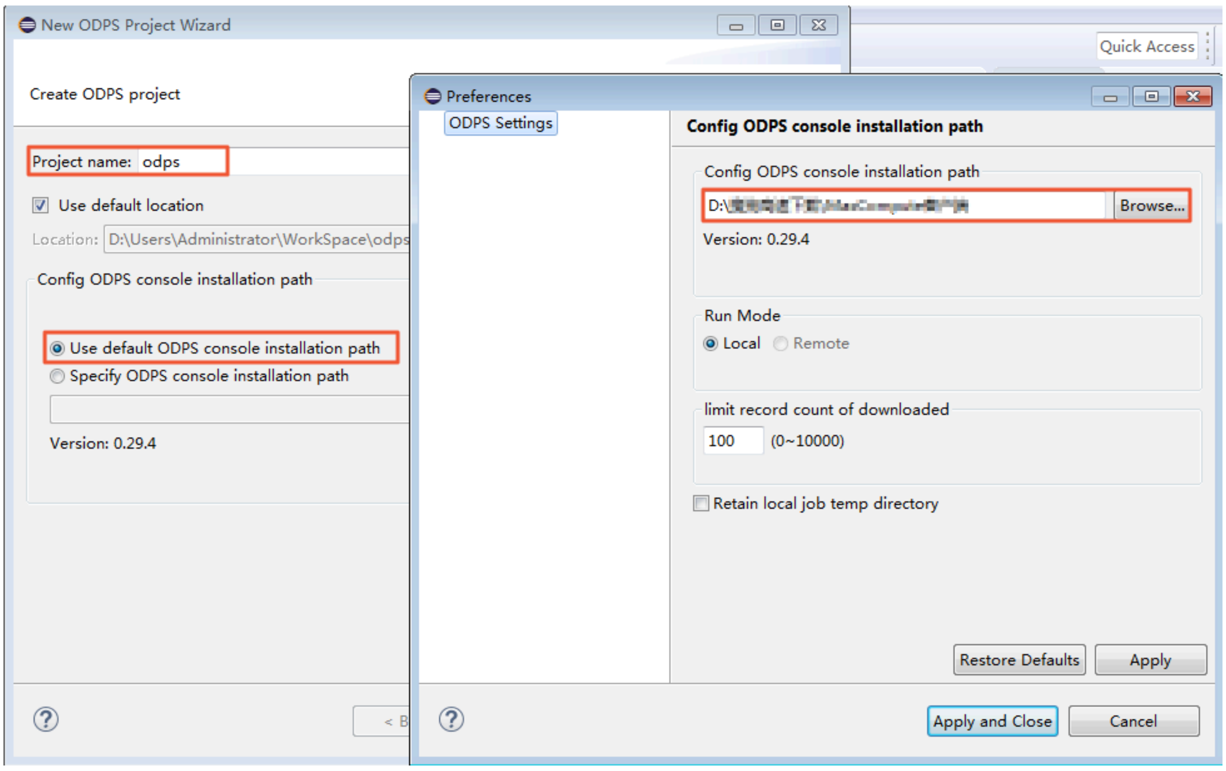
You create a MaxCompute project in two following ways.

Method 1

Select File > > New > > Project > > MaxCompute > > MaxCompute Project to create the project (in the example, use ODPS as the project name),as follows.



After creating MaxCompute project, the following dialog box is displayed. Enter the project name, select the MaxCompute client path, and click FinishFinish. The MaxCompute client needs to be downloaded in advance.As follows.


New ODPS Project Wizard

Create ODPS project

Project name: odps

[✓] Use default location

Location: D:\Users\Administrator\WorkSpace\odps

Config ODPS console installation path

(●) Use default ODPS console installation path

( ) Specify ODPS console installation path

Version: 0.29.4

Preferences

ODPS Settings

Config ODPS console installation path

Config ODPS console installation path

D:\[redacted] Browse...

Version: 0.29.4

Run Mode

(●) Local ( ) Remote

limit record count of downloaded

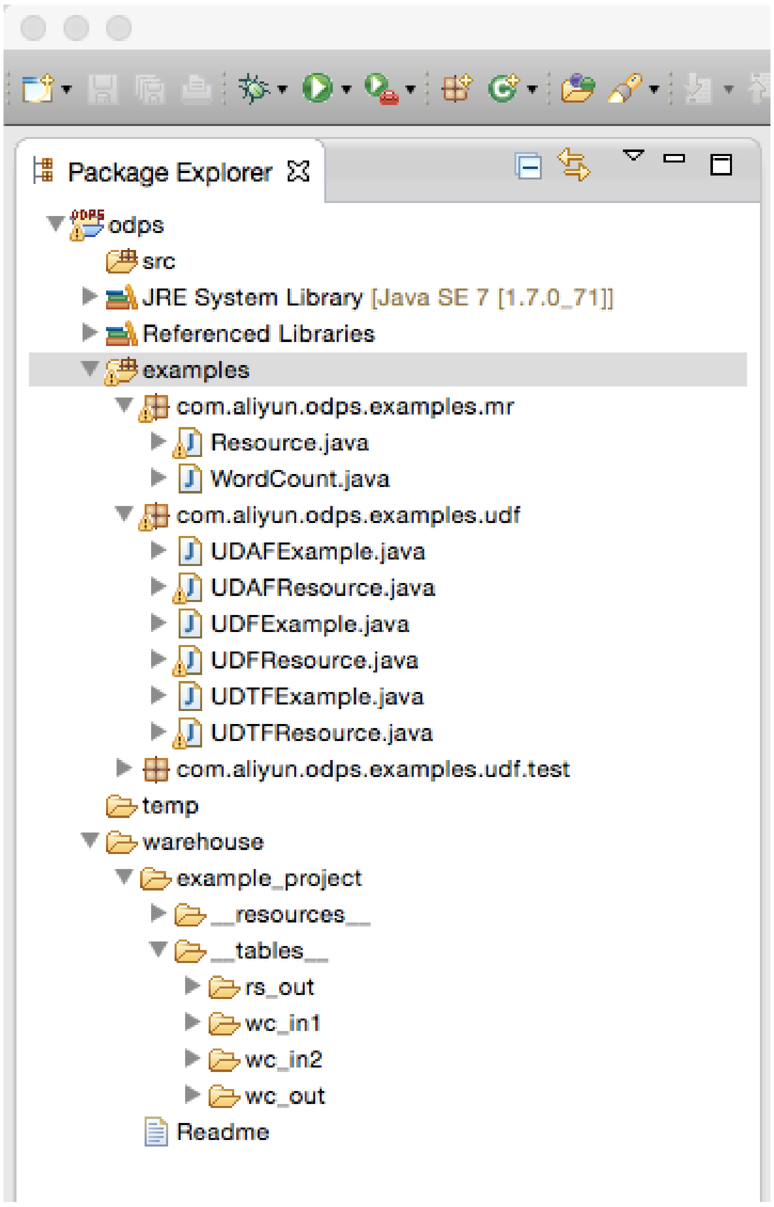
100 (0~10000)

[ ] Retain local job temp directory

Note:

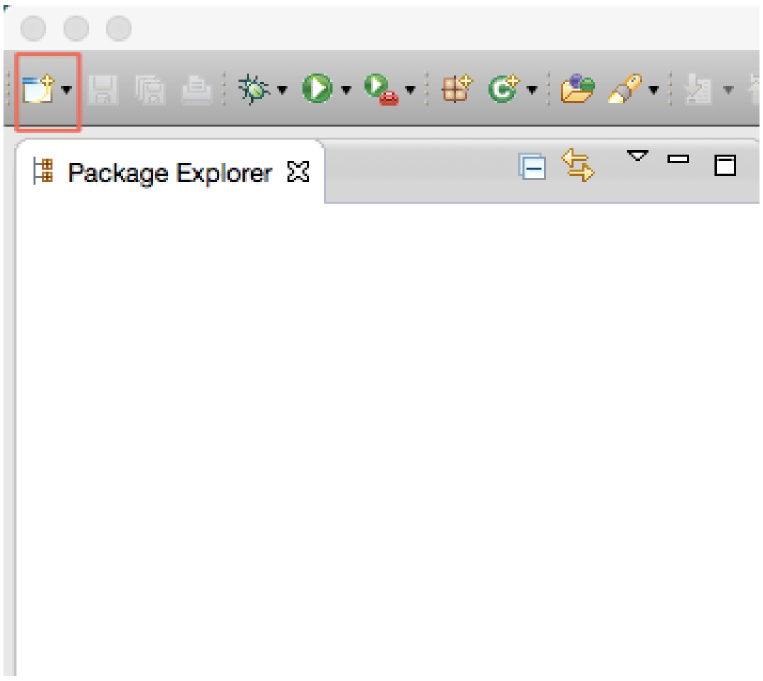
For more information about MaxCompute client, see *Client*.

After creating the project, the following directory structure is displayed in the left Package Explorer,as follows.

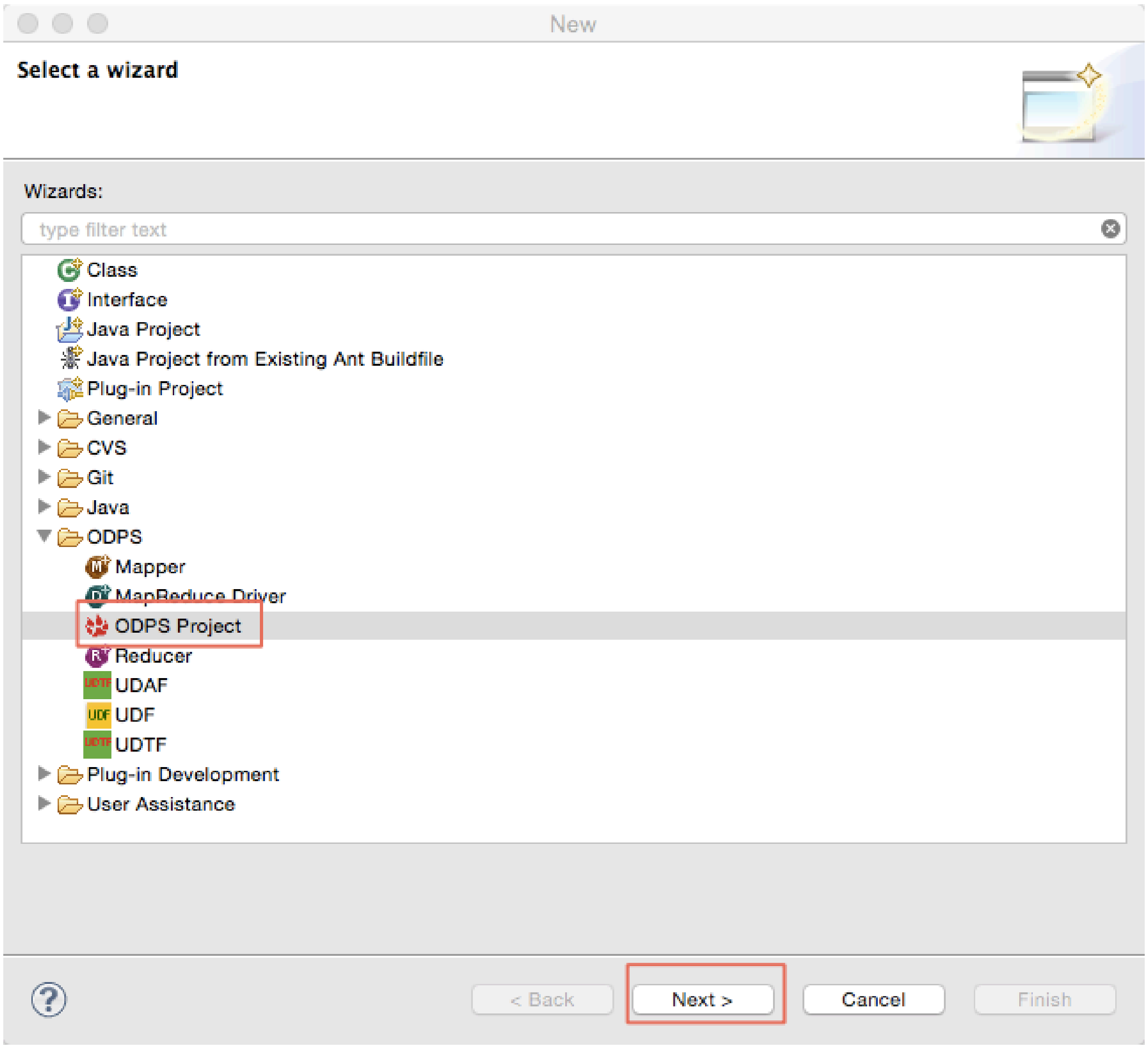


Method 2

Click New in the upper left corner,as follows.



After the dialog box is displayed, select ODPS Project and click Next,as follows.



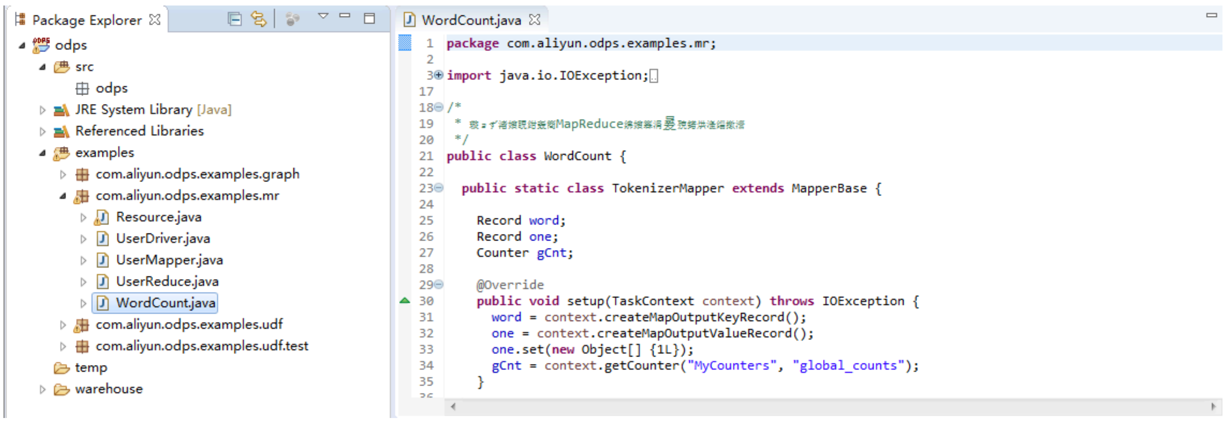
The subsequent operations are similar to Method 1.

The installation of MaxCompute Eclipse plugin is completed. You can use this plugin to write MapReduce or UDF programs. For more information about MapReduce, see MapReduce.For more information the UDF programming, see UDF.

3.3 MapReduce

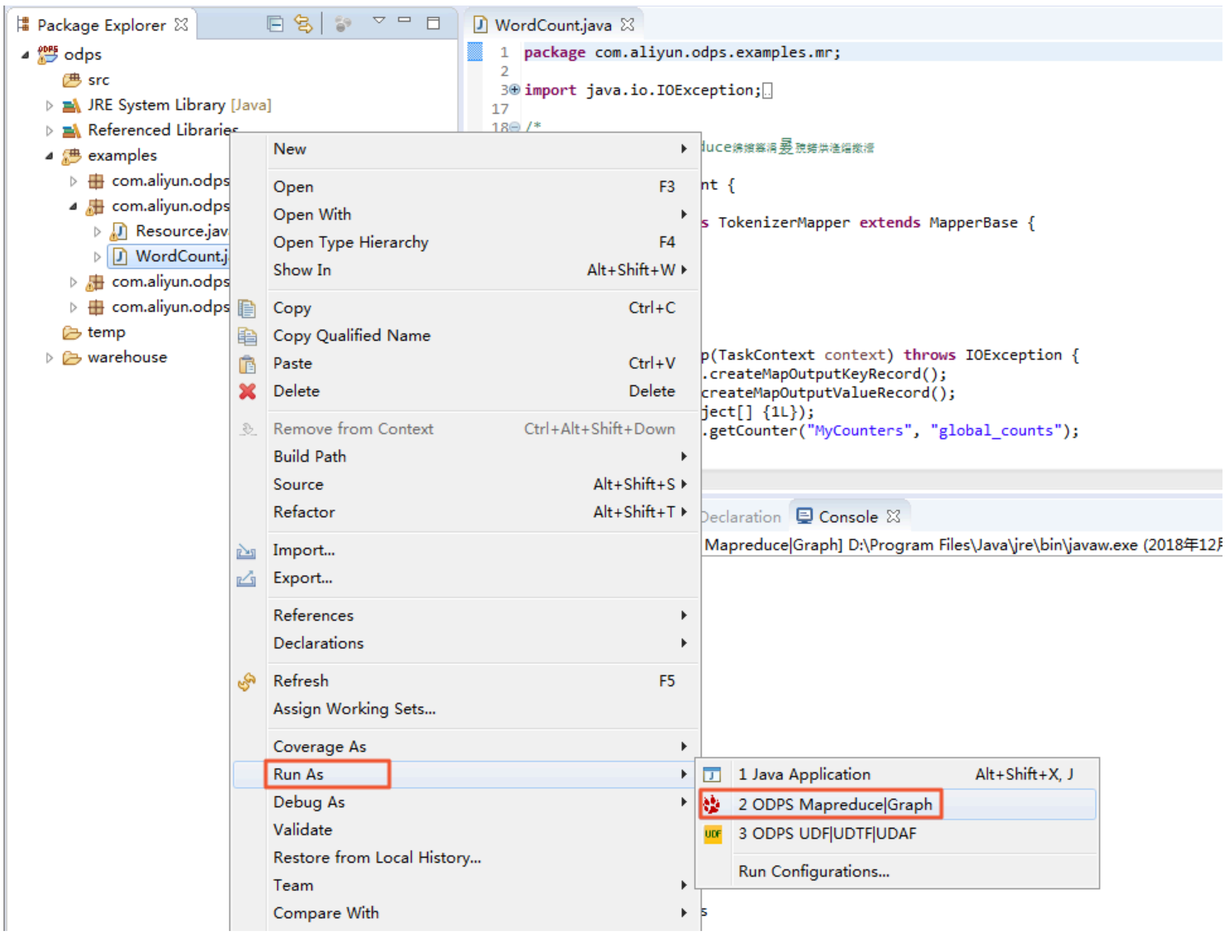
This article shows you how to use Eclipse to develop and run MapReduce programs.

Select WordCount example in MaxCompute project,as follows.



```java
package com.aliyun.odps.examples.mr;

import java.io.IOException;

/*
 * [illegible]MapReduce[illegible]
 */
public class WordCount {

  public static class TokenizerMapper extends MapperBase {

    Record word;
    Record one;
    Counter gCnt;

    @Override
    public void setup(TaskContext context) throws IOException {
      word = context.createMapOutputKeyRecord();
      one = context.createMapOutputValueRecord();
      one.set(new Object[] {1L});
      gCnt = context.getCounter("MyCounters", "global_counts");
    }
```

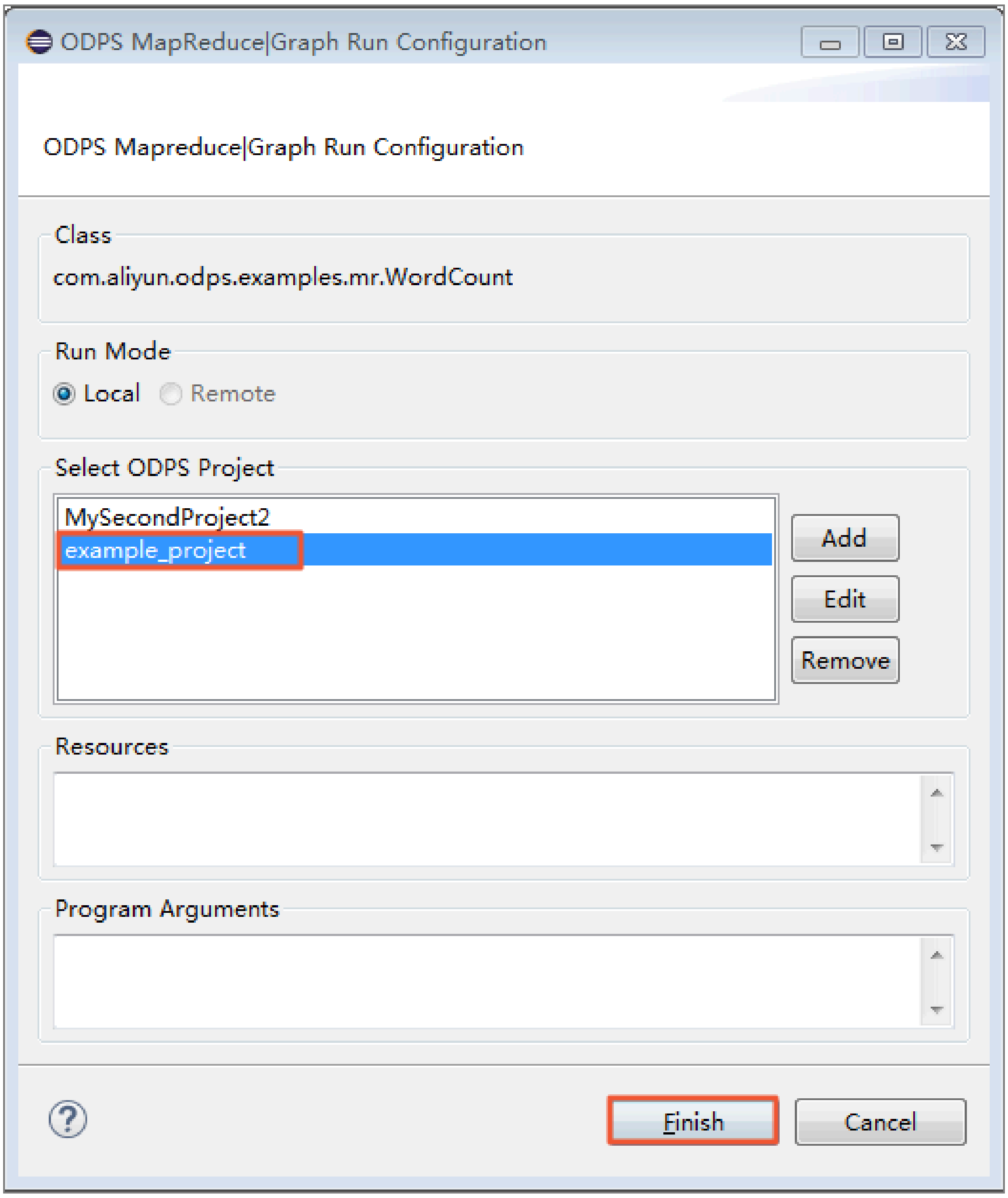
Right-click WordCount.java and choose Run As -> ODPS MapReduce, as follows.



```
package com.aliyun.odps.examples.mr;

import java.io.IOException;
/*
s TokenizerMapper extends MapperBase {
p(TaskContext context) throws IOException {
.createMapOutputKeyRecord();
createMapOutputValueRecord();
ject[] {1L});
.getCounter("MyCounters", "global_counts");
```

```
Mapreduce|Graph] D:\Program Files\Java\jre\bin\javaw.exe (2018年12
```

After the dialog box is popped up, select example_project and click Finish。 As follows.



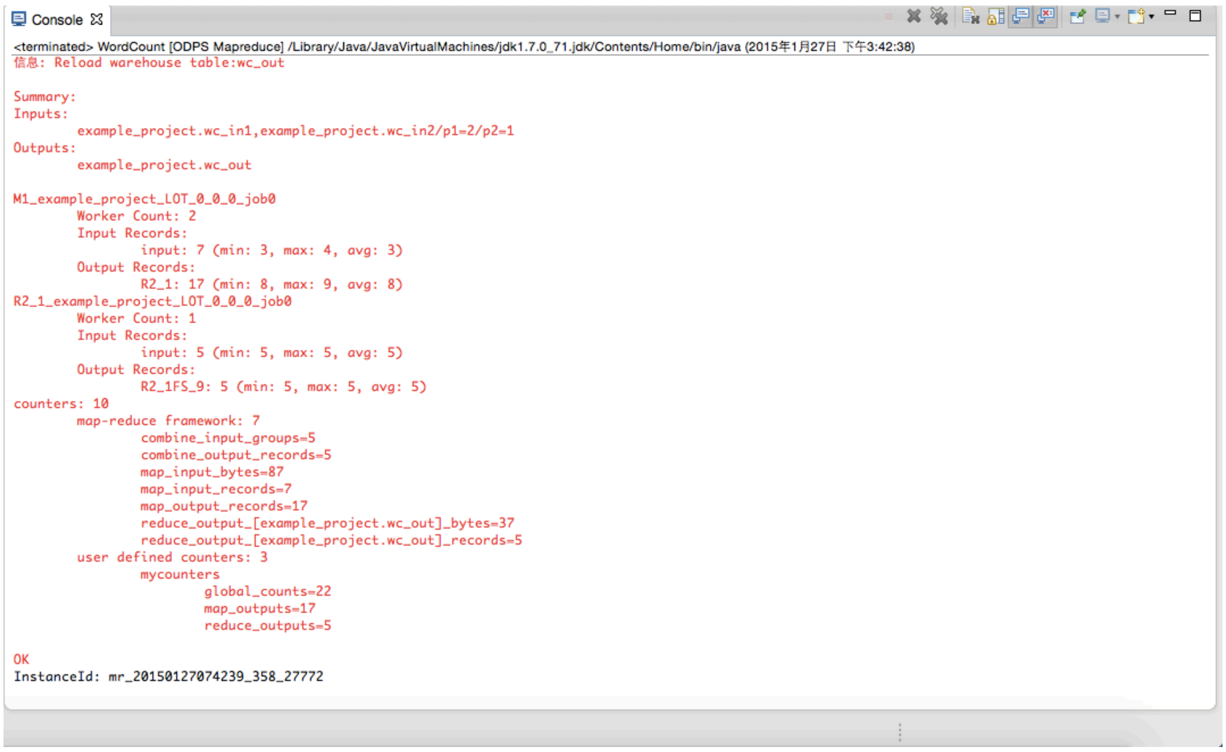
ODPS Mapreduce|Graph Run Configuration

Class
com.aliyun.odps.examples.mr.WordCount

Run Mode
[✓ Local] Remote

Select ODPS Project
MySecondProject2
example_project

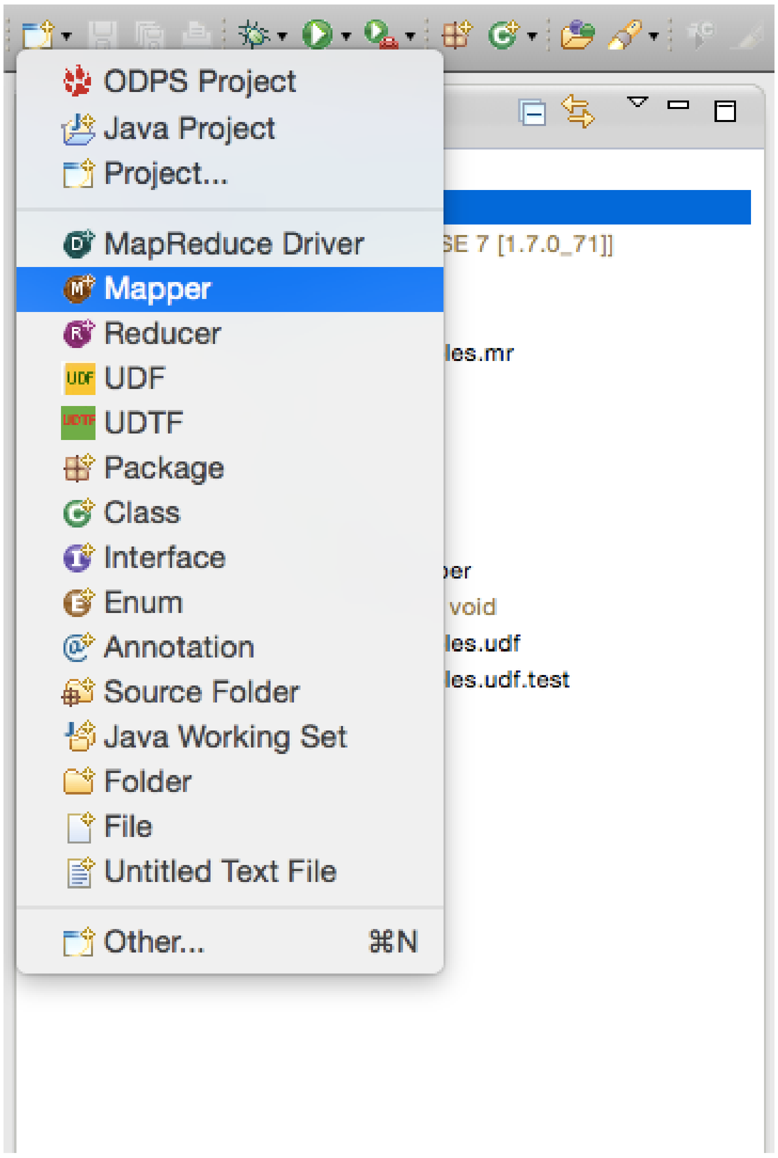
After running is completed, the following result is displayed,as follows.



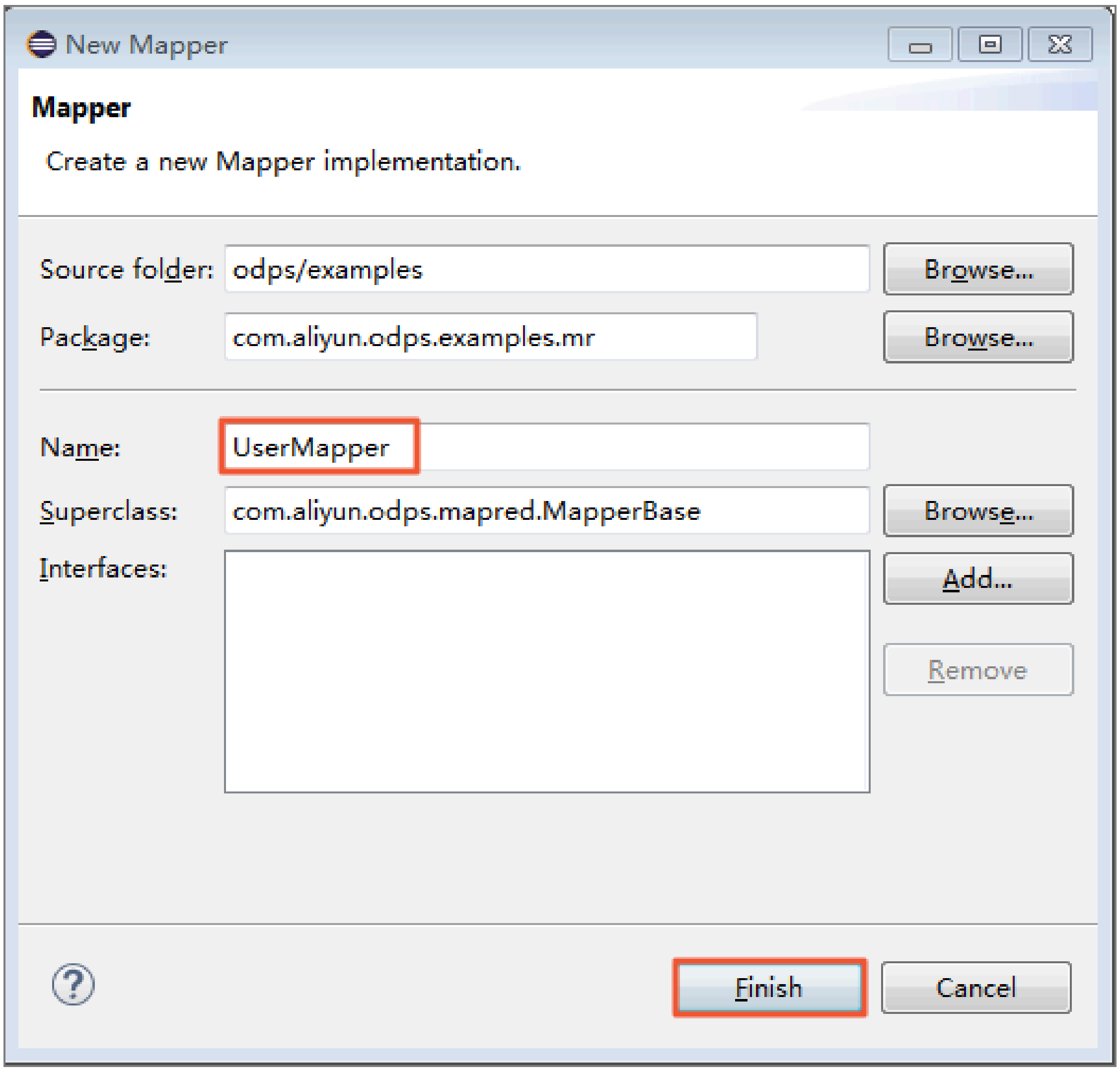
```
<terminated> WordCount [ODPS Mapreduce] /Library/Java/JavaVirtualMachines/jdk1.7.0_71.jdk/Contents/Home/bin/java (2015年1月27日 下午3:42:38)
信息: Reload warehouse table:wc_out

Summary:
Inputs:
        example_project.wc_in1,example_project.wc_in2/p1=2/p2=1
Outputs:
        example_project.wc_out

M1_example_project_LOT_0_0_0_job0
        Worker Count: 2
        Input Records:
                input: 7 (min: 3, max: 4, avg: 3)
        Output Records:
                R2_1: 17 (min: 8, max: 9, avg: 8)
R2_1_example_project_LOT_0_0_0_job0
        Worker Count: 1
        Input Records:
                input: 5 (min: 5, max: 5, avg: 5)
        Output Records:
                R2_1FS_9: 5 (min: 5, max: 5, avg: 5)
counters: 10
        map-reduce framework: 7
                combine_input_groups=5
                combine_output_records=5
                map_input_bytes=87
                map_input_records=7
                map_output_records=17
                reduce_output_[example_project.wc_out]_bytes=37
                reduce_output_[example_project.wc_out]_records=5
        user defined counters: 3
                mycounters
                        global_counts=22
                        map_outputs=17
                        reduce_outputs=5

OK
InstanceId: mr_20150127074239_358_27772
```

Run User-defined MapReduce Program

Right-click src directory. Select New -> > Mapper,as follows.



After selecting Mapper, the following dialog box is displayed. Input the name of Mapper class and click Finish.as follows.



New Mapper

Mapper

Create a new Mapper implementation.

Source folder: odps/examples

Package: com.aliyun.odps.examples.mr

Name: UserMapper

Superclass: com.aliyun.odps.mapred.MapperBase

Interfaces:

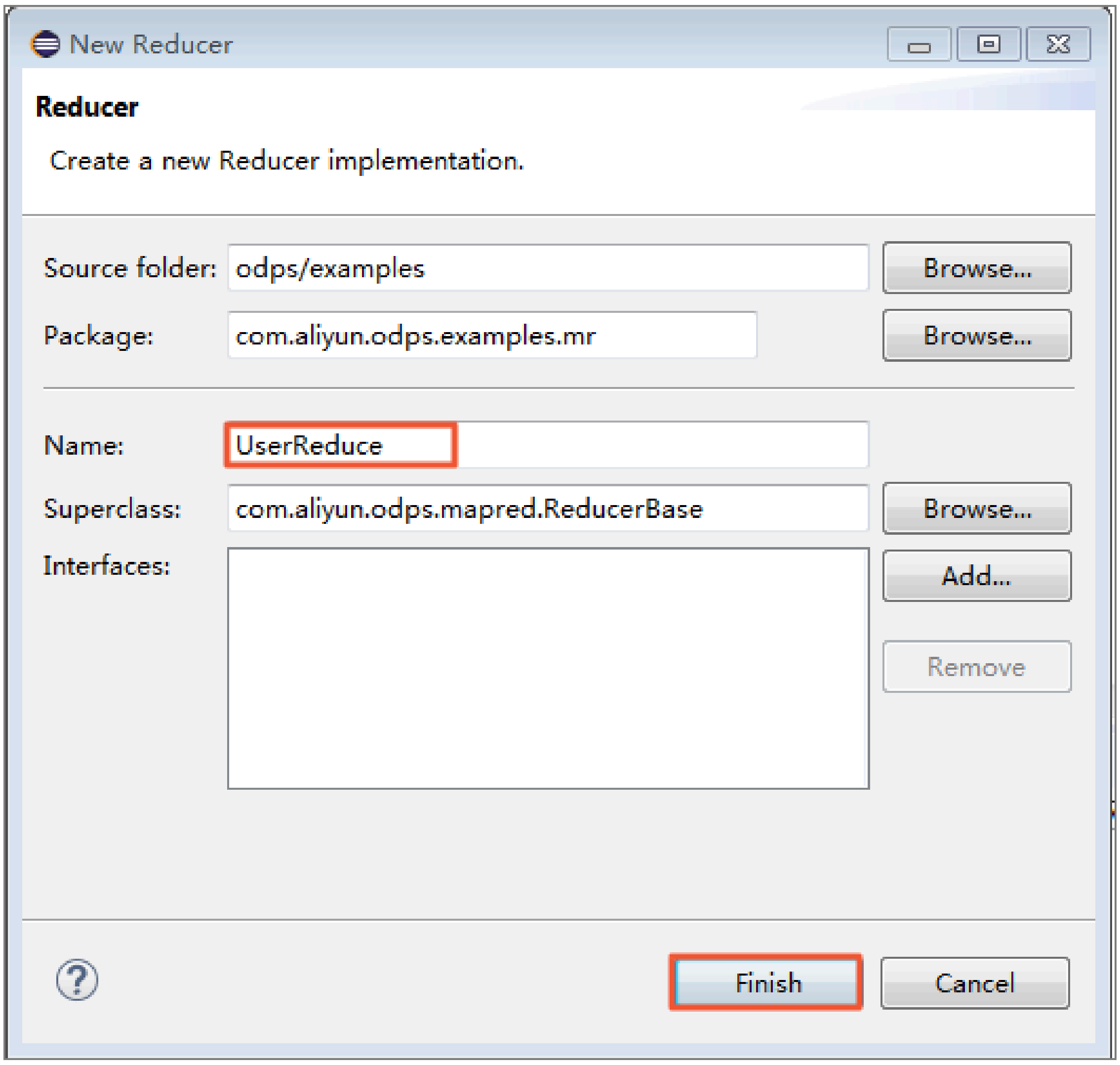
The file UserMapper.java is generated in the src directory in Package Explorer. The content of this file is a template of Mapper class,as follows.

```
package   odps ;
import   java . io . IOExceptio  n ;
import   com . aliyun . odps . data . Record ;
import   com . aliyun . odps . mapred . MapperBase ;
public   class   UserMapper   extends   MapperBase  {
   @ Override
    public   void   setup ( TaskContex  t   context )  throws
IOExceptio  n  {

   @ Override
    public   void   map ( long   recordNum ,  Record   record ,
TaskContex  t   context )
            throws   IOExceptio  n  {

   @ Override
    public   void   cleanup ( TaskContex  t   context )  throws
IOExceptio  n  {
```

In the template, the configured package name defaults to odps. You can modify it according to your actual requirement. Write the template content as follows,as follows.

```
package   odps ;
import   java . io . IOExceptio  n ;
import   com . aliyun . odps . counter . Counter ;
import   com . aliyun . odps . data . Record ;
import   com . aliyun . odps . mapred . MapperBase ;
public   class   UserMapper   extends   MapperBase  {
    Record   word ;
    Record   one ;
    Counter   gCnt ;
   @ Override
    public   void   setup ( TaskContex  t   context )  throws
IOExceptio  n  {
          word  =  context . createMapO  utputKeyRe  cord ();
          one  =  context . createMapO  utputValue  Record ();
          one . set ( new   Object [] {  1L  });
          gCnt  =  context . getCounter (" MyCounters ", "
global_cou  nts ");

   @ Override
    public   void   map ( long   recordNum ,  Record   record ,
TaskContex  t   context )
            throws   IOExceptio  n  {
          for  ( int   i  =  0 ;  i  <  record . getColumnC  ount
();  i ++) {
              String []  words  =  record . get ( i ). toString ().
split ("\\ s +");
              for  ( String   w  :  words ) {
                word . set ( new   Object [] {  w  });
                Counter   cnt  =  context . getCounter (" MyCounters
", " map_output  s ");
                cnt . increment ( 1 );
                gCnt . increment ( 1 );
                context . write ( word ,  one );



   @ Override
    public   void   cleanup ( TaskContex  t   context )  throws
IOExceptio  n  {
```

Similarly, right-click src directory and select New -> > Reduce:,as follows.



New Reducer

Reducer

Create a new Reducer implementation.

Source folder: odps/examples

Package: com.aliyun.odps.examples.mr

Name: UserReduce

Superclass: com.aliyun.odps.mapred.ReducerBase

Interfaces:

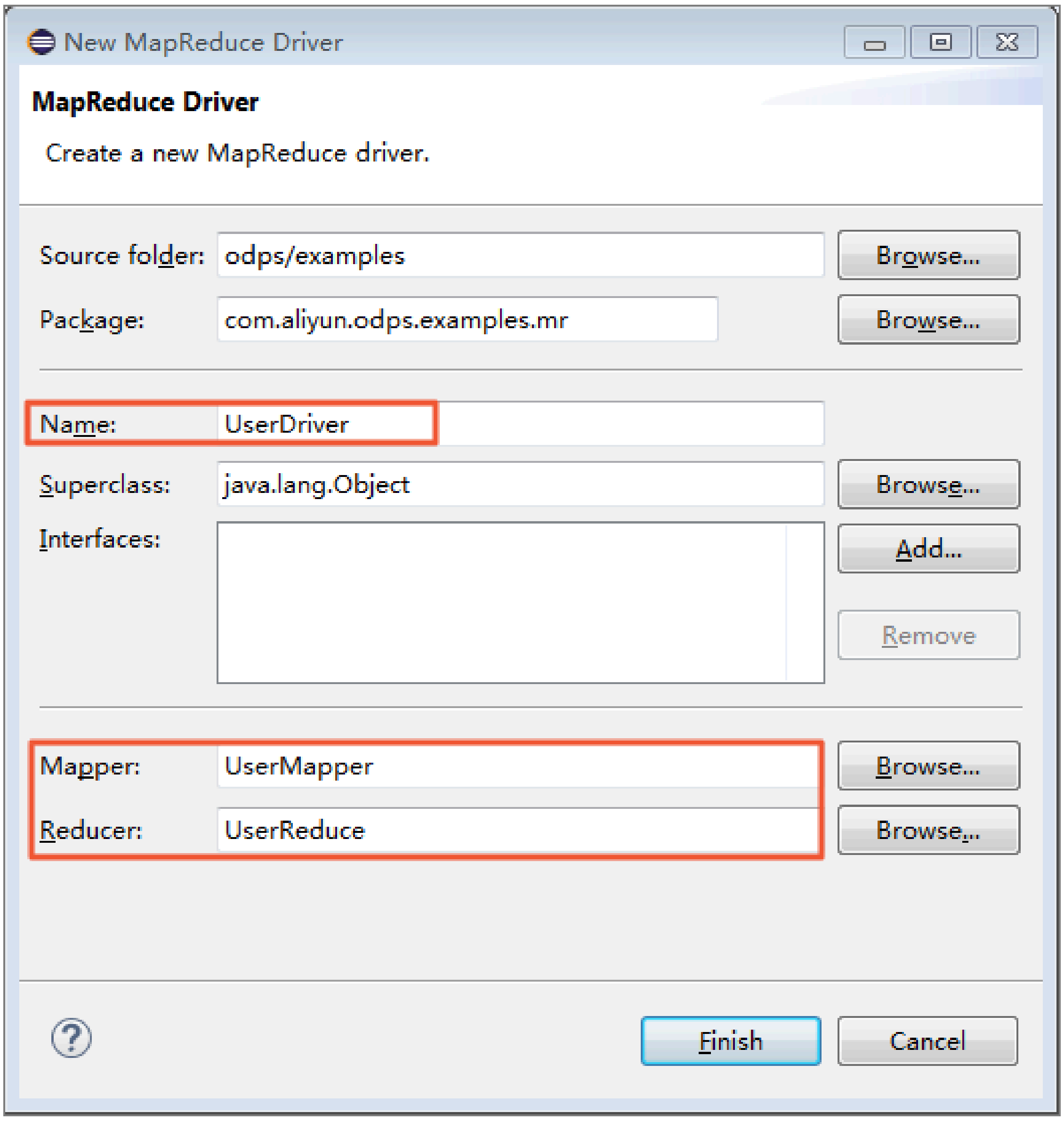
Input the name of Reduce class. (In this example, use UserReduce as the class name.)

In Package Explorer, a file name UserReduce.java is generated in the src directory. This file content is a template of Reduce class. Edit the template,as follows.

```
package   odps ;
import   java . io . IOExceptio  n ;
import   java . util . Iterator ;
import   com . aliyun . odps . counter . Counter ;
import   com . aliyun . odps . data . Record ;
import   com . aliyun . odps . mapred . ReducerBas  e ;
public   class   UserReduce   extends   ReducerBas  e  {
    private   Record   result ;
    Counter   gCnt ;
   @ Override
    public   void   setup ( TaskContex  t   context )  throws
IOExceptio  n  {
          result  =  context . createOutp  utRecord ();
          gCnt  =  context . getCounter (" MyCounters ", "
global_cou  nts ");

   @ Override
    public   void   reduce ( Record   key ,  Iterator < Record >
values ,  TaskContex  t   context )
```

```
              throws   IOExceptio  n  {
            long   count  =  0 ;
            while  ( values . hasNext ()) {
              Record   val  =  values . next ();
              count  += ( Long )  val . get ( 0 );

            result . set ( 0 ,  key . get ( 0 ));
            result . set ( 1 ,  count );
            Counter   cnt  =  context . getCounter (" MyCounters ", "
reduce_out  puts ");
            cnt . increment ( 1 );
            gCnt . increment ( 1 );
            context . write ( result );

   @ Override
    public   void   cleanup ( TaskContex  t   context )  throws
IOExceptio  n  {
```

Create main function: right-click src and select New -> > MapReduce Driver. Enter Driver Name (in this example, use UserDriver as the name), Mapper and Reduce (in this example use UserMapper and UserReduce as corresponding names) and click Finish. The file MyDriver.java is also displayed in src directory,as follows.


New MapReduce Driver

MapReduce Driver

Create a new MapReduce driver.

Source folder: odps/examples

Package: com.aliyun.odps.examples.mr

Name: UserDriver

Superclass: java.lang.Object

Interfaces:

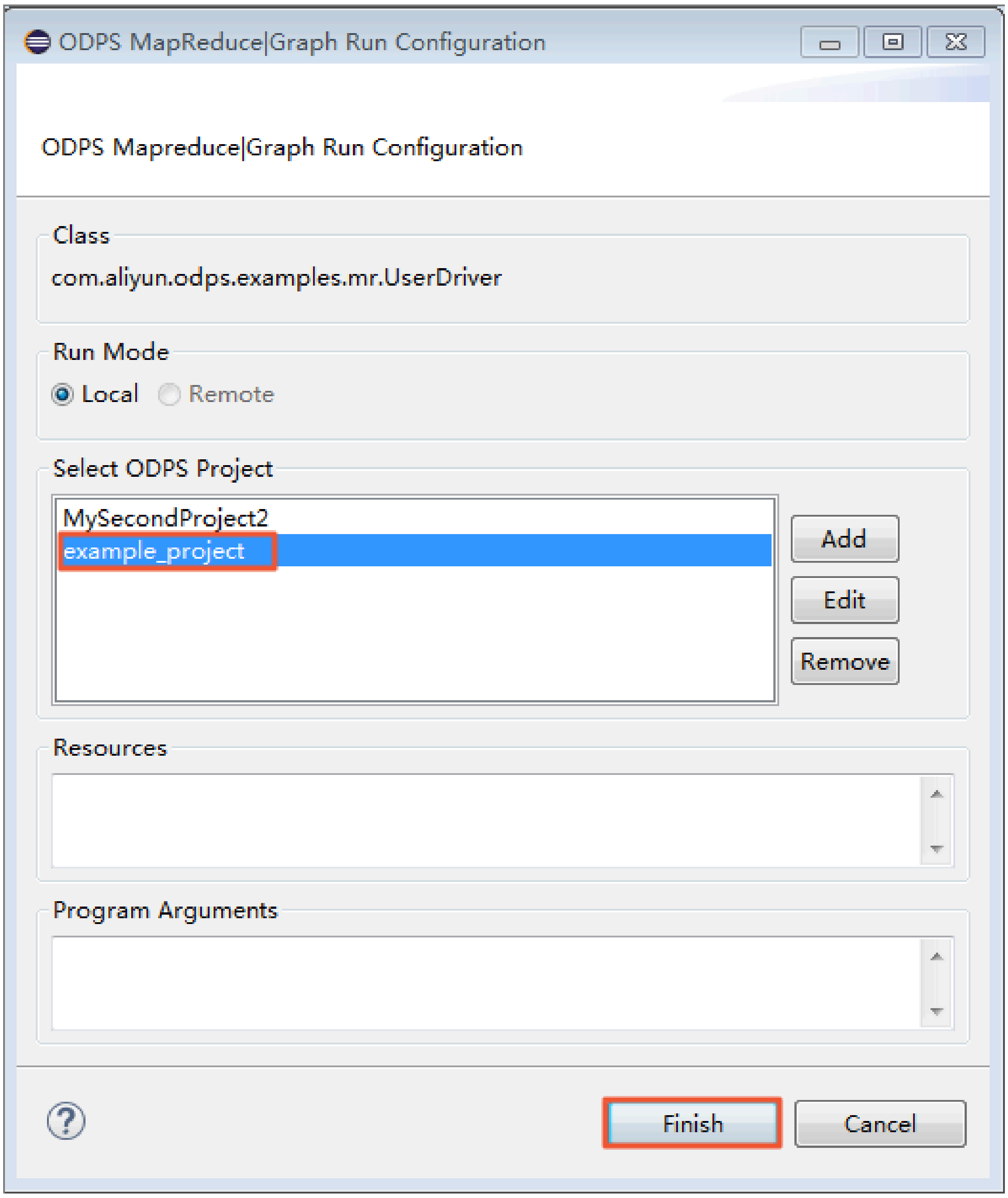
Mapper: UserMapper

Reducer: UserReduce

Edit the driver content,as follows.

```
package   odps ;
import   com . aliyun . odps . OdpsExcept  ion ;
import   com . aliyun . odps . data . TableInfo ;
import   com . aliyun . odps . examples . mr . WordCount .
SumCombine  r ;
import   com . aliyun . odps . examples . mr . WordCount . SumReducer
;
import   com . aliyun . odps . examples . mr . WordCount .
TokenizerM  apper ;
import   com . aliyun . odps . mapred . JobClient ;
import   com . aliyun . odps . mapred . RunningJob ;
import   com . aliyun . odps . mapred . conf . JobConf ;
import   com . aliyun . odps . mapred . utils . InputUtils ;
import   com . aliyun . odps . mapred . utils . OutputUtil  s ;
import   com . aliyun . odps . mapred . utils . SchemaUtil  s ;
public   class   UserDriver  {
    public   static   void   main ( String []  args )  throws
OdpsExcept  ion  {
        JobConf   job  =  new   JobConf ();
        job . setMapperC  lass ( TokenizerM  apper . class );
        job . setCombine  rClass ( SumCombine  r . class );
        job . setReducer  Class ( SumReducer . class );
        job . setMapOutp  utKeySchem  a ( SchemaUtil  s . fromString
(" word : string "));
        job . setMapOutp  utValueSch  ema ( SchemaUtil  s .
fromString (" count : bigint "));
        InputUtils . addTable (
            TableInfo . builder (). tableName (" wc_in1 "). cols (
new   String [] { " col2 ", " col3 " }). build (),  job );
        InputUtils . addTable ( TableInfo . builder (). tableName ("
wc_in2 "). partSpec (" p1 = 2 / p2 = 1 "). build (),  job );
        OutputUtil  s . addTable ( TableInfo . builder (). tableName
(" wc_out "). build (),  job );
        RunningJob   rj  =  JobClient . runJob ( job );
        rj . waitForCom  pletion ();
```

Run MapReduce program. Right-click UserDriver.java and select Run As -> > ODPS MapReduce, the following dialog box is displayed,as follows.


ODPS MapReduce|Graph Run Configuration

ODPS Mapreduce|Graph Run Configuration

Class

com.aliyun.odps.examples.mr.UserDriver

Run Mode

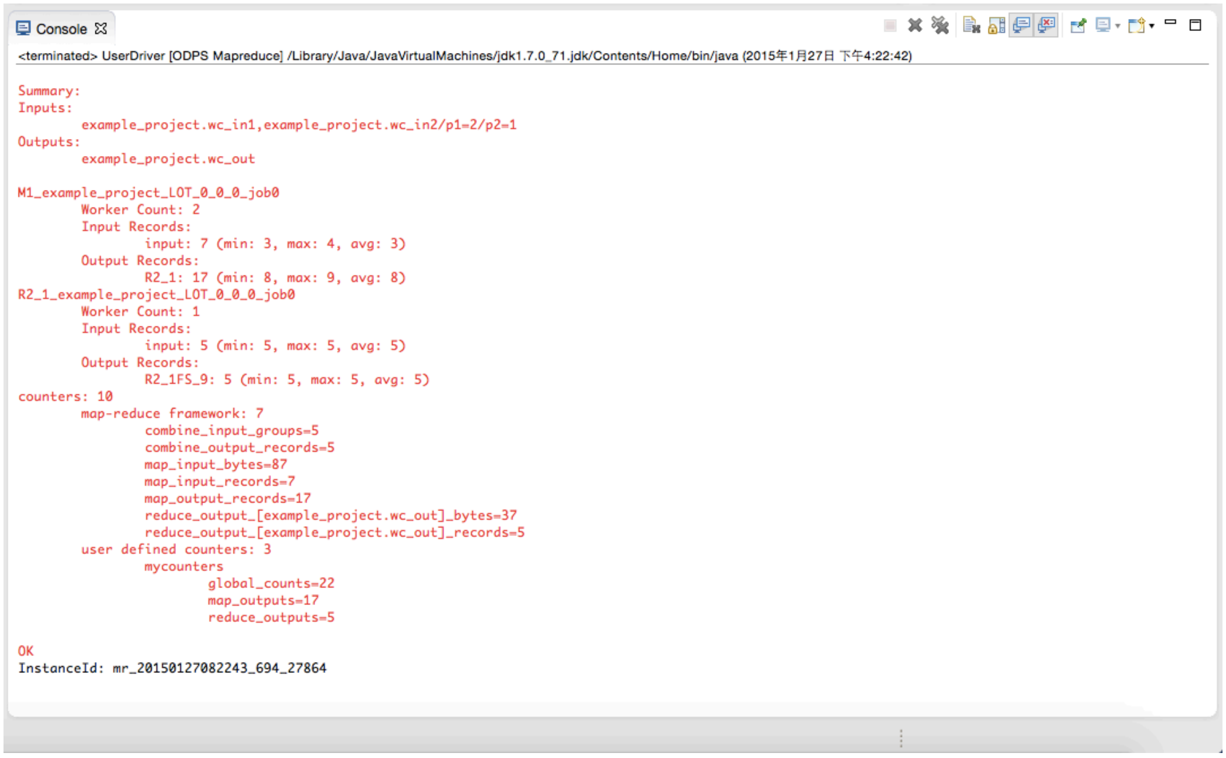
[✓ Local] Remote

Select ODPS Project

MySecondProject2

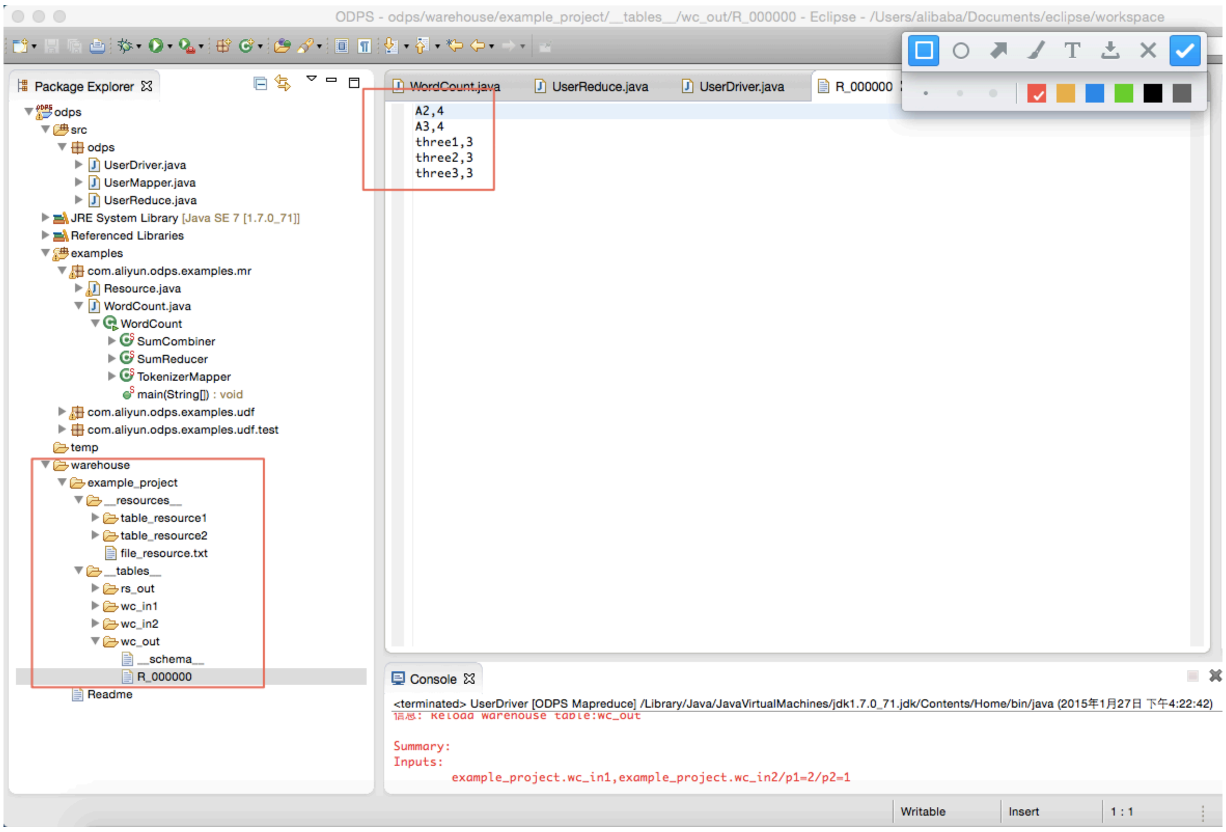
[✓ example_project]

Select example_project as the MaxCompute Project and click Finish to run MapReduce program in the local,as follows.



```
<terminated> UserDriver [ODPS Mapreduce] /Library/Java/JavaVirtualMachines/jdk1.7.0_71.jdk/Contents/Home/bin/java (2015年1月27日 下午4:22:42)

Summary:
Inputs:
        example_project.wc_in1,example_project.wc_in2/p1=2/p2=1
Outputs:
        example_project.wc_out

M1_example_project_LOT_0_0_0_job0
        Worker Count: 2
        Input Records:
                input: 7 (min: 3, max: 4, avg: 3)
        Output Records:
                R2_1: 17 (min: 8, max: 9, avg: 8)
R2_1_example_project_LOT_0_0_0_job0
        Worker Count: 1
        Input Records:
                input: 5 (min: 5, max: 5, avg: 5)
        Output Records:
                R2_1FS_9: 5 (min: 5, max: 5, avg: 5)
counters: 10
        map-reduce framework: 7
                combine_input_groups=5
                combine_output_records=5
                map_input_bytes=87
                map_input_records=7
                map_output_records=17
                reduce_output_[example_project.wc_out]_bytes=37
                reduce_output_[example_project.wc_out]_records=5
        user defined counters: 3
                mycounters
                        global_counts=22
                        map_outputs=17
                        reduce_outputs=5

OK
InstanceId: mr_20150127082243_694_27864
```

If the output is the same as in the preceding figure, it indicates that local operation runs successfully. The output result is saved in the warehouse directory. Refresh MaxCompute project,as follows.



```
ODPS - odps/warehouse/example_project/__tables__/wc_out/R_000000 - Eclipse - /Users/alibaba/Documents/eclipse/workspace
```

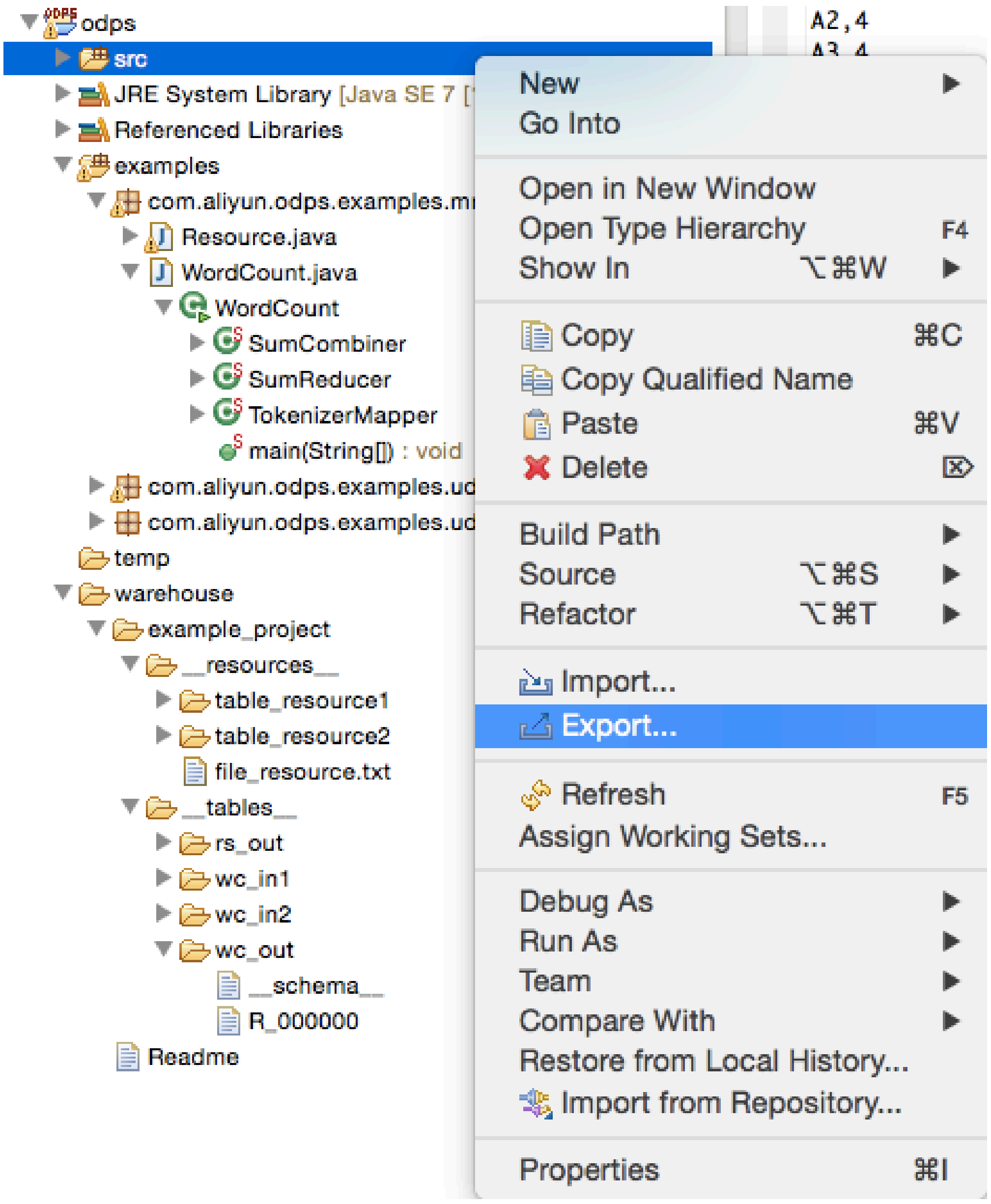
```
odps
  src
    odps
      UserDriver.java
      UserMapper.java
      UserReduce.java
  JRE System Library [Java SE 7 [1.7.0_71]]
  Referenced Libraries
  examples
    com.aliyun.odps.examples.mr
      Resource.java
      WordCount.java
        WordCount
          SumCombiner
          SumReducer
          TokenizerMapper
          main(String[]) : void
    com.aliyun.odps.examples.udf
    com.aliyun.odps.examples.udf.test
  temp
  warehouse
    example_project
      __resources__
        table_resource1
        table_resource2
        file_resource.txt
      __tables__
        rs_out
        wc_in1
        wc_in2
        wc_out
          __schema__
          R_000000
  Readme
```

```
A2,4
A3,4
three1,3
three2,3
three3,3
```

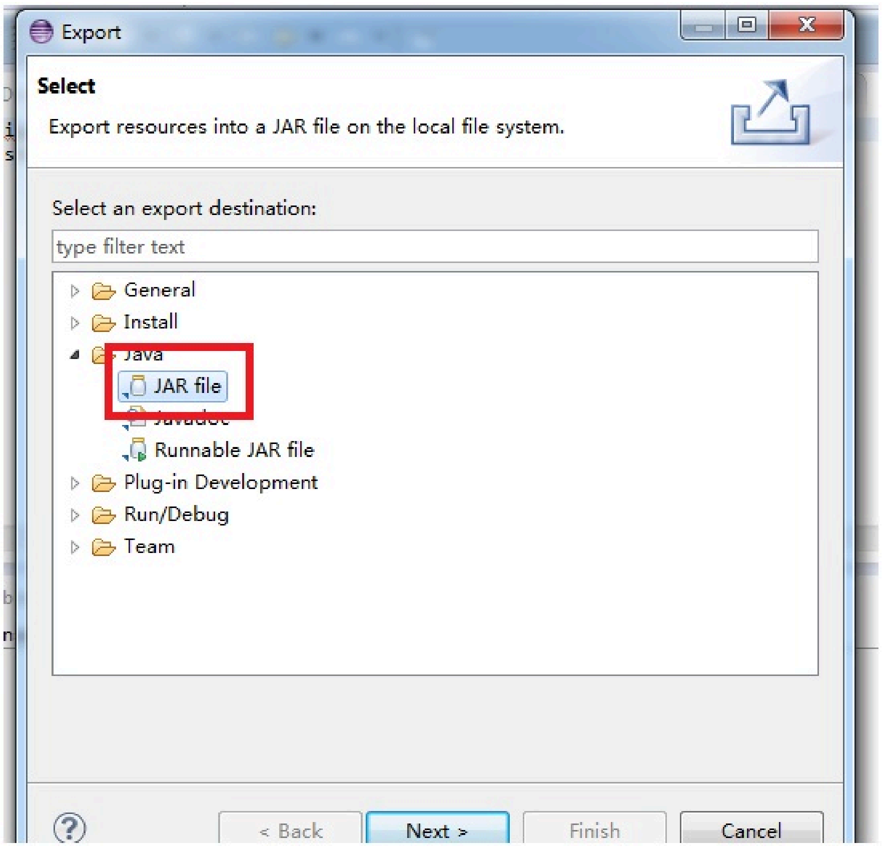
```
<terminated> UserDriver [ODPS Mapreduce] /Library/Java/JavaVirtualMachines/jdk1.7.0_71.jdk/Contents/Home/bin/java (2015年1月27日 下午4:22:42)
信息: Reload warehouse table:wc_out

Summary:
Inputs:
        example_project.wc_in1,example_project.wc_in2/p1=2/p2=1
```

wc_out is the output directory and R_000000 is the result file. By local debugging, the result is confirmed to be correct and you can package MapReduce program using Eclipse export function. After it is packaged, upload the jar package to MaxCompute. For more information how to run MapReduce in distributed environment, see *Quick Start*.

After the local debugging is completed, you can package the codes in jar package using Eclipse Export function, provided for subsequent distributed environment. In this example, the package name is mr-examples.jar. Select the src directory and click Export,as follows.



Select Jar File as an export mode,as follows.


Export

Select

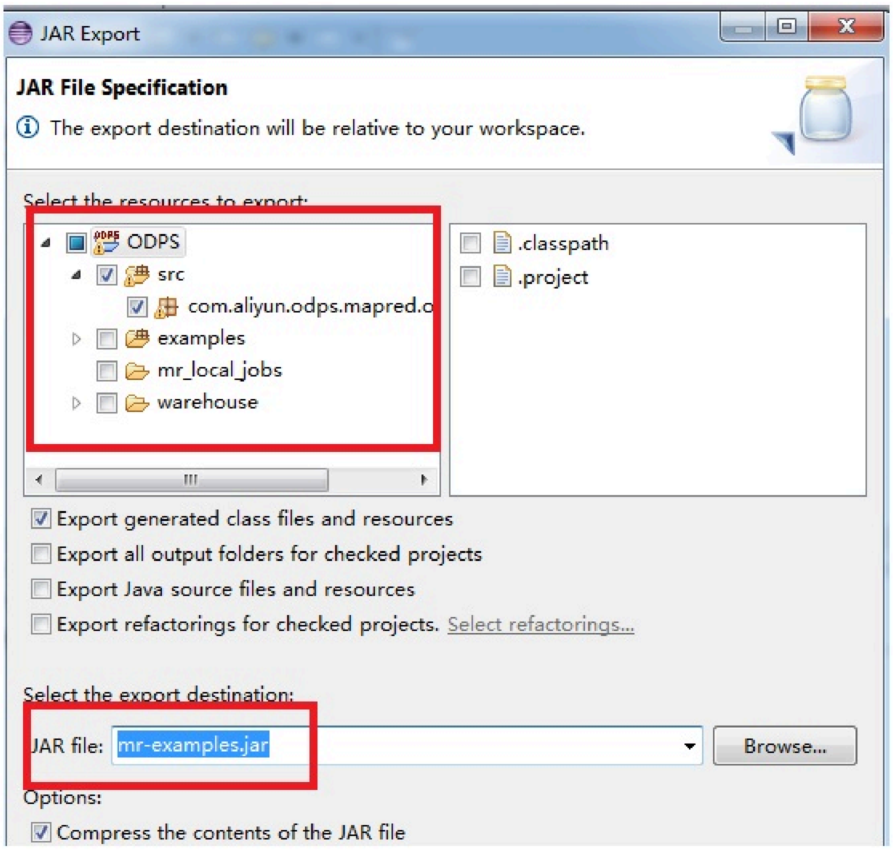
Export resources into a JAR file on the local file system.

Select an export destination:

type filter text

- General
- Install
- Java
  - JAR file
  - Javadoc
  - Runnable JAR file
- Plug-in Development
- Run/Debug
- Team

You must only export the package in src. The Jar File name must be specified as `mr - examples . jar` ,as follows.


JAR Export

JAR File Specification

The export destination will be relative to your workspace.

Select the resources to export:

- [■] ODPS
  - [✓] src
    - [✓] com.aliyun.odps.mapred.o
  - [ ] examples
  - [ ] mr_local_jobs
  - [ ] warehouse

[ ] .classpath
[ ] .project

[✓] Export generated class files and resources
[ ] Export all output folders for checked projects
[ ] Export Java source files and resources
[ ] Export refactorings for checked projects. Select refactorings...

Select the export destination:

JAR file: mr-examples.jar

Options:

[✓] Compress the contents of the JAR file

Click Next to export the jar file.

If you want to simulate new Project creation in the local, you can create a subdirectory (has same level with example_project) in the warehouse directory. The directory hierarchy structure is shown as follows.

```
< warehouse >
   | ____exampl  e_project ( Project   Dirctory )
          | ____  < __tables__ >
          | | __table_na  me1 ( non - partition   table )
          | | | ____   data ( File )

          | | | ____  < __schema__ > ( File )

          | | __table_na  me2 ( Partition   Table )
          | | ____   partition_  name = partition_  value ( partition
   directory )
          | | | ____   data ( file )

          | | ____  < __schema__ > ( file )

          | ____  < __resource  s__ >

                  | ___table_r  esource_na  me  ( table   resource )
                  | | ____ < __ref__ >

                  | ___   file_resou  rce_name ( file   resource )
```

schema Example,as follows.

```
 Non - partiton   table :
 project = project_na  me
 table = table_name
 columns = col1 : BIGINT , col2 : DOUBLE , col3 : BOOLEAN , col4 :
 DATETIME , col5 : STRING
 Partition   table :
 project = project_na  me
 table = table_name
 columns = col1 : BIGINT , col2 : DOUBLE , col3 : BOOLEAN , col4 :
 DATETIME , col5 : STRING
 partitions = col1 : BIGINT , col2 : DOUBLE , col3 : BOOLEAN , col4 :
 DATETIME , col5 : STRING
 Note :
 Currently ,  the   following   five   data   formats   are
 supported :  bigint , double , boolean , datetime , string ,  which
 correspond   to   the   data   types   in   java : - long , double ,
 boolean , java . util . Date , java . lang . String .
```

data Example,as follows.

```
 1 , 1 . 1 , true , 2015 - 06 - 04   11 : 22 : 42   896 , hello
 world
\ N ,\ N ,\ N ,\ N ,\ N
 Note :
```

```
 The   time   format   is   accurate   to   the   millisecon  d
 level   and   all   types   are   represente  d   NULL   by  '\ N '.
```

Note:

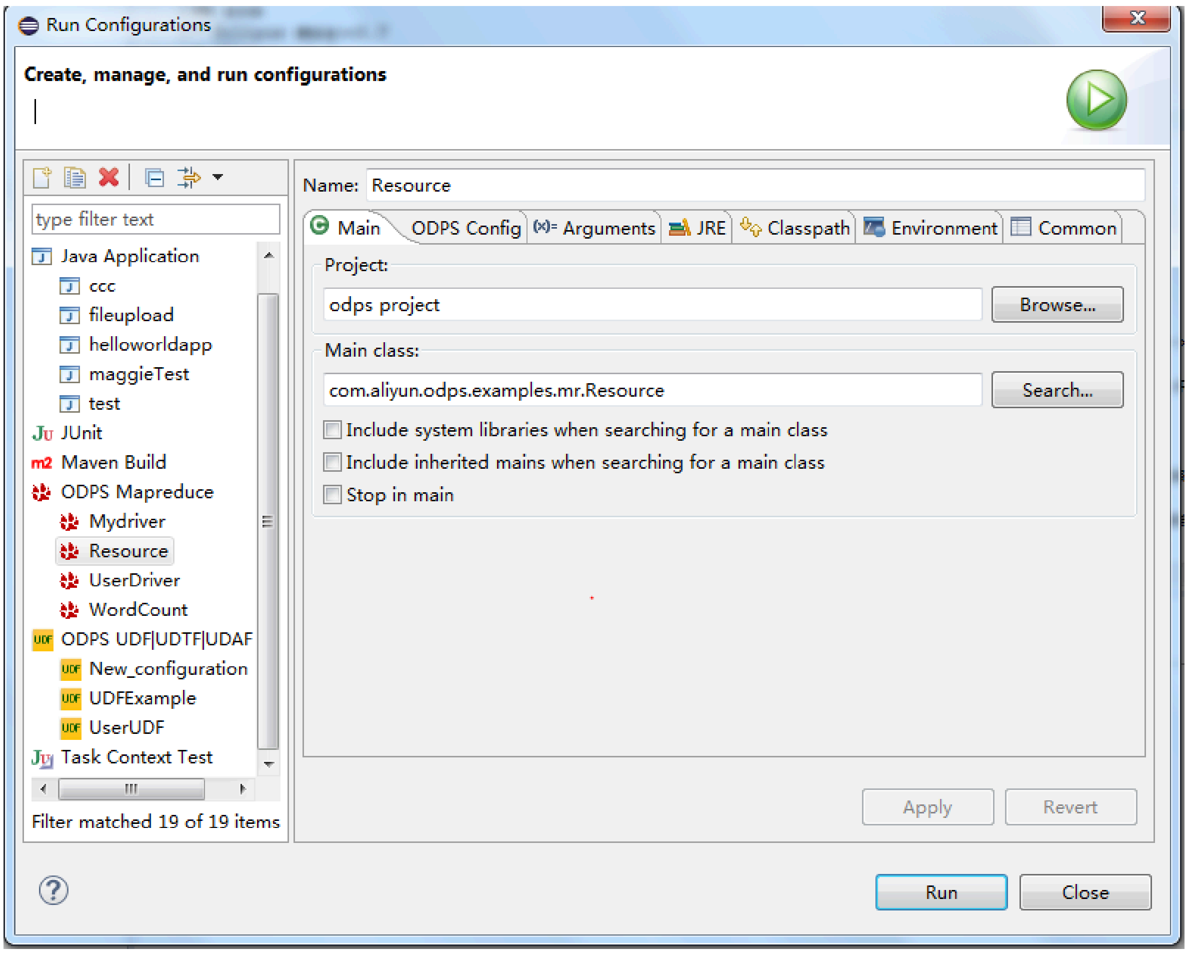
- If MapReduce program runs in the local, the default is to search corresponding tables or resources from the warehouse directory. If the tables or resources do not exist, corresponding data will be downloaded from the server and saved in warehouse. Then run MapReduce in the local.
- After running MapReduce is finished, refresh the warehouse directory to view the generated result.

3.4 UDF

This section describes how to develop UDF with the Eclipse plug-in and how to run UDF on local. The preparation and implement process is similar to UDF. You can see the example of UDF. MaxCompute Eclipse plug-in provides two methods to run UDF: Menu Bar and run by right-clicking it.

Run UDF using Menu Bar

1. Select Run > > Run Configurations··· from the menu bar and the following dialog box appears,as follows.



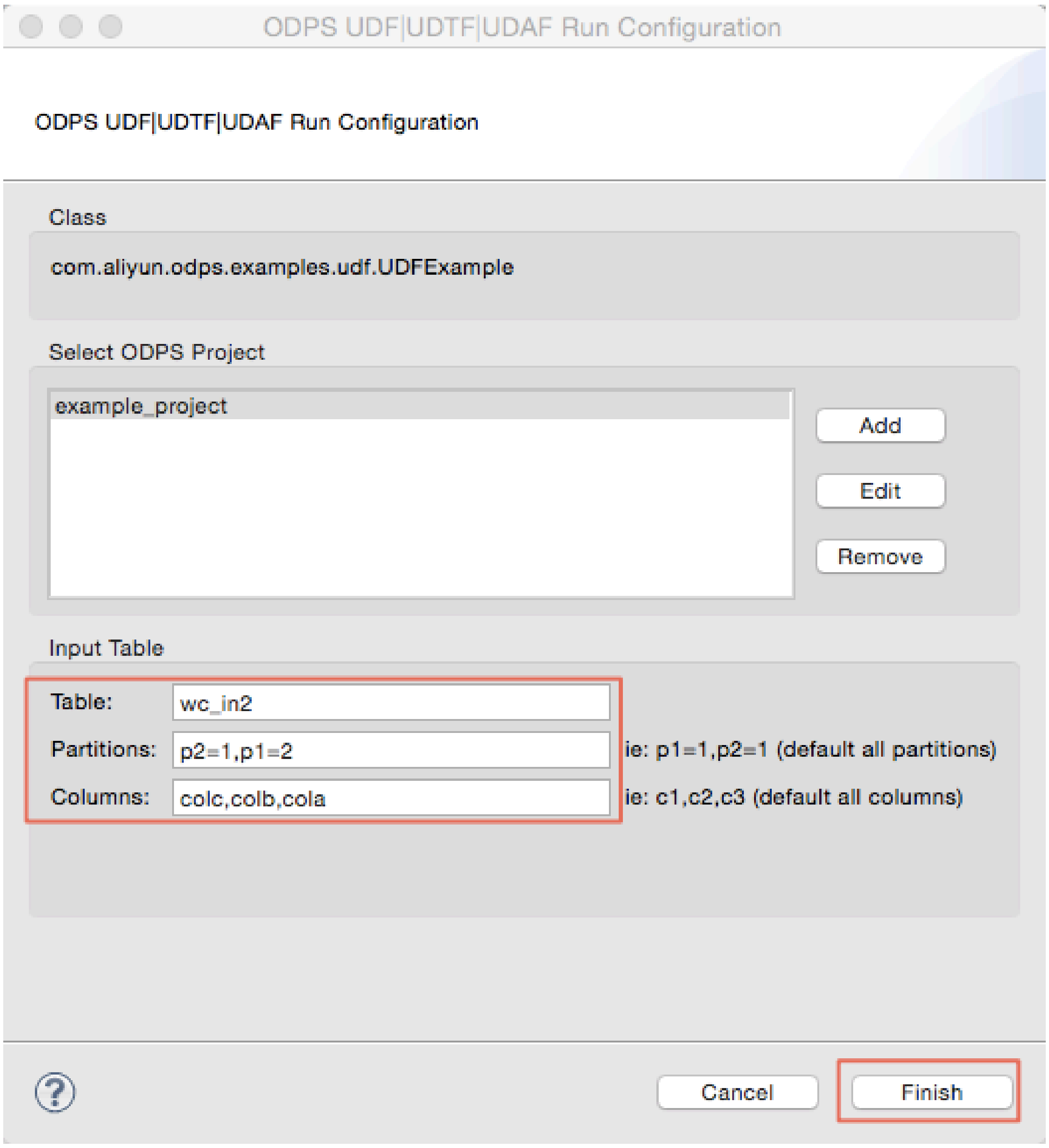
Name: Resource

Project: odps project

Main class: com.aliyun.odps.examples.mr.Resource

☐ Include system libraries when searching for a main class
☐ Include inherited mains when searching for a main class
☐ Stop in main

2. You can create a new Run Configuration. Select the UDF class and type to be executed, select MaxCompute Project and enter the information of input table. Aa follows.



ODPS UDF|UDTF|UDAF Run Configuration

ODPS UDF|UDTF|UDAF Run Configuration

Class

com.aliyun.odps.examples.udf.UDFExample

Select ODPS Project

example_project

Input Table

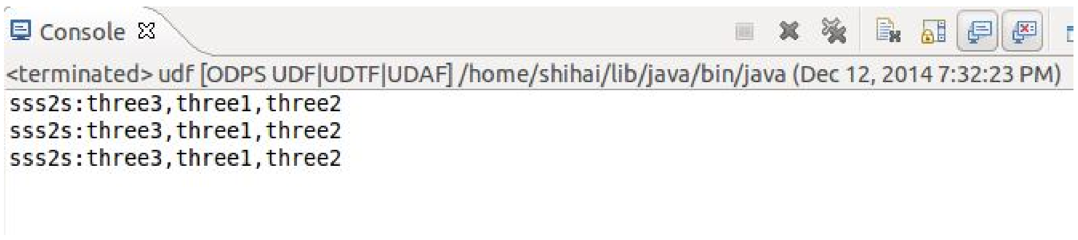
Table: wc_in2

Partitions: p2=1,p1=2 ie: p1=1,p2=1 (default all partitions)

Columns: colc,colb,cola ie: c1,c2,c3 (default all columns)

In the preceding configuration, `Table` indicates the input table of UDF. `Partitions` indicates the partitions from which the data is read, separated by commas. `Columns` indicates the columns, which are considered as the parameters of UDF to be introduced. The columns are separated by commas.

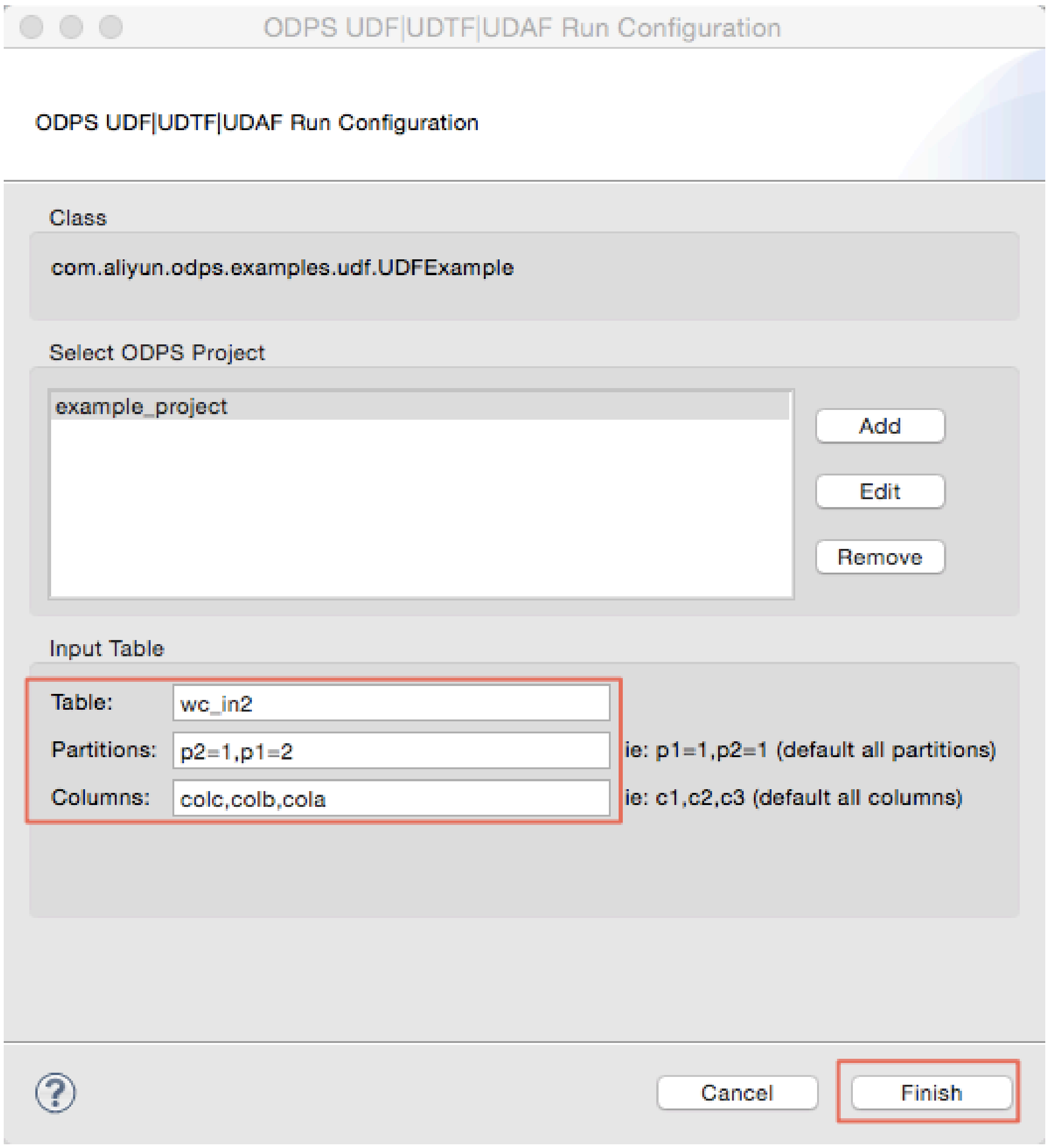
3. Click Run to run the program and the running result is displayed in the console,as follows.



```
<terminated> udf [ODPS UDF|UDTF|UDAF] /home/shihai/lib/java/bin/java (Dec 12, 2014 7:32:23 PM)
sss2s:three3,three1,three2
sss2s:three3,three1,three2
sss2s:three3,three1,three2
```

Run by right-clicking

1. Select a udf.java file (such as UDFExample.java) and right-click it. Then select Run As > > Run UDF|UDAF|UDTF:



ODPS UDF|UDTF|UDAF Run Configuration

Class

com.aliyun.odps.examples.udf.UDFExample

Select ODPS Project

example_project

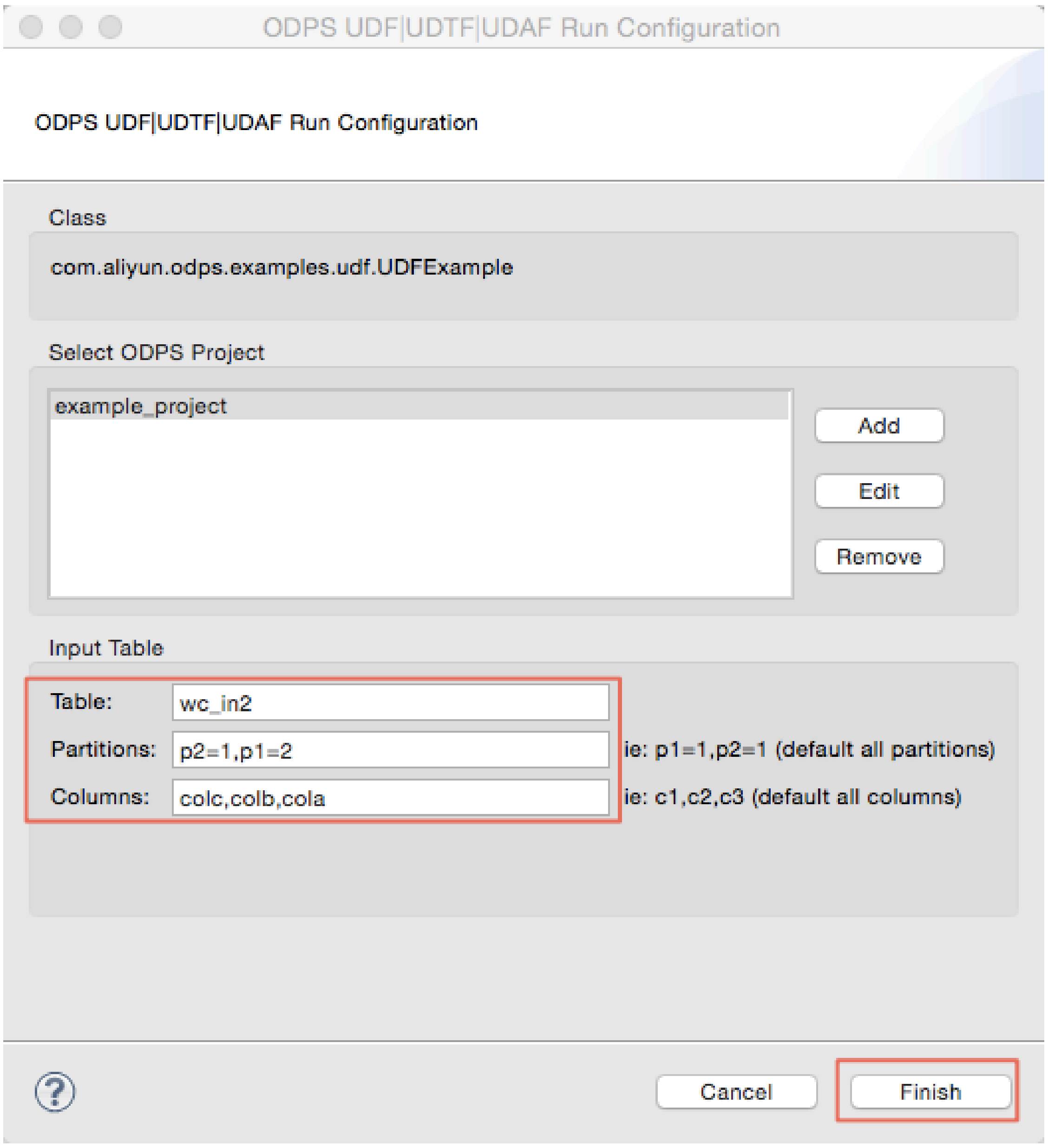
Input Table

Table: wc_in2

Partitions: p2=1,p1=2 ie: p1=1,p2=1 (default all partitions)

Columns: colc,colb,cola ie: c1,c2,c3 (default all columns)

2. The configuration information is shown as follows.



ODPS UDF|UDTF|UDAF Run Configuration

Class

com.aliyun.odps.examples.udf.UDFExample

Select ODPS Project

example_project

Input Table

Table:	wc_in2	
Partitions:	p2=1,p1=2	ie: p1=1,p2=1 (default all partitions)
Columns:	colc,colb,cola	ie: c1,c2,c3 (default all columns)

In the preceding configuration, `Table` indicates the input table of UDF. `Partitions` indicates the partitions from which the data is read, separated by commas. `Columns` indicates the columns, which are considered as the parameters of UDF to be introduced. The columns are separated by commas.

3. Click Finish to run UDF and get the output result.

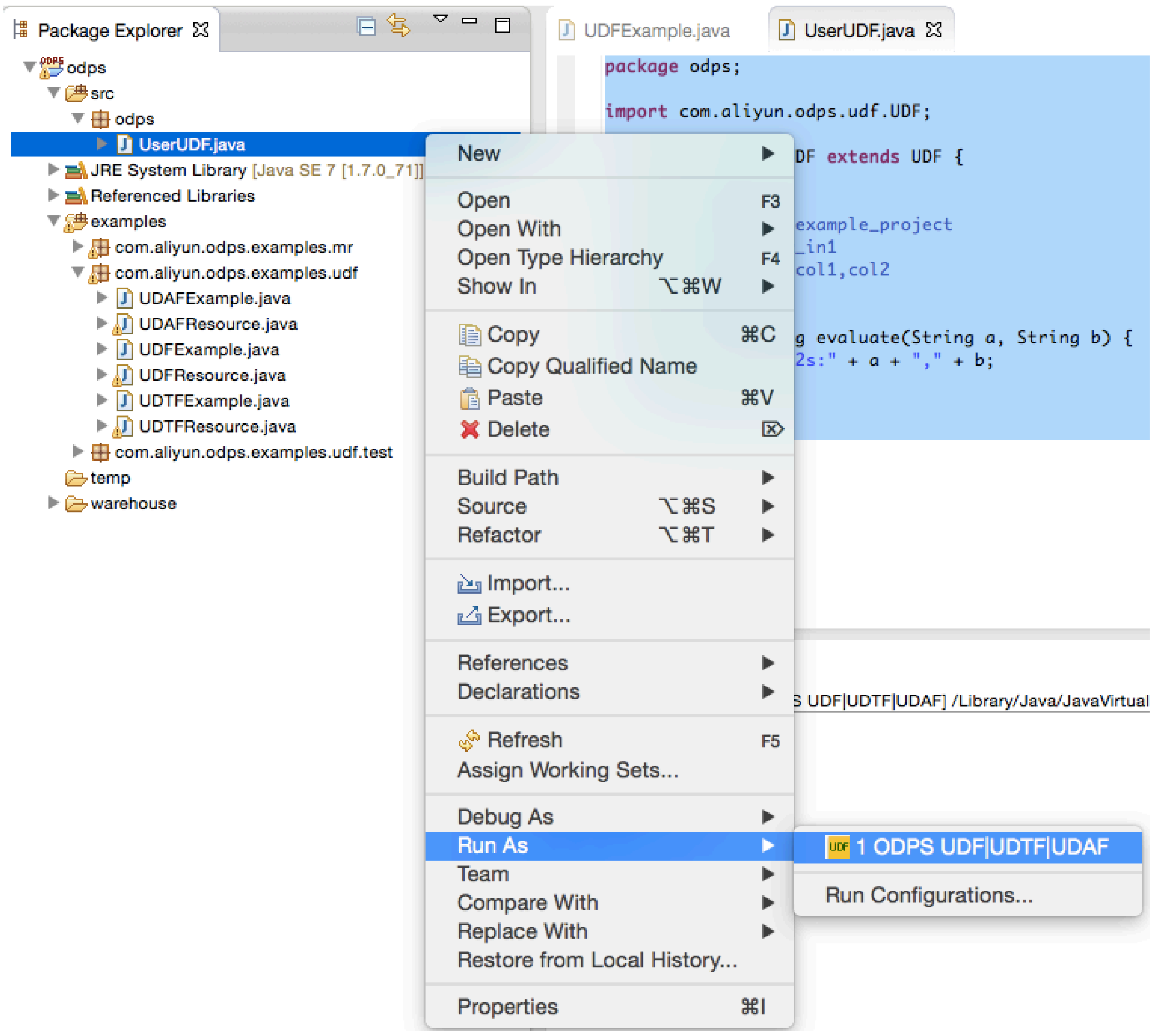
Running customized UDF program

Right-click a project and select New > >UDF (or select the menu bar File > > New > > UDF).

Enter the UDF class name and click Finish. Generate a Java file in corresponding src directory with the same name as this UDF class. Edit this java file as follows.

```
package   odps ;
import   com . aliyun . odps . udf . UDF ;
public   class   UserUDF   extends   UDF  {

      *  project :  example_pr  oject
      *  table :  wc_in1
      *  columns :  col1 , col2


      public   String   evaluate ( String   a ,  String   b ) {
        return  " ss2s :" +  a  + "," +  b ;
     }
```

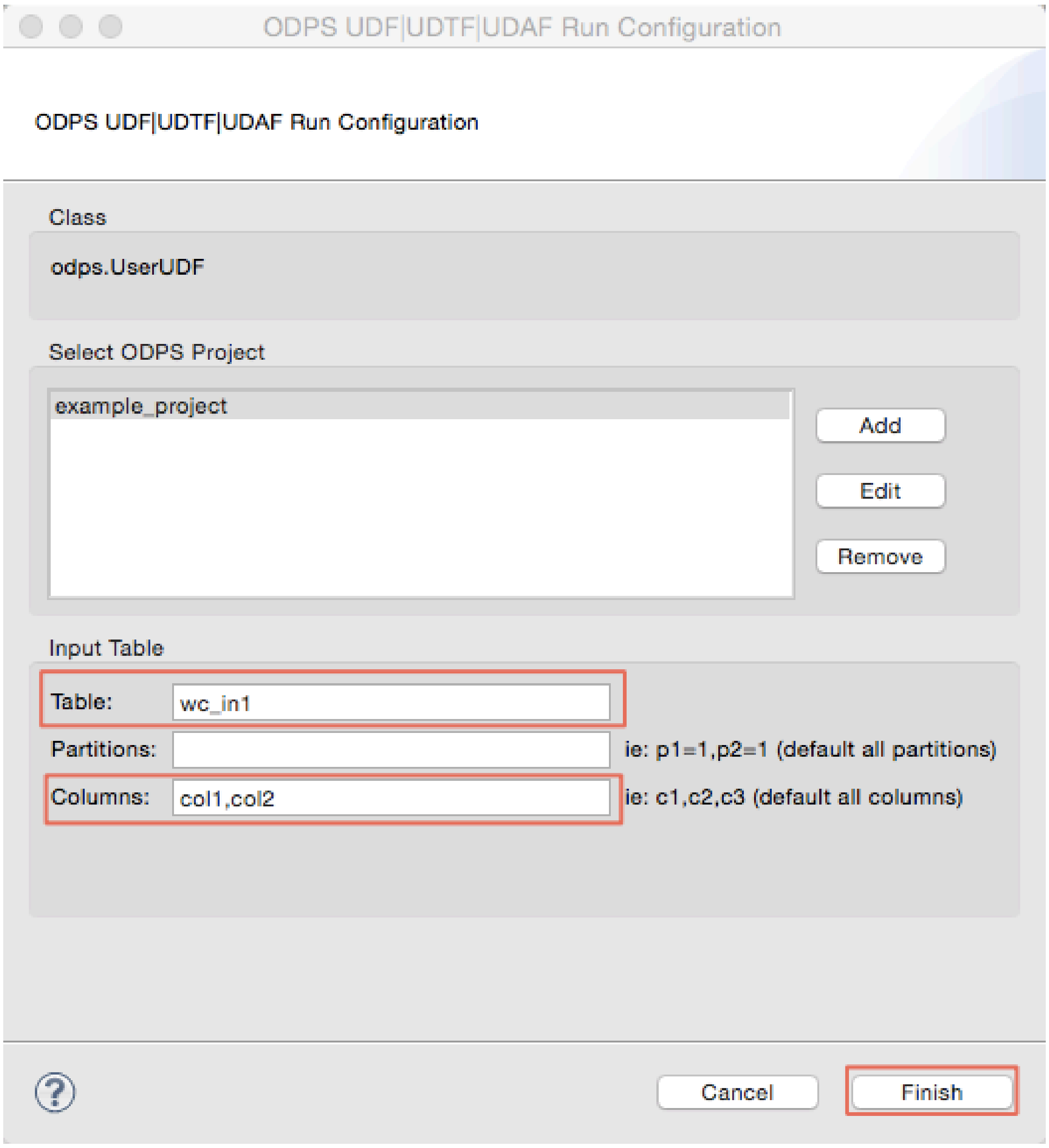
Right-click this java file (such as UserUDF.java) and select Run As -> ODPS UDF|UDTF|UDAF:



```
package odps;

import com.aliyun.odps.udf.UDF;

DF extends UDF {

example_project
_in1
col1,col2

g evaluate(String a, String b) {
2s:" + a + "," + b;
```

```
S UDF|UDTF|UDAF] /Library/Java/JavaVirtual
```

Configure the following dialog box,as follows.


ODPS UDF|UDTF|UDAF Run Configuration

ODPS UDF|UDTF|UDAF Run Configuration

Class

odps.UserUDF

Select ODPS Project

example_project

Input Table

Table: wc_in1

Partitions: ie: p1=1,p2=1 (default all partitions)

Columns: col1,col2 ie: c1,c2,c3 (default all columns)

Click Finish to get the result:

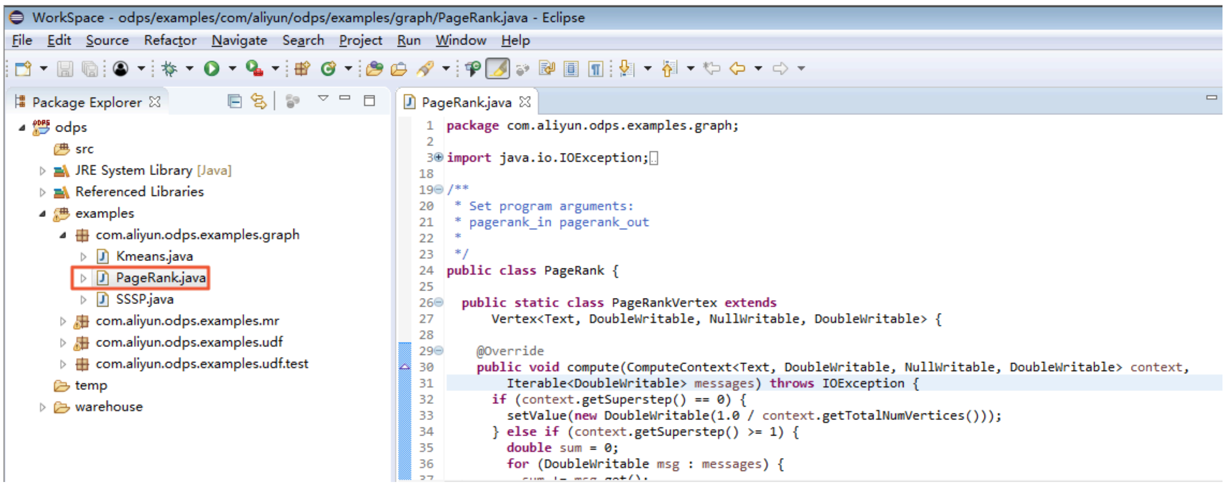
```
ss2s : A1 , A2
ss2s : A1 , A2
ss2s : A1 , A2
ss2s : A1 , A2
```

Only the operation instance of UDF is described in this section, and the way of UDTF operating is basically similar to the UDF.

3.5 Graph

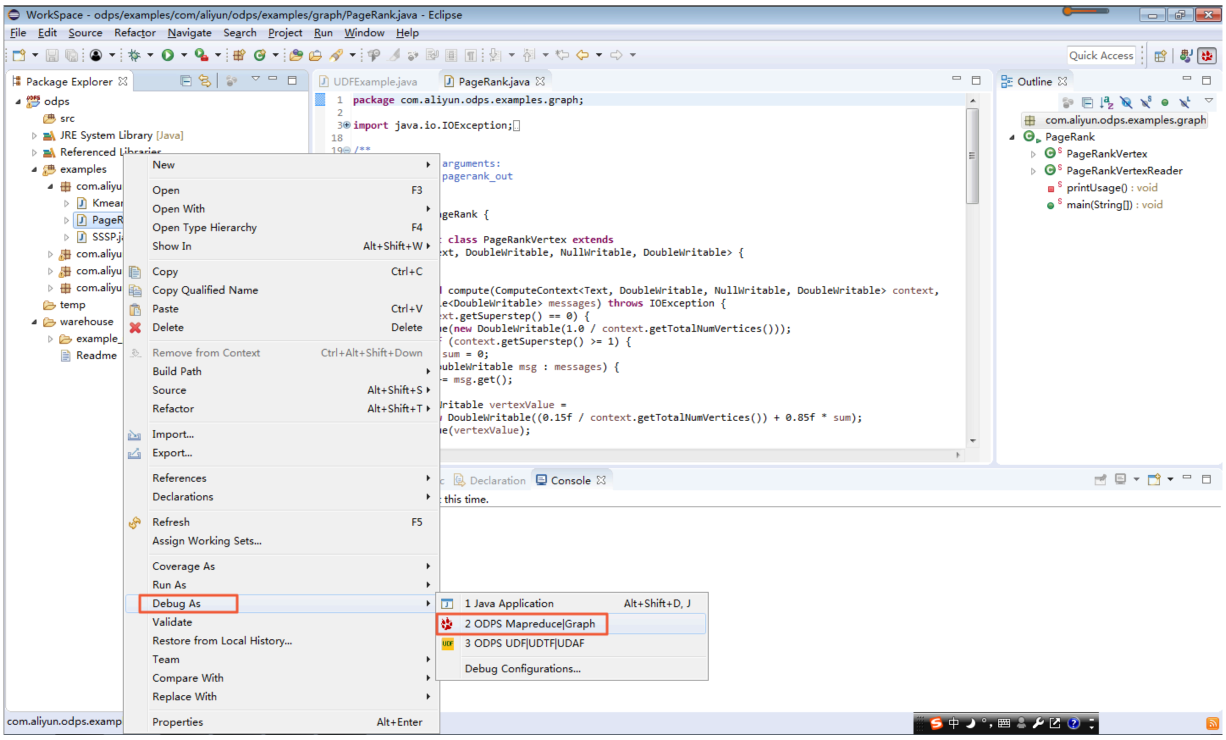
After creating a MaxCompute project, you can write Graph program and complete the local debugging according to the following steps.

In this example, `PageRank . java` provided by the plug-in is selected to complete the debugging. Select `PageRank . java` in examples, as follows.

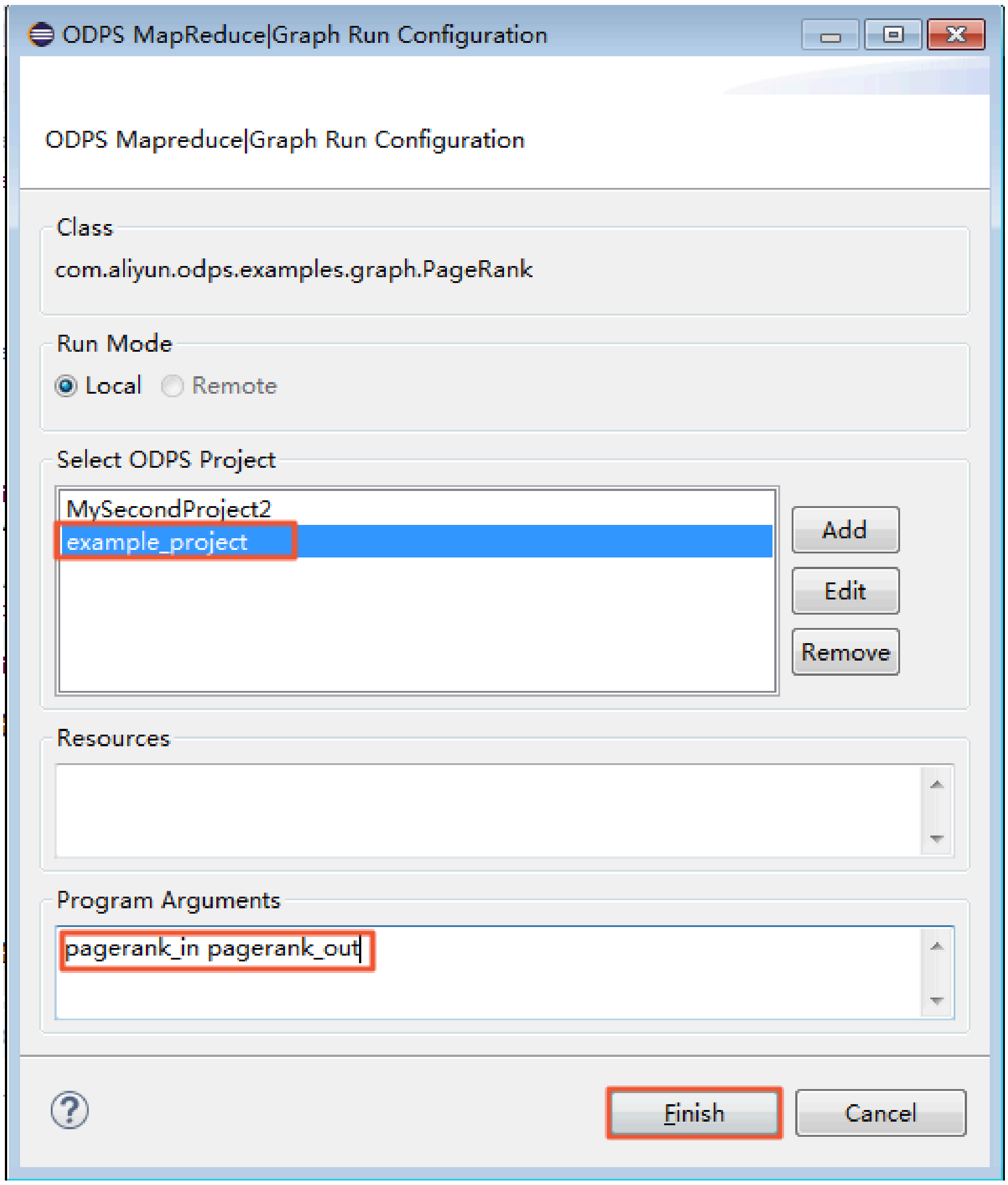

WorkSpace - odps/examples/com/aliyun/odps/examples/graph/PageRank.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
odps
src
JRE System Library [Java]
Referenced Libraries
examples
com.aliyun.odps.examples.graph
Kmeans.java
PageRank.java
SSSP.java
com.aliyun.odps.examples.mr
com.aliyun.odps.examples.udf
com.aliyun.odps.examples.udf.test
temp
warehouse
PageRank.java

```
package com.aliyun.odps.examples.graph;

import java.io.IOException;
/**
 * Set program arguments:
 * pagerank_in pagerank_out
 *
 */
public class PageRank {

  public static class PageRankVertex extends
      Vertex<Text, DoubleWritable, NullWritable, DoubleWritable> {

    @Override
    public void compute(ComputeContext<Text, DoubleWritable, NullWritable, DoubleWritable> context,
        Iterable<DoubleWritable> messages) throws IOException {
      if (context.getSuperstep() == 0) {
        setValue(new DoubleWritable(1.0 / context.getTotalNumVertices()));
      } else if (context.getSuperstep() >= 1) {
        double sum = 0;
        for (DoubleWritable msg : messages) {
```

Right-click and select Debug As > ODPS MapReduce|Graph,as follows.



The dialog box appears, and configure it as follows.


ODPS MapReduce|Graph Run Configuration

ODPS Mapreduce|Graph Run Configuration

Class

com.aliyun.odps.examples.graph.PageRank

Run Mode

(●) Local ( ) Remote

Select ODPS Project

MySecondProject2

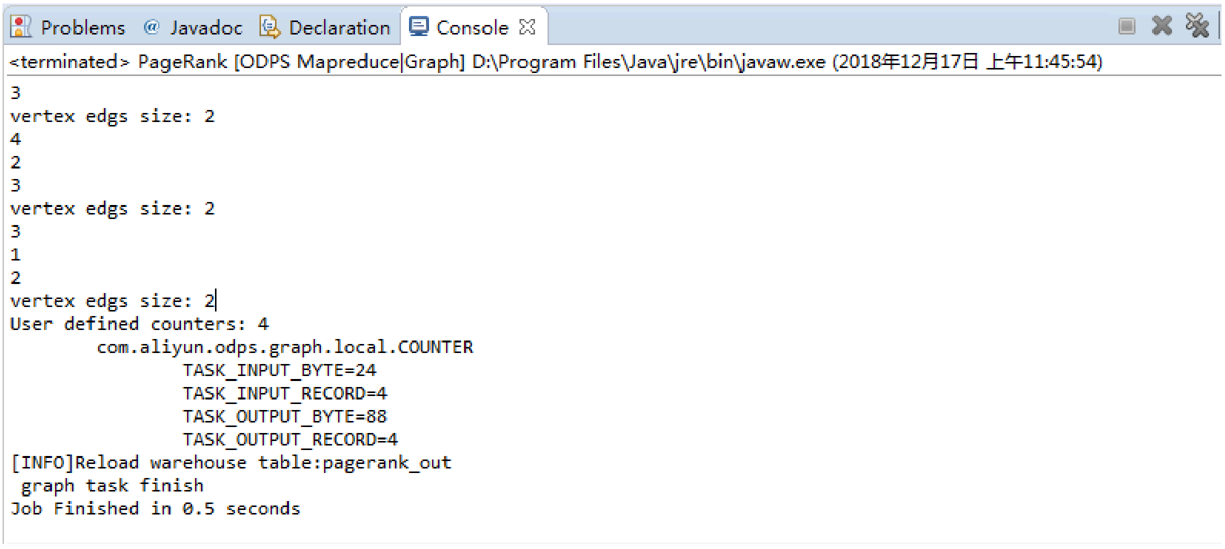
example_project

Resources

Program Arguments

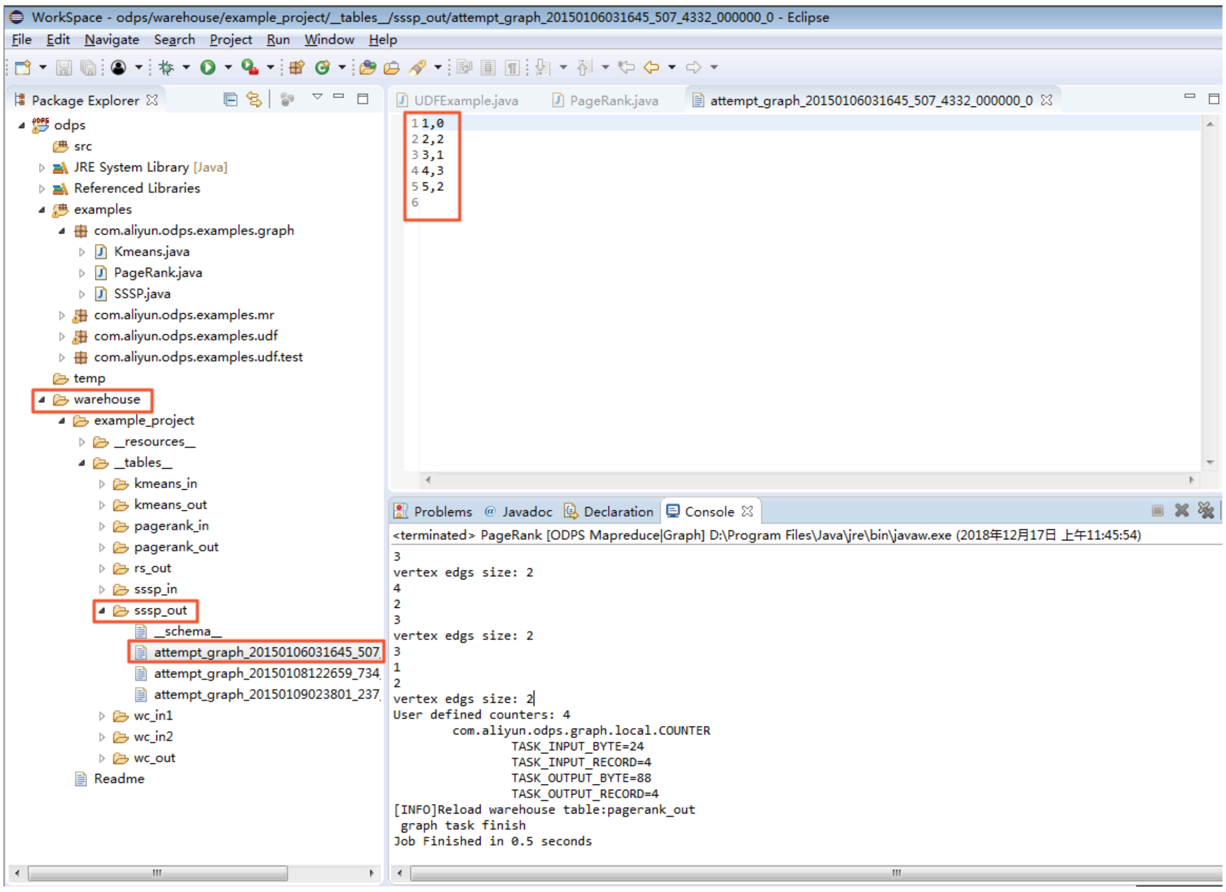
pagerank_in pagerank_out

View the running result,as follows.



```
<terminated> PageRank [ODPS Mapreduce|Graph] D:\Program Files\Java\jre\bin\javaw.exe (2018年12月17日 上午11:45:54)
3
vertex edgs size: 2
4
2
3
vertex edgs size: 2
3
1
2
vertex edgs size: 2
User defined counters: 4
        com.aliyun.odps.graph.local.COUNTER
                TASK_INPUT_BYTE=24
                TASK_INPUT_RECORD=4
                TASK_OUTPUT_BYTE=88
                TASK_OUTPUT_RECORD=4
[INFO]Reload warehouse table:pagerank_out
 graph task finish
Job Finished in 0.5 seconds
```

You can view the computing result on the local,as follows.


WorkSpace - odps/warehouse/example_project/__tables__/sssp_out/attempt_graph_20150106031645_507_4332_000000_0 - Eclipse
File Edit Navigate Search Project Run Window Help
Package Explorer
odps
src
JRE System Library [Java]
Referenced Libraries
examples
com.aliyun.odps.examples.graph
Kmeans.java
PageRank.java
SSSP.java
com.aliyun.odps.examples.mr
com.aliyun.odps.examples.udf
com.aliyun.odps.examples.udf.test
temp
warehouse
example_project
__resources__
__tables__
kmeans_in
kmeans_out
pagerank_in
pagerank_out
rs_out
sssp_in
sssp_out
__schema__
attempt_graph_20150106031645_507
attempt_graph_20150108122659_734
attempt_graph_20150109023801_237
wc_in1
wc_in2
wc_out
Readme
UDFExample.java
PageRank.java
attempt_graph_20150106031645_507_4332_000000_0
Problems
Javadoc
Declaration
Console

```
1,0
2,2
3,1
4,3
5,2
```

```
<terminated> PageRank [ODPS Mapreduce|Graph] D:\Program Files\Java\jre\bin\javaw.exe (2018年12月17日 上午11:45:54)
3
vertex edgs size: 2
4
2
3
vertex edgs size: 2
3
1
2
vertex edgs size: 2
User defined counters: 4
        com.aliyun.odps.graph.local.COUNTER
                TASK_INPUT_BYTE=24
                TASK_INPUT_RECORD=4
                TASK_OUTPUT_BYTE=88
                TASK_OUTPUT_RECORD=4
[INFO]Reload warehouse table:pagerank_out
 graph task finish
Job Finished in 0.5 seconds
```

After the debugging is complete, you can package the program and upload it to MaxCompute as a Jar resource. Then submit Graph job.

Note:

- For the package process, see *MapReduce Eclipse Plug-in Introduction*.

- For the structure introduction of local result, see *MapReduce Eclipse Plug-in Introduction*.
- For the detailed introduction of uploading Jar resource, see Add Resource in *Basic Introduction* .
- For submitting the Graph job, see *Graph Function*.

4 Downloads

This document provides you with the download address of the relevant tools and plugins.

- SDK Downloads: Maven users can search odps-sdk from Maven library to get different versions of the Java SDK.
- MaxCompute client: *Click here* to download the new version of MaxCompute client.
- *Eclipse plugin*
- *IntelliJ plugin*, *Studio*
- *JDBC*
- *PHP SDK*