Alibaba Cloud DataWorks

Quick Start

Issue: 20190117

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.
5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectu

al property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion , or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos , marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).

6. Please contact Alibaba Cloud directly if you discover any errors in this document.

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

Contents

1 Instructions

This topic will guide you through data development and O&M.

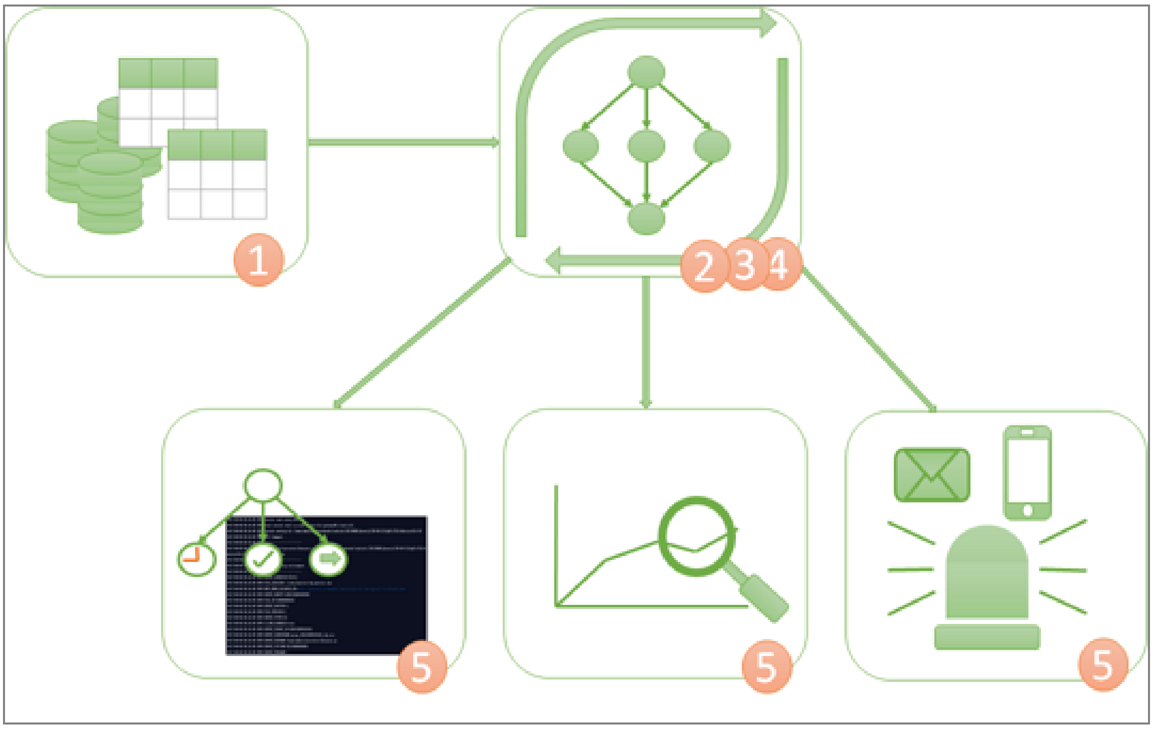
Note:

If you are using DataWorks for the first time, make sure you have completed all procedures listed in the *preparation* topic, prepare accounts, project roles, project space, and so on, then enter the DataWorks Management Console, start the data development operation by clicking **enter workspace** after the corresponding project.

Typically, data development and operations on the project space of DataWorks include the following actions:

- *Step 1: Create a table and upload data*
- *Step 2: Create a Business Flow*
- *Step 3: Create a synchronization task*
- *Step 4: Scheduling and dependency settings*
- *Step 5: OM and view log troubleshooting results*

A general process is shown in the following figure:



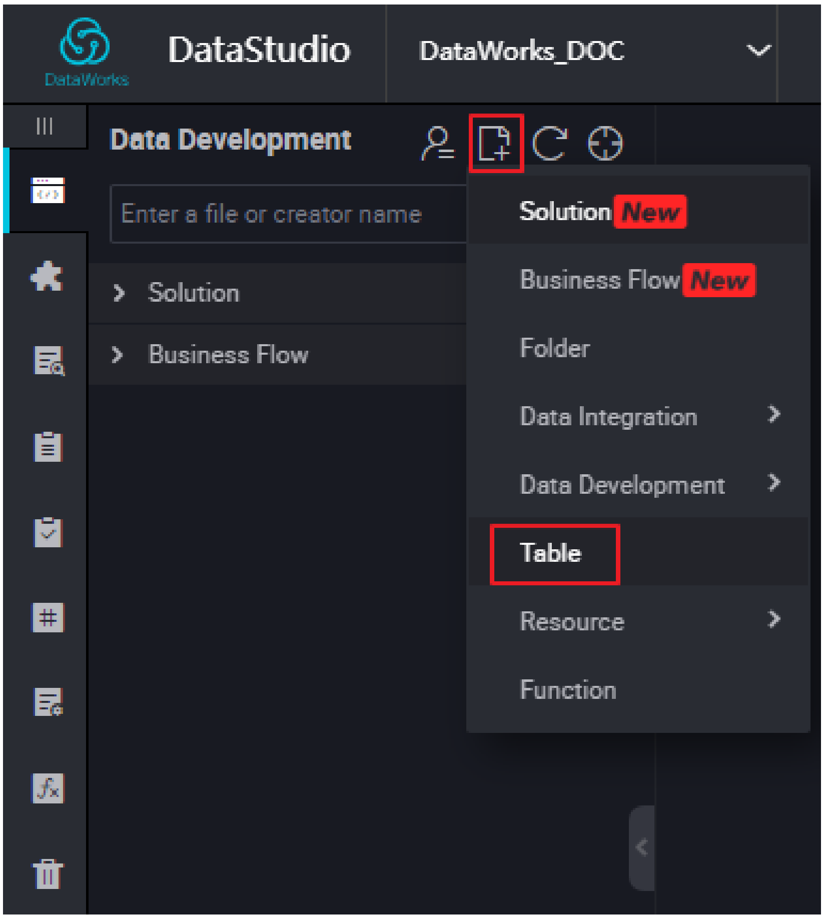
2 Step 1: Create a table and upload data

In this topic, the created tables bank_data and result_table are used as an example for creating a table and uploading data. The bank_data table stores business data, while the result_table stores the data analysis results.

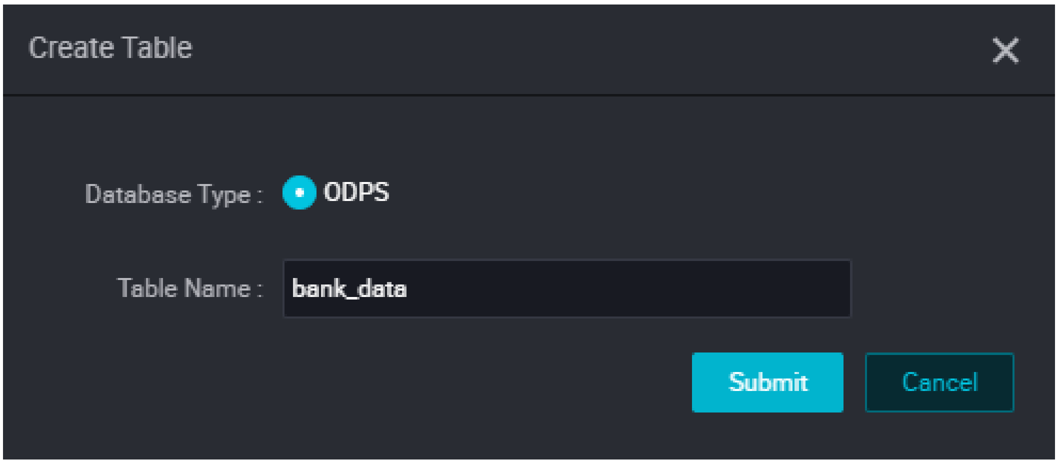
Procedure

Create a table called bank_data

1. After *Create a project* , click **Enter workspace**in the corresponding project.
2. Go to the **Data Studio (original data development)** page and select **new** > **table**.



3. Enter the table name in the **new table** dialog box.



Create Table

Database Type : ● ODPS

Table Name : bank_data

4. Click **Submit**.
5. Enter the **new table** page, and select the **DDL mode**.
6. Enter the table creation statement in the **DDL schema** dialog box, and click **build table structure**.

   For more SQL syntax for creating tables, see *creating/viewing/deleting tables*.

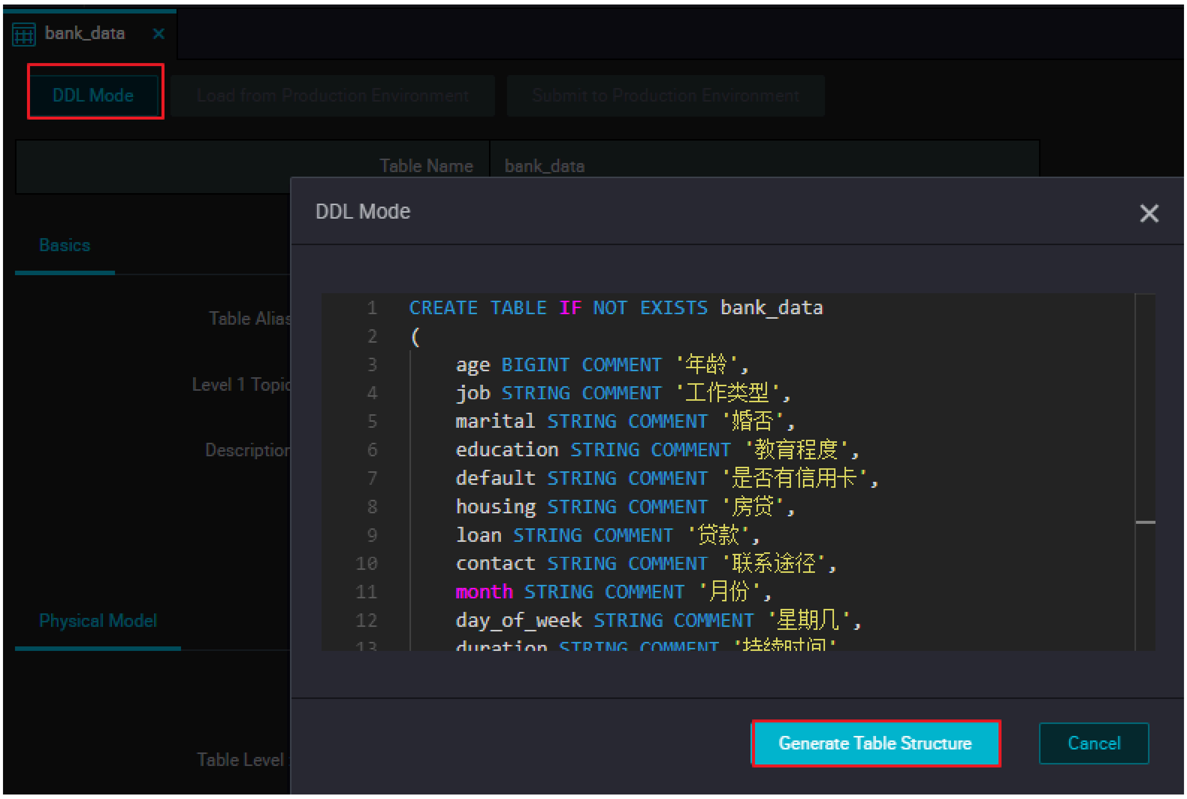


Table Name bank_data

DDL Mode

```
CREATE TABLE IF NOT EXISTS bank_data
(
    age BIGINT COMMENT '年龄',
    job STRING COMMENT '工作类型',
    marital STRING COMMENT '婚否',
    education STRING COMMENT '教育程度',
    default STRING COMMENT '是否有信用卡',
    housing STRING COMMENT '房贷',
    loan STRING COMMENT '贷款',
    contact STRING COMMENT '联系途径',
    month STRING COMMENT '月份',
    day_of_week STRING COMMENT '星期几',
    duration STRING COMMENT '持续时间'
```

The statements used for table creation in this example are as follows:

```
CREATE TABLE IF NOT EXISTS bank_data
(
 age             BIGINT COMMENT 'age',
 job             STRING COMMENT 'job type',
 marital         STRING COMMENT 'marital status',
 education       STRING COMMENT 'educational level',
 default          STRING COMMENT 'credit card ownership',
 housing          STRING COMMENT 'mortgage',
 loan             STRING COMMENT 'loan',
 contact          STRING COMMENT 'contact information',
 month            STRING COMMENT 'month',
 day_of_week      STRING COMMENT 'day of the week',
 duration         STRING COMMENT 'Duration',
 campaign         BIGINT COMMENT 'contact times during the campaign',
 pdays            DOUBLE COMMENT 'time interval from the last contact
',
 previous         DOUBLE COMMENT 'previous contact times with the
customer',
 poutcome         STRING COMMENT 'marketing result',
 emp_var_rate     DOUBLE COMMENT 'employment change rate',
 cons_price_idx   DOUBLE COMMENT 'consumer price index',
 cons_conf_idx    DOUBLE COMMENT 'consumer confidence index',
 euribor3m        DOUBLE COMMENT 'euro deposit rate',
 nr_employed      DOUBLE COMMENT 'number of employees',
 y                BIGINT COMMENT 'has time deposit or not'
);
```

7. After the table structure is generated, enter the table name and click **Submit to Production Environment**.

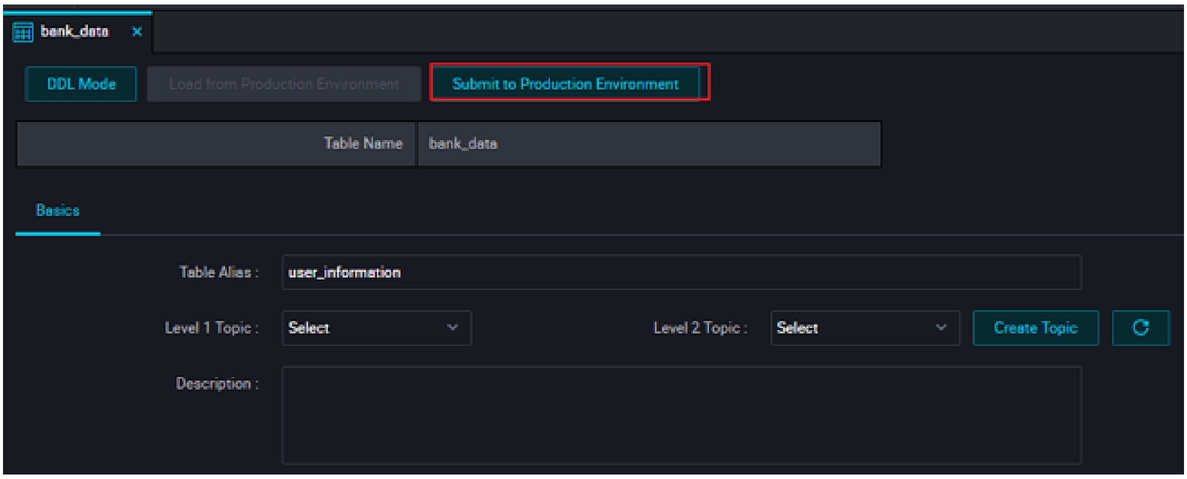

bank_data
DDL Mode
Load from Production Environment
Submit to Production Environment
Basics
Create Topic

Table Name bank_data

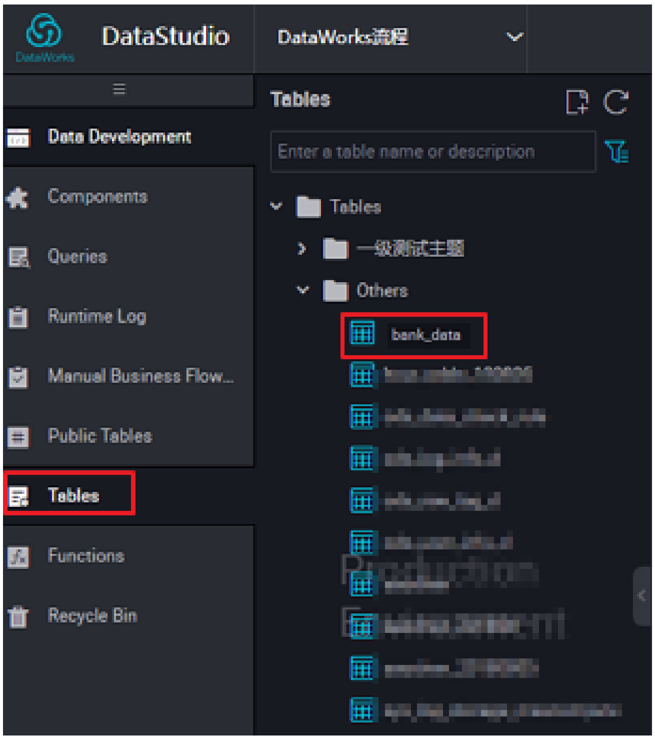
Table Alias : user_information

Level 1 Topic : Select

Level 2 Topic : Select

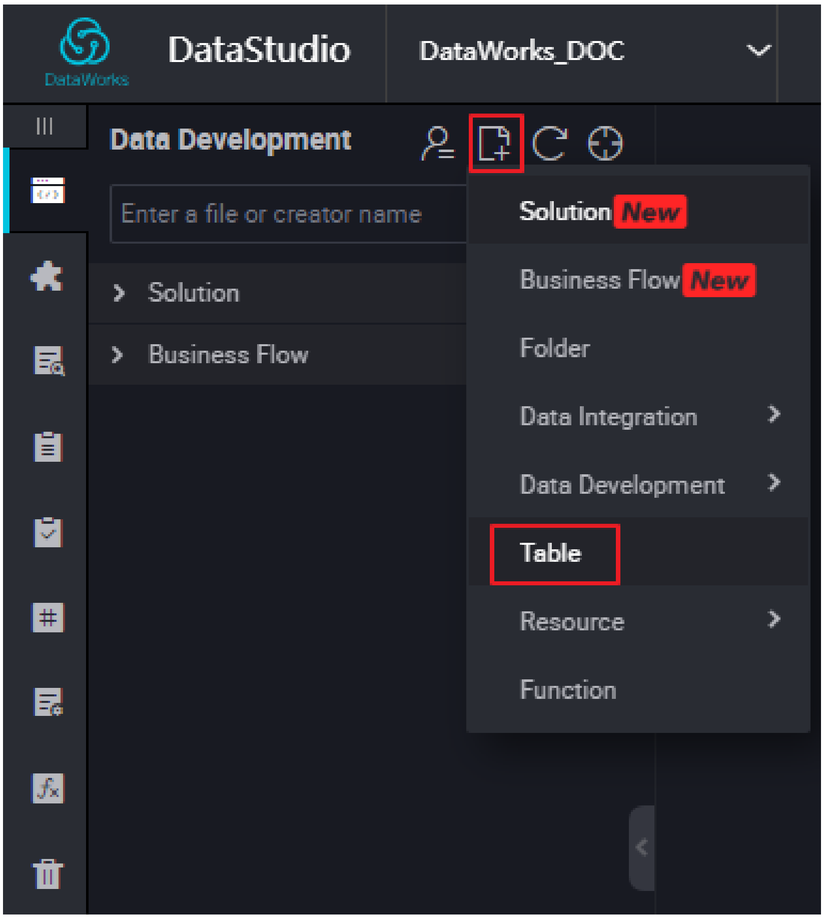
Description :

8. You can search the created table by entering the table name in the left-hand navigation **table management** to view the table information.



Create result_table

1. Go to the **DataStudio** page and select **new** > **table**.



2. Enter the table name in the **new table** dialog box and click **Submit**.

3. Enter the **new table** page and select **DDL mode**.
4. Enter the build TABLE statement in the **DDL schema** dialog box, and click **build table structure**. The following is a create table example:

```
CREATE TABLE IF NOT EXISTS result_table
(
 education   STRING COMMENT 'educational level',
 num         BIGINT COMMENT 'number of people'
);
```

5. You can search the created table by the table name in the left-hand navigation **table management** and view table information.

Upload local data to bank_data

DataWorks supports the following actions:

- Uploading data in locally stored text files to the workspace table.
- Using data integration to import business data from various data sources to the workspace.

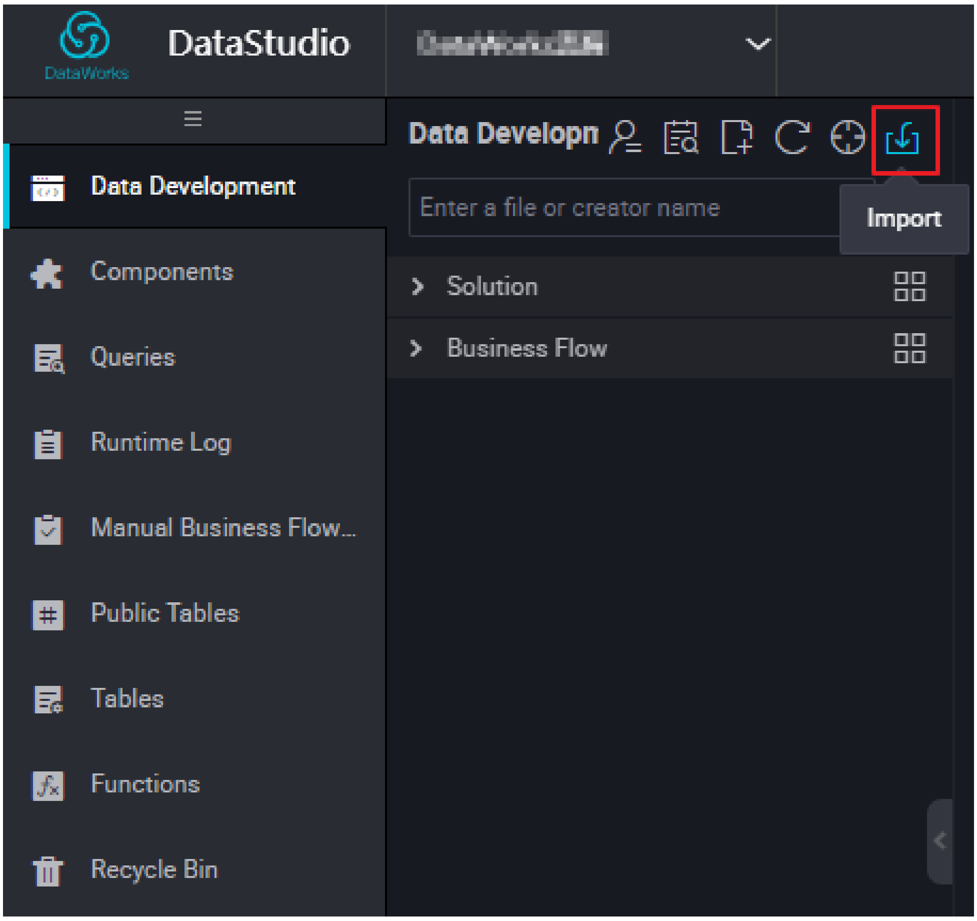
Note:

In this section, local files are used as the data source. Local text file uploads have the following restrictions:

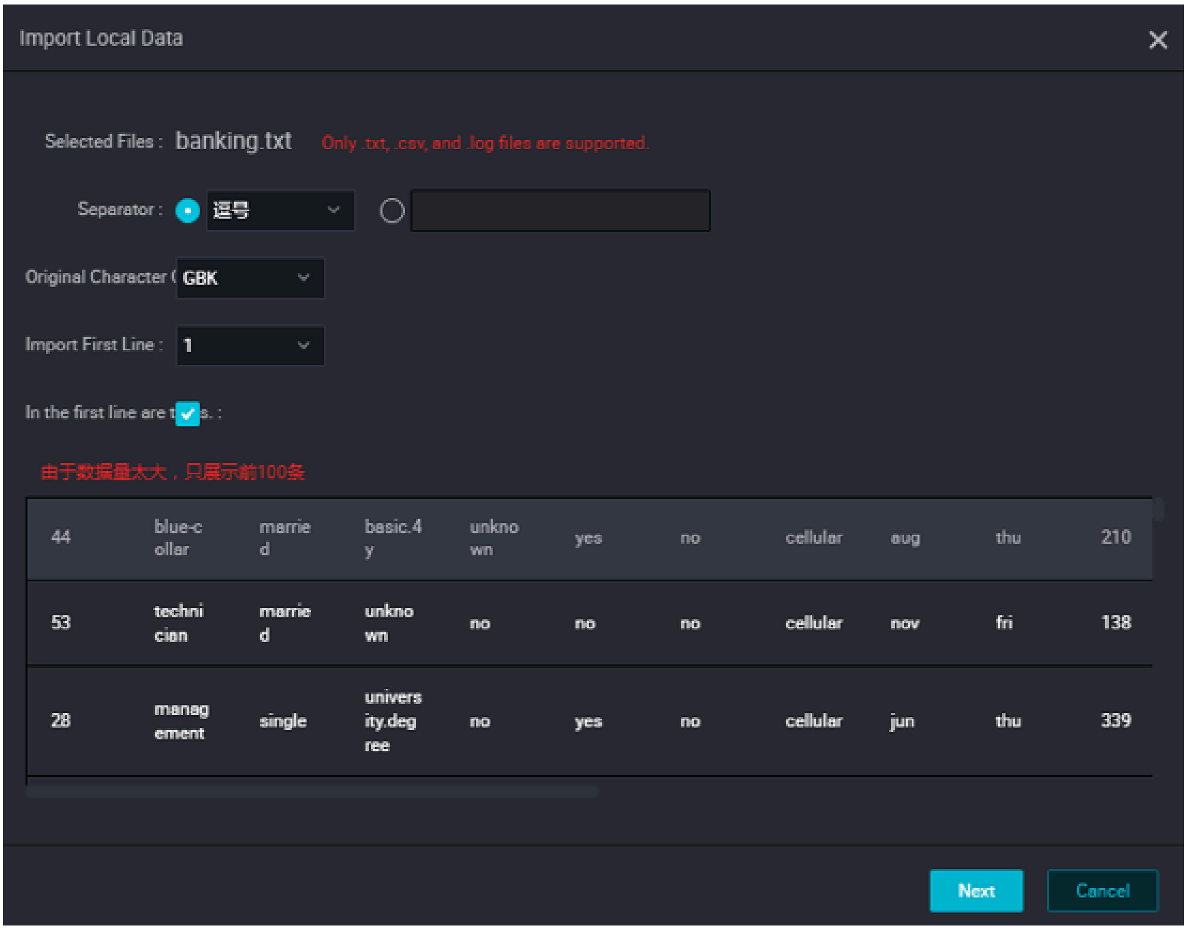
- File type: Only .txt and .csv files are supported.
- File Size: Not exceeding 10 M.
- Operation objects: Partition and non-partition tables can be imported, but Chinese partition values are not supported.

For example, import local file *banking.txt*to DataWorks, the operation is as follows:

1. Click **Import** to select **import local data**.



2. Select a local data file, configure the import information, and click **Next**,



Import Local Data

Selected Files : banking.txt Only .txt, .csv, and .log files are supported.

Separator : ● 逗号 ○

Original Character (GBK

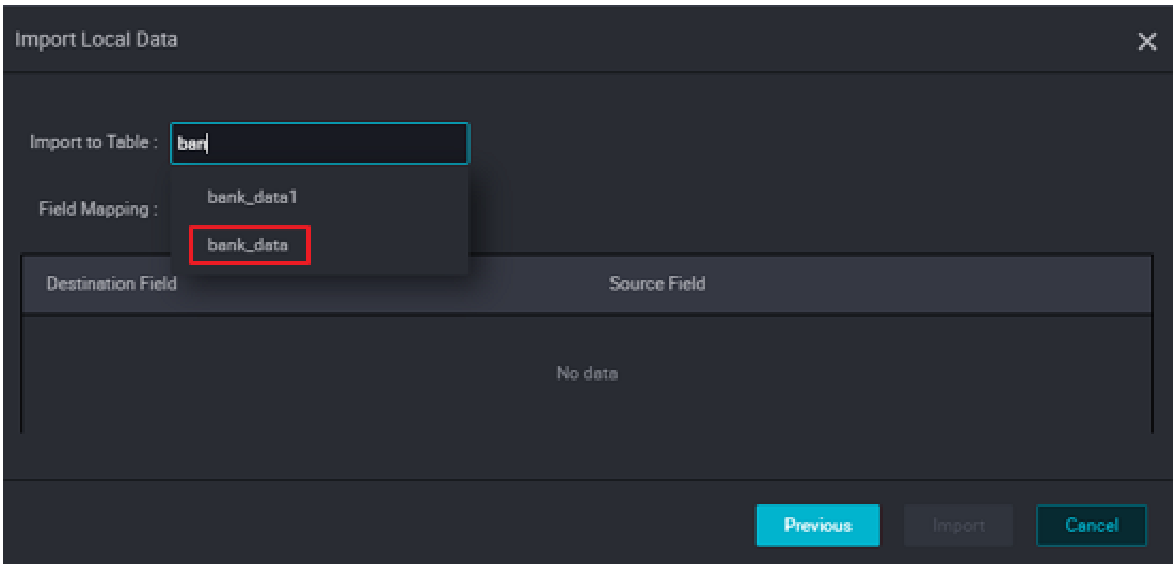
Import First Line : 1

In the first line are t☑s. :

由于数据量太大，只展示前100条

44	blue-collar	married	basic.4y	unknown	yes	no	cellular	aug	thu	210
53	technician	married	unknown	no	no	no	cellular	nov	fri	138
28	management	single	university.degree	no	yes	no	cellular	jun	thu	339

3. Enter at least two letters to search the table by name. Select the table for the data to be imported, for example, bank_data.



Import Local Data

Import to Table : ban

Field Mapping :

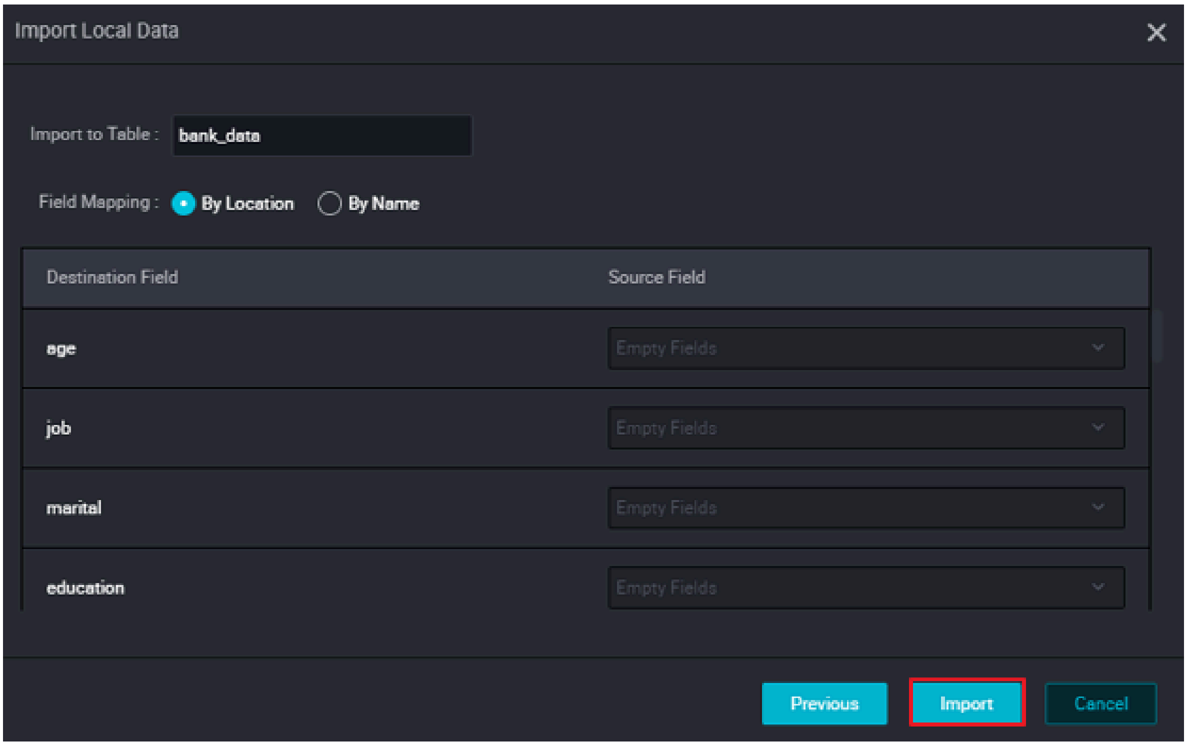
bank_data1

bank_data

Destination Field	Source Field

No data

4. Select the field matching method ("Match by Position" used in this example) and click **Import** ,



Import Local Data

Import to Table : bank_data

Field Mapping : ● By Location ○ By Name

Destination Field	Source Field
age	Empty Fields
job	Empty Fields
marital	Empty Fields
education	Empty Fields

After the file is imported, the system returns the number of lines that were successful in the data import or an exception that failed.

Other data import methods

- Create a Data Synchronization task

  This method applies to saving RDS, MySQL, SQL Server, PostgreSQL, MaxCompute, OSS , DRDs, OSS data from a variety of data sources, such as, Oracle, FTP, DM, HDFS, and MongoDB.

  For details on creating data synchronization tasks with DataWorks, see *creating a data synchronization task*.

- Local file uploads

  This file upload method is suitable for .txt and .csv files smaller than 30M , and the target supports both partition and non-partition tables , but does not support Chinese partition.

  For DataWorks local file upload, see preceding local data upload to bank_data for details.

- Upload files using tunnel command

  This method applies to local files and other resource files greater than 10M in size.

  Upload and download the data through tunnel commands provided by the *MaxCompute client*, when local data files need to be uploaded to the partition table, so they can be uploaded using the client tunnel command. See *Tunnel command actions* for details.

Next steps

You have learned how to create a table and upload data now. You can go to the next topic, which will show you how to create a work flow for further data analysis and project space computing . For more information, see *creating a business process*.

3 Step 2: Create a Business Flow

This topic uses create business flows as an example to describe how to create nodes and configure dependencies in your business flow to facilitate the design and presentation of steps and sequences of data analysis. This article briefly explains how to use the data development function to further analyze and calculate the workspace data.

DataWorks data development features support visual drag-and-drop in the business flow to complete inter-node dependency settings. The data flow and interdependencies are implemented in the form of operational business flows. Currently supports multiple task types, such as MaxCompute SQL, data synchronization, open_mr, shell, machine learning, and virtual nodes. For specific usage methods for each task type, see*Node type overview*.

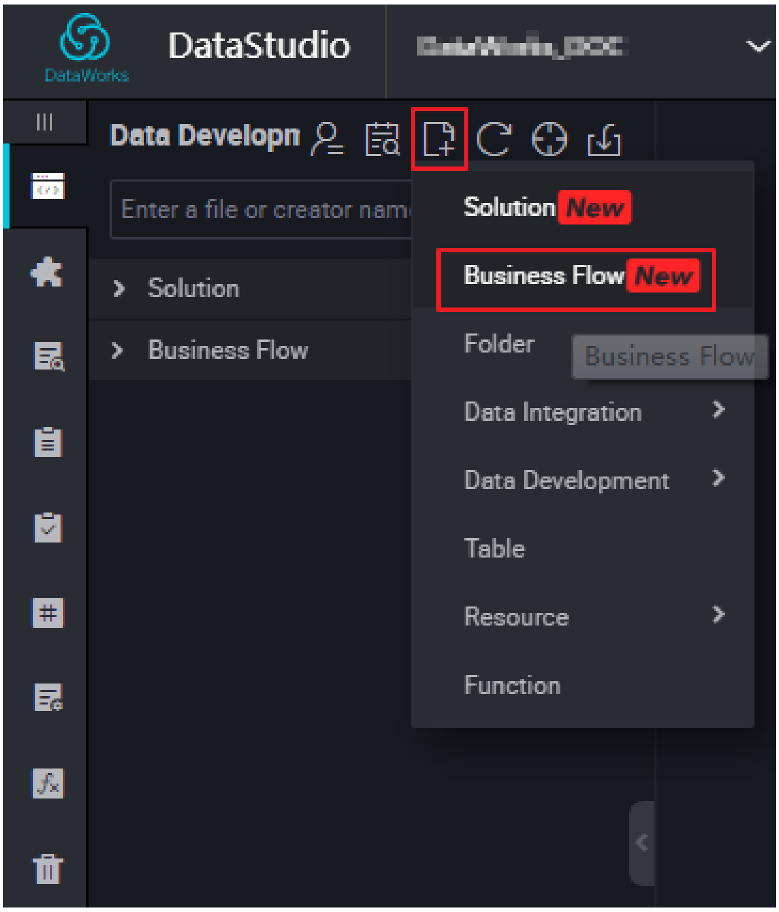
Prerequisites

Make sure you have *built the table and uploaded the data*, prepare the business data table bank_data and data in the workspace, as well as the result table.

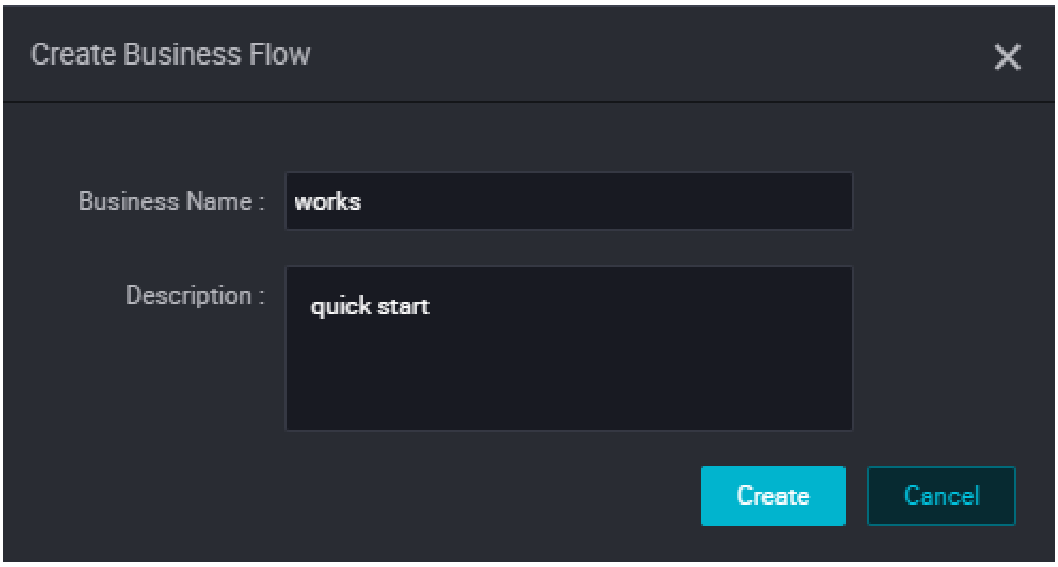
Procedure

Create a Business Flow

1. After *Create a project*, click **Enter workspace** in the corresponding project.
2. Go to the **DataStudio** page and select **create** > **business flow**.



3. Enter the name and description of the business flow.


Create Business Flow

Business Name : works

Description : quick start

Create a node and dependency on the flow canvas

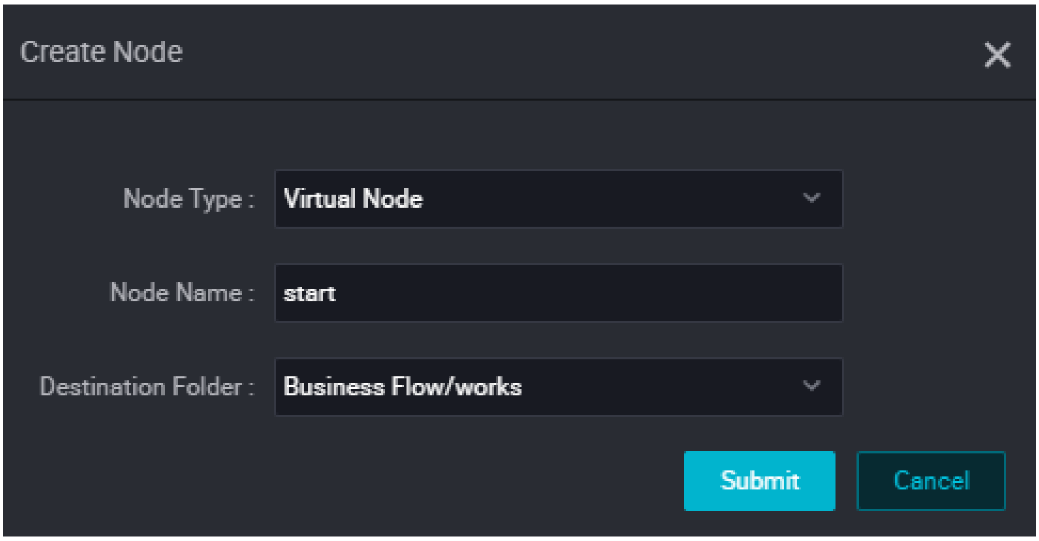
This section shows how to create a virtual node “start” and a MaxCompute SQL node “insert_data”, and to configure “insert_data” to depend on “start”.

Note:

- The virtual node is a control-type node that does not affect the data during flow operation and is only used to control O&M of downstream nodes.
- When a virtual node depends on other nodes and the status is manually set to error by the O&M personnel, downstream nodes that have not run yet cannot be triggered. This prevents further propagation of erroneous upstream data during the O&M flow. For more information, see the section on virtual nodes in *Node type overview*.
- The upstream task of a virtual node in a business flow is typically set as the root node of the project, the format of the Project root node is: Project name _ root.

We recommend you create a virtual node as the root node to control the whole workflow when designing a flow.

1. Double-click the virtual node and enter the node name start.

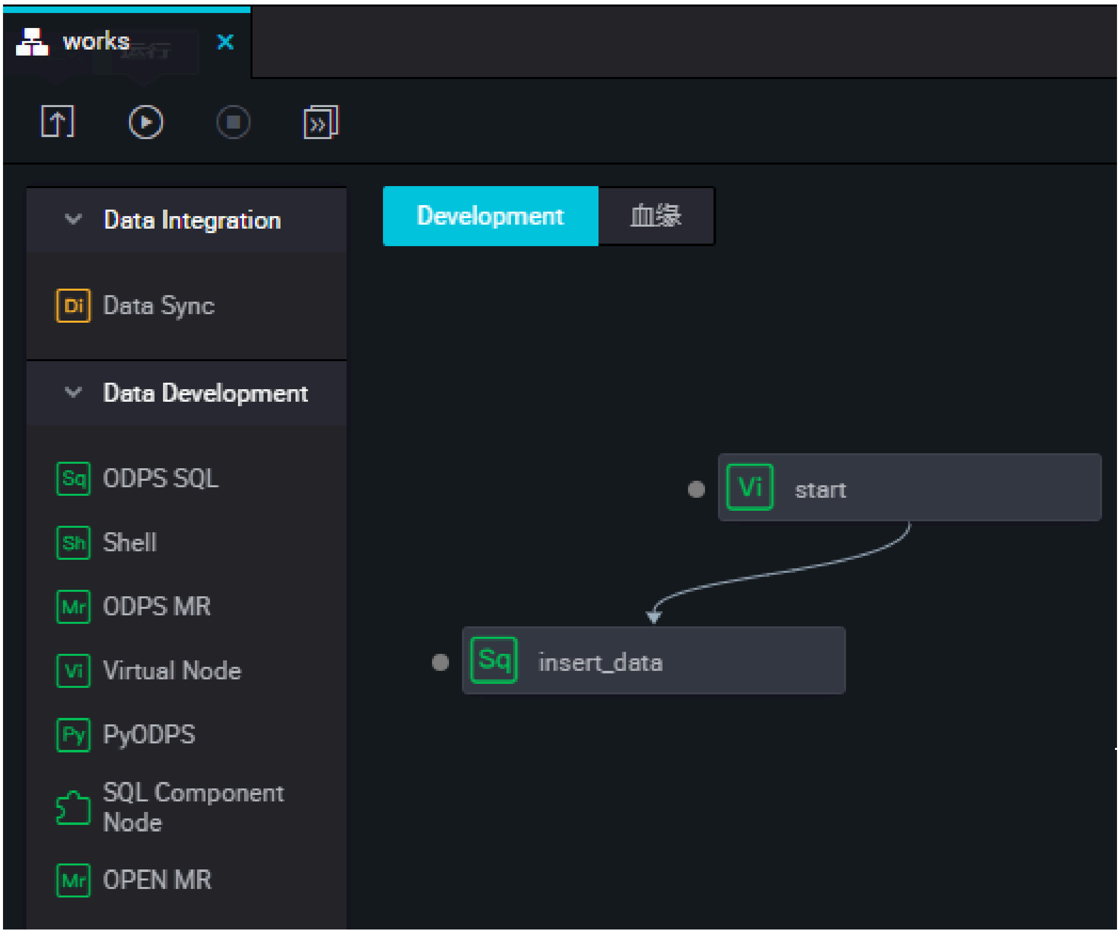


Node Type : Virtual Node

Node Name : start

Destination Folder : Business Flow/works

2. Double-click **MaxCompute SQL** to enter the node name “insert_data”.
3. Click the start node, and draw a line between start and insert_data to make insert_data a dependency on start, as shown in the following figure:



Editing code in the MaxCompute SQL Node

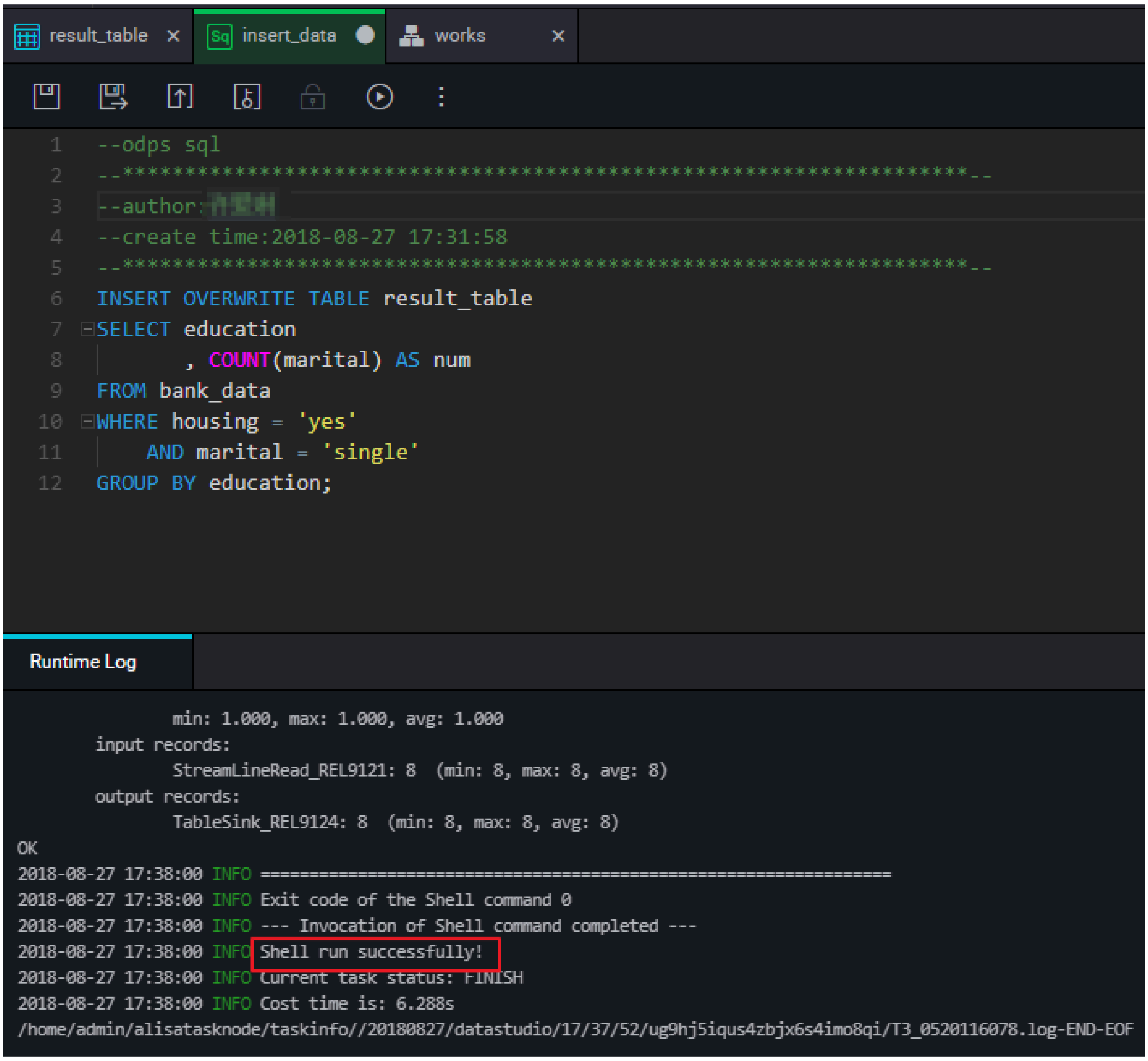
This section describes how to use SQL code in the MaxCompute SQL node **insert_data** to query the number of mortgages available for individuals with different educational backgrounds and save results for analysis or display by the following nodes.

The SQL statements are as follows. For more information about the syntax, see *MaxCompute SQL*.

```
INSERT OVERWRITE TABLE result_table  --Insert data to result_table
SELECT education
    , COUNT(marital) AS num
FROM bank_data
WHERE housing = 'yes'
    AND marital = 'single'
GROUP BY education
```

Run and debug MaxCompute SQL

1. After editing the SQL statements in the insert_data node, click **Save** to prevent code loss.
2. Click **Run** to view the operations logs and results,

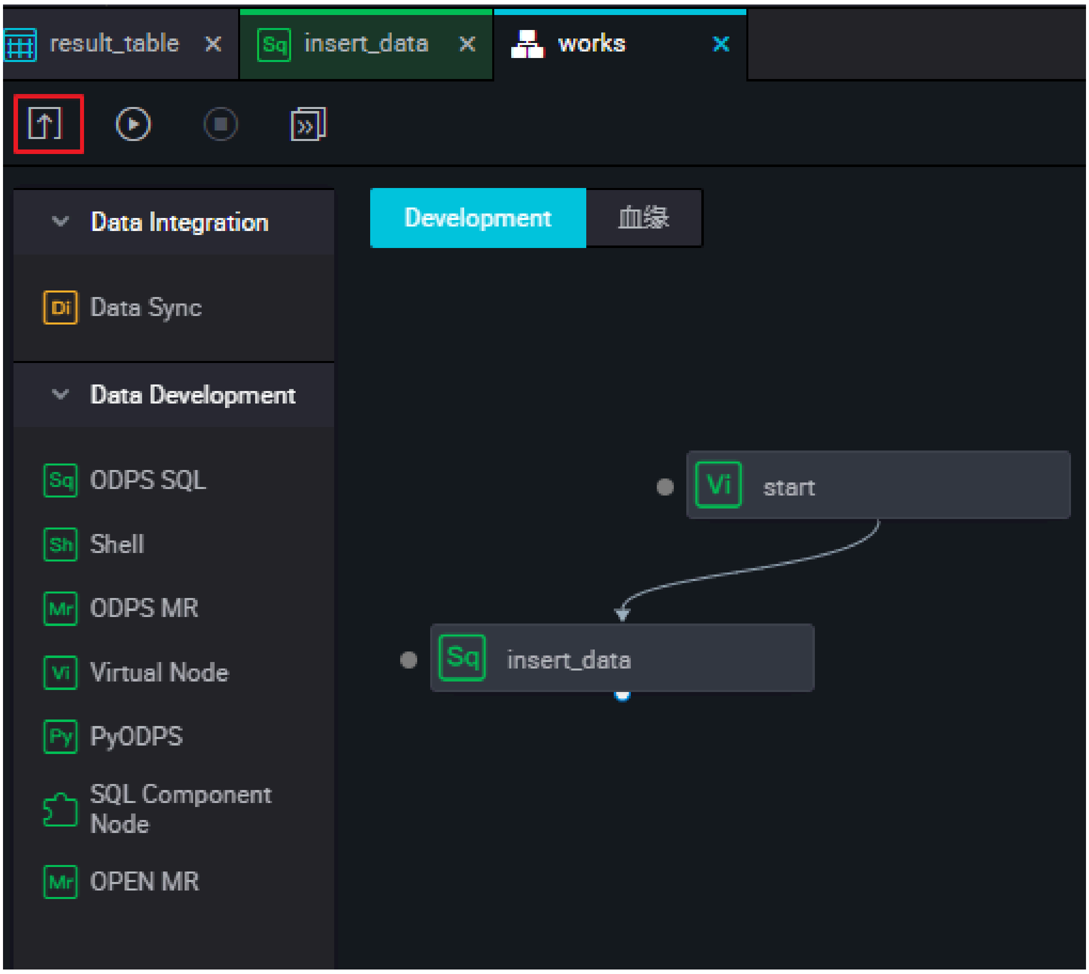


```
--odps sql
--*********************************************************************--
--author:[redacted]
--create time:2018-08-27 17:31:58
--*********************************************************************--
INSERT OVERWRITE TABLE result_table
SELECT education
       , COUNT(marital) AS num
FROM bank_data
WHERE housing = 'yes'
    AND marital = 'single'
GROUP BY education;
```

```
                min: 1.000, max: 1.000, avg: 1.000
        input records:
                StreamLineRead_REL9121: 8  (min: 8, max: 8, avg: 8)
        output records:
                TableSink_REL9124: 8  (min: 8, max: 8, avg: 8)
OK
2018-08-27 17:38:00 INFO =================================================================
2018-08-27 17:38:00 INFO Exit code of the Shell command 0
2018-08-27 17:38:00 INFO --- Invocation of Shell command completed ---
2018-08-27 17:38:00 INFO Shell run successfully!
2018-08-27 17:38:00 INFO Current task status: FINISH
2018-08-27 17:38:00 INFO Cost time is: 6.288s
/home/admin/alisatasknode/taskinfo//20180827/datastudio/17/37/52/ug9hj5iqus4zbjx6s4imo8qi/T3_0520116078.log-END-EOF
```

Save and submit business flows

After running and debugging the MaxCompute SQL node “insert_data”, return to the flow page. Click **Save** and **Submit** the whole flow.



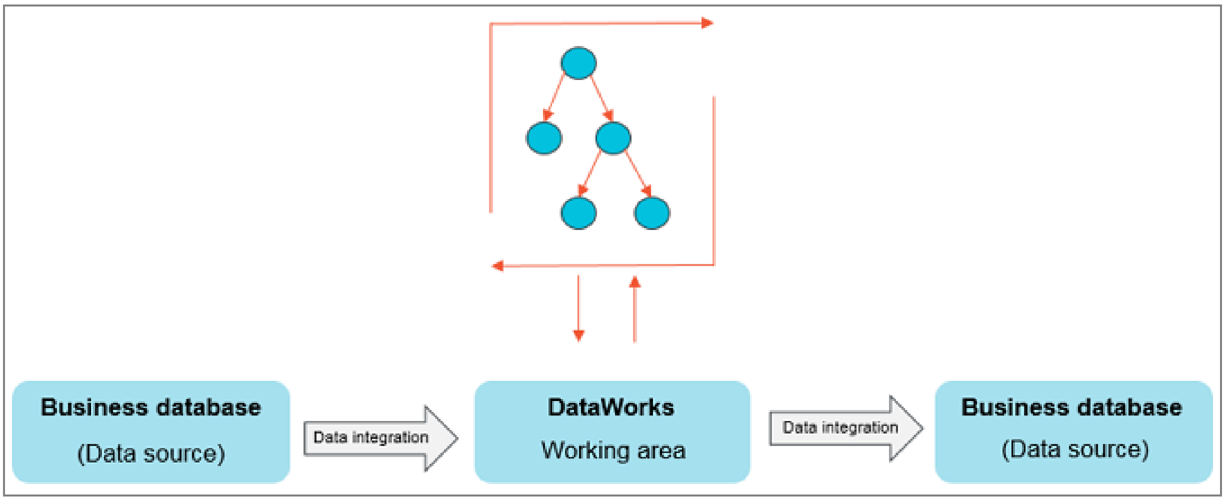
Subsequent steps

Now you have learned how to create, save, and submit the workflow. You can proceed to the next topic which shows how to create a synchronization task to export data to the different types of data sources. For more information, see *create synchronization task export results*.

4 Step 3: Create a synchronization task

This topic uses MySQL Data sources as an example, to show how to export data from MaxCompute to a MySQL data source through the data integration feature.

In DataWorks, data integration is typically used to periodically import business data generated in your system into the workspace after the SQL task calculation. The calculation results are periodically exported to the data source that you specify, for further details or running usage.



Currently, the following data sources can be imported or exported from the workspace through the data integration function: RDS, MySQL, SQL Server, PostgreSQL, MaxCompute, ApsaraDB for Memcache, DRDS, OSS, Oracle, FTP, DM, Hdfs, MongoDB, and so on. For more information, see *Supported data sources*.

Prerequisites

- If you are using a on-premises database on ECS, you need to *add security groups* to your ECS.
- If you are using data sources such as RDS or MongoDB, you need to *add a white list* to the data source console.

> **Note:**
> If you use a custom resource group to schedule the RDS data synchronization task, you must add the IP address of the computer hosting the custom resource group to the RDS whitelist.

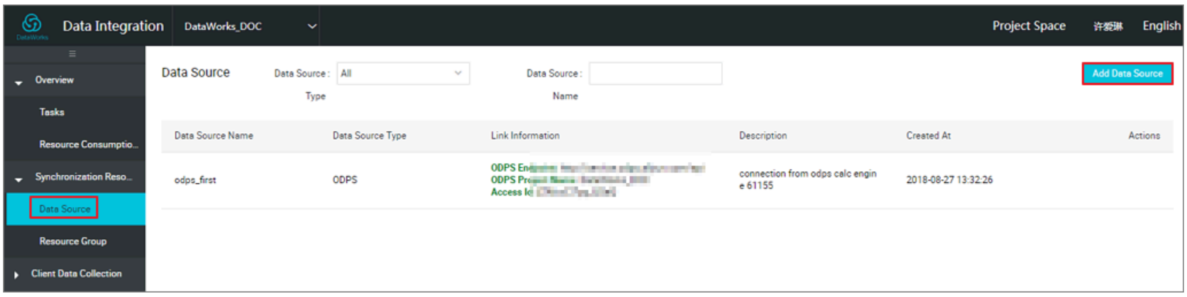
Procedure

Add data source

Note:

Only the Project Administrator role can create new data sources, and members of other roles can view data sources only.

1. Log on to the *DataWorks management console* as the Project Administrator.
2. Select **enter workspace** in the corresponding item actions column under the **list of items**.
3. Click **data integration** in the top menu bar.
4. Click **data sources** in the left-hand navigation bar.
5. Click **add data source** in the upper-right corner.

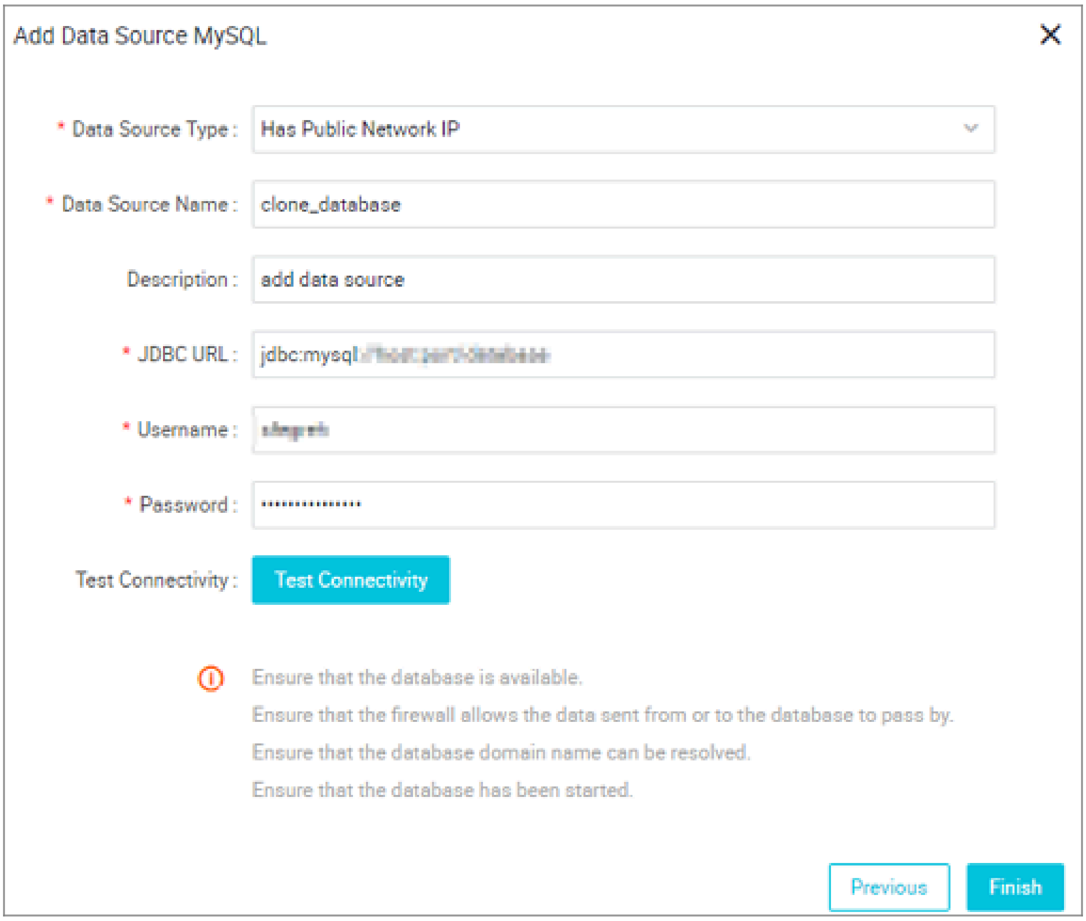


Data Source Type : All

Data Source Name :

Data Source Name	Data Source Type	Link Information	Description	Created At	Actions
odps_first	ODPS	ODPS En[redacted] ODPS Pr[redacted] Access I[redacted]	connection from odps calc engine 61155	2018-08-27 13:32:26	

6. Enter each configuration item in the **Add Data Source** dialog box.



* Data Source Type : Has Public Network IP

* Data Source Name : clone_database

Description : add data source

* JDBC URL : jdbc:mysql:[redacted]

* Username : [redacted]

* Password : ••••••••••••••••

Test Connectivity : Test Connectivity

Ensure that the database is available.

Ensure that the firewall allows the data sent from or to the database to pass by.

Ensure that the database domain name can be resolved.

Ensure that the database has been started.

- Data Source Type: With a public IP address.

- Data Source Name: The name must contain letters, numbers, and underlines, but cannot begin with a number or underline.For example: abc_1123.
- Data Source Description: The description cannot exceed 80 characters.
- JDBC URL: `jdbc:mysql://host:port/database`.
- User name/Password: The user name and password used to connect to the database.

For configuration instructions of different data source types, see *Data source configuration*.

7. (Optional) Click **Test Connectivity** after entering all the required information in the relevant fields.
8. If the test connectivity is successful, click **Finish**.

Note:

Make sure the target MySQL database contains tables.

Create the table odps_result in MySQL database. The statements used for table creation are as follows:

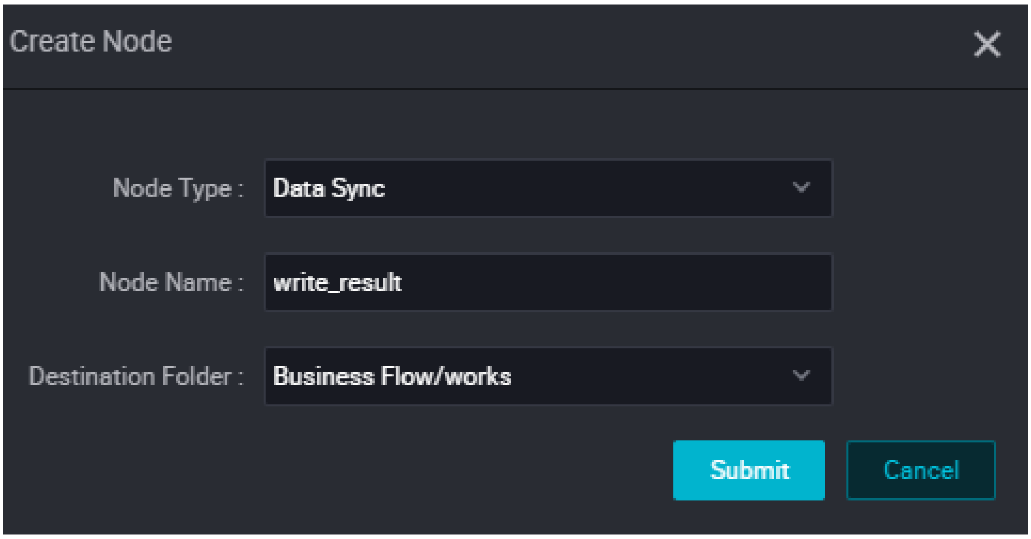
```
CREATE TABLE `ODPS_RESULT` (
`education`  varchar(255) NULL ,
`num`  int(10) NULL
)
```

After the table has been built, you can execute the `desc odps_result;` to view the table details.

Creating and configuring synchronization node

This section shows how to create and configure the synchronization node **write_result**, and write data from result_table to the MySQL database. The specific steps are as follows.

1. Create the node write_result, as shown in the following figure.



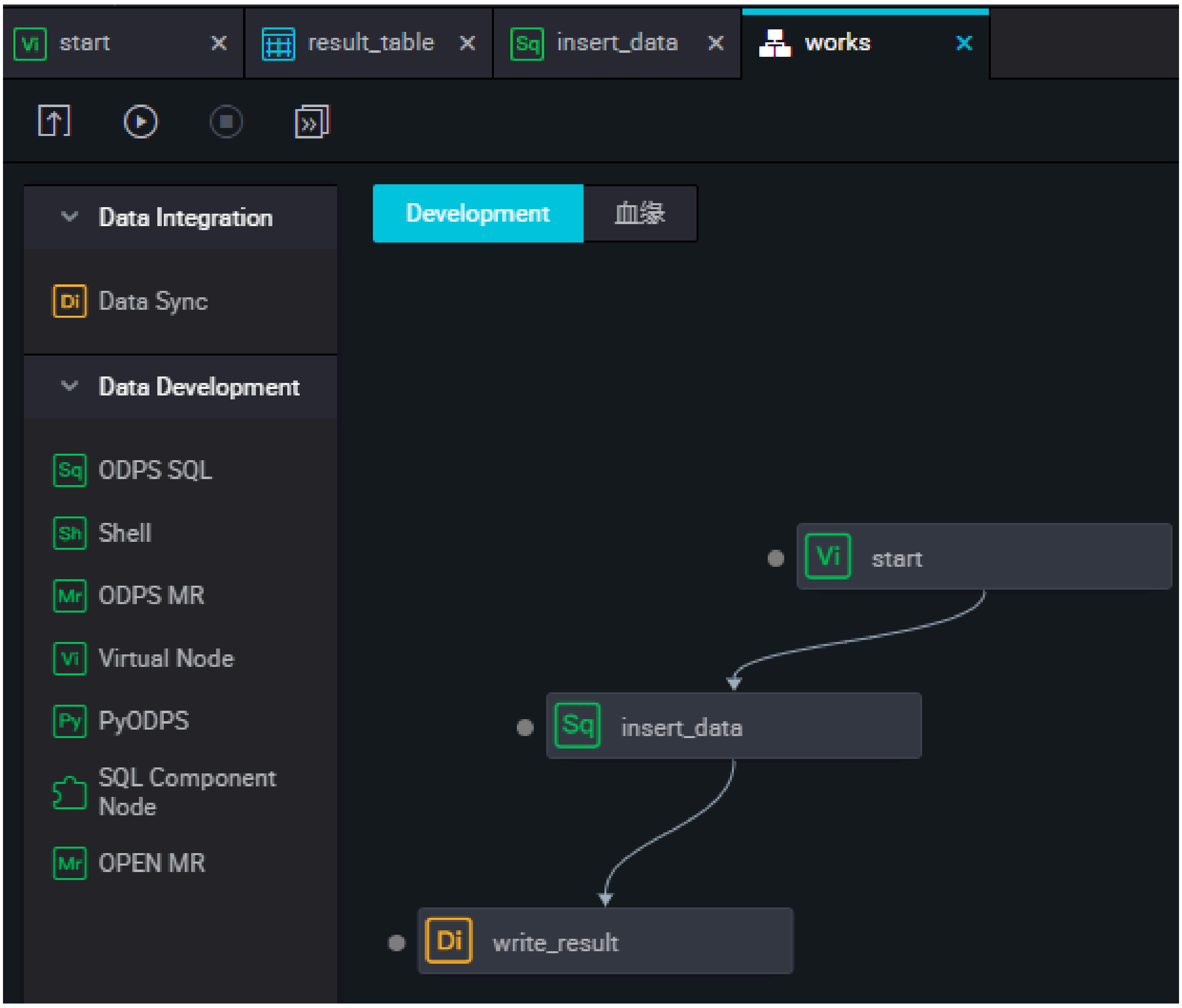
Create Node

Node Type : Data Sync

Node Name : write_result

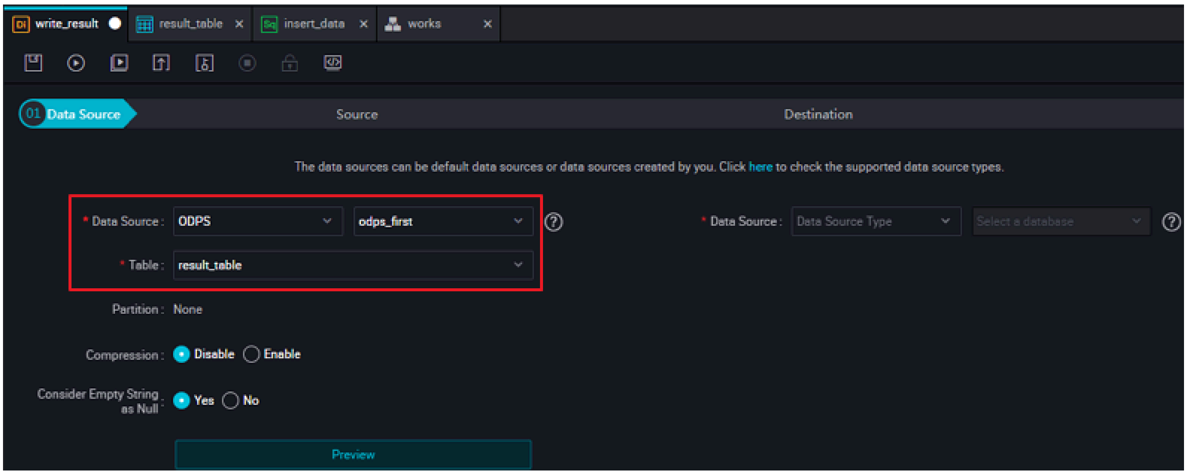
Destination Folder : Business Flow/works

2. Sets the dependencies between nodes so the write_result node is dependent on the insert_data node.



3. Select the source.

Select the MaxCompute data source and the source table result_table and click **Next**.



The data sources can be default data sources or data sources created by you. Click here to check the supported data source types.

* Data Source : ODPS odps_first

* Table : result_table

Partition : None

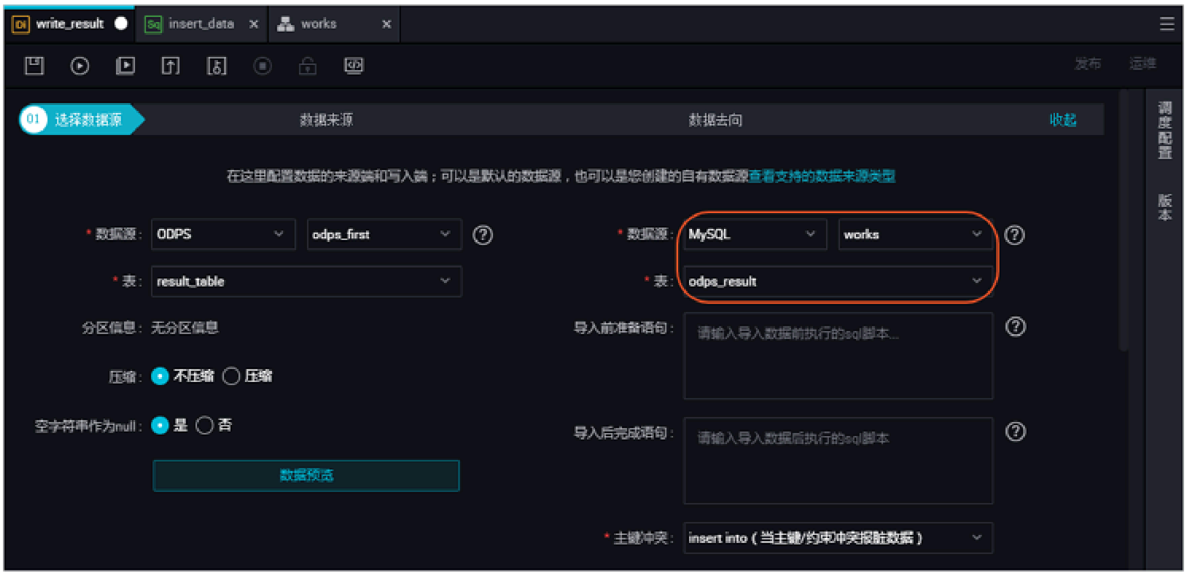
Compression : ● Disable ○ Enable

Consider Empty String as Null : ● Yes ○ No

* Data Source : Data Source Type Select a database

4. Select a Target.

   Select the MySQL data source and target table ODPS _result, and click **Next**.



在这里配置数据的来源端和写入端；可以是默认的数据源，也可以是您创建的自有数据源查看支持的数据来源类型

* 数据源 : ODPS odps_first

* 表 : result_table

分区信息 : 无分区信息

压缩 : ● 不压缩 ○ 压缩

空字符串作为null : ● 是 ○ 否

* 数据源 : MySQL works

* 表 : odps_result

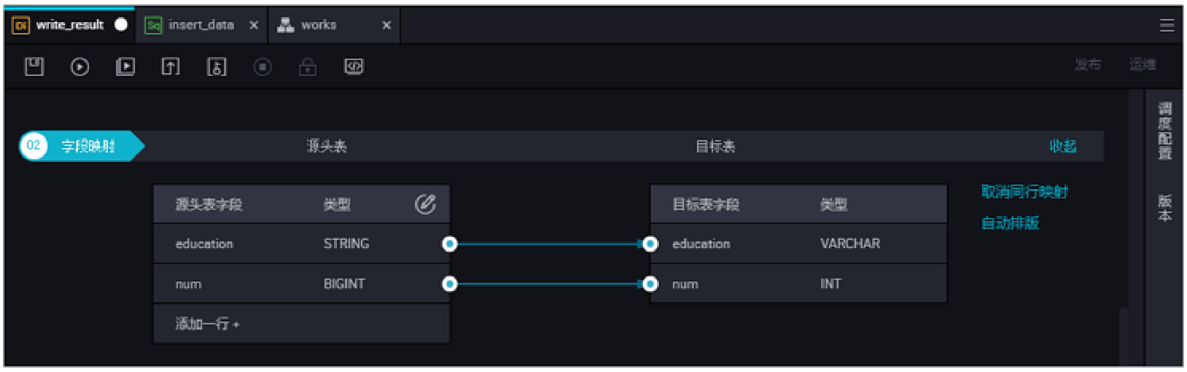
导入前准备语句 : 请输入导入数据前执行的sql脚本...

导入后完成语句 : 请输入导入数据后执行的sql脚本

* 主键冲突 : insert into（当主键/约束冲突报脏数据）

5. Map the fields.

   Select mapping between fields. You need to configure the field mapping relationships. The "Source Table Fields" on the left correspond one to one with the "Target Table Fields" on the right.

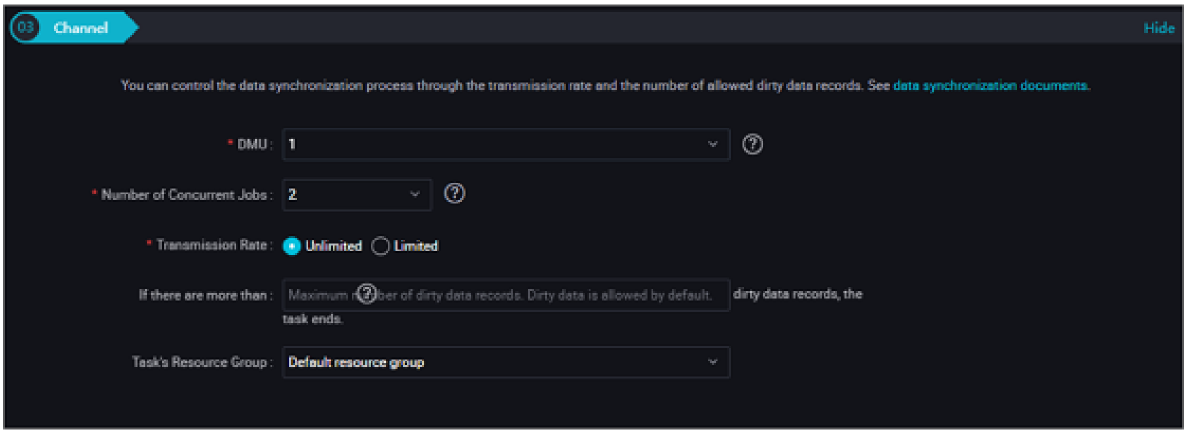


源头表字段	类型
education	STRING
num	BIGINT

目标表字段	类型
education	VARCHAR
num	INT

6. Control the channel.

   Click **Next** to configure the maximum job rate and dirty data check rules.



03 Channel

Hide

You can control the data synchronization process through the transmission rate and the number of allowed dirty data records. See data synchronization documents.

* DMU : 1

* Number of Concurrent Jobs : 2

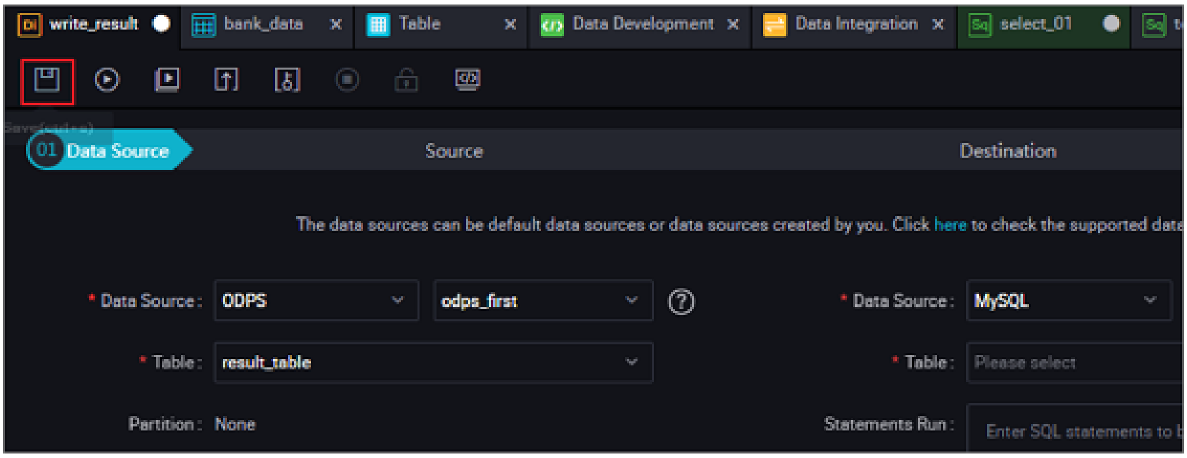
* Transmission Rate : ● Unlimited ○ Limited

If there are more than : Maximum number of dirty data records. Dirty data is allowed by default. dirty data records, the task ends.

Task's Resource Group : Default resource group

7. Preview and store.

   After completing the above configuration, scroll the mouse up and down to view the task configuration, and if it is not configured, click **Save**.



01 Data Source

Source

Destination

The data sources can be default data sources or data sources created by you. Click here to check the supported data

* Data Source : ODPS odps_first

* Data Source : MySQL

* Table : result_table

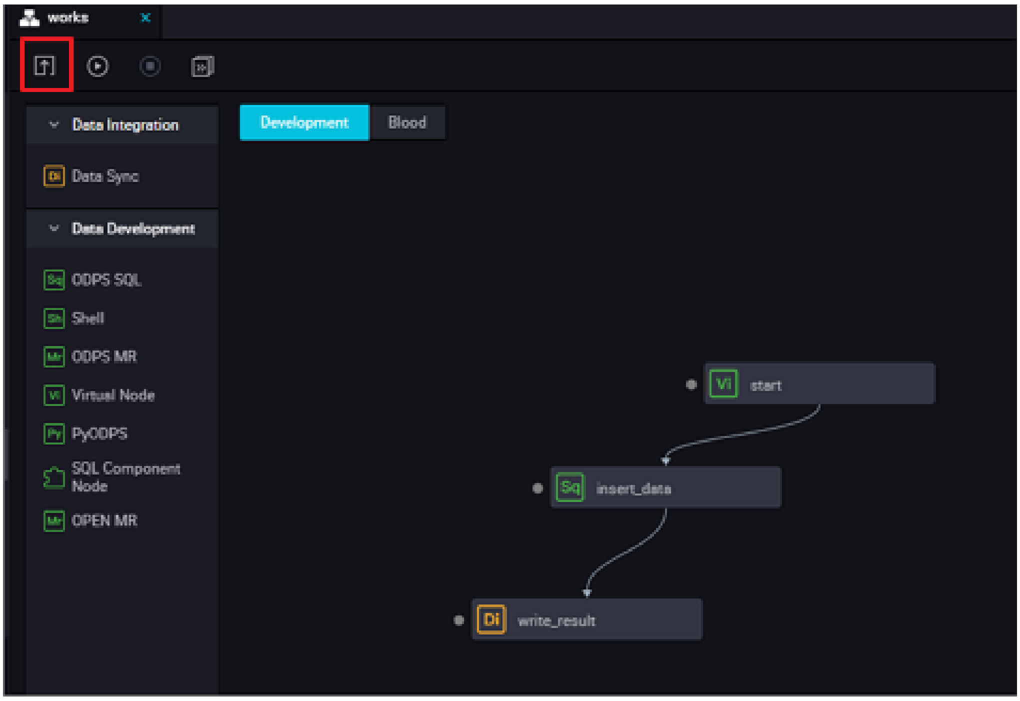
* Table : Please select

Partition : None

Statements Run : Enter SQL statements to b

Submit a data synchronization task

Once the syncrhonization task is saved, click **Submit** to submit the task to the scheduling system. The scheduling system automatically and periodically runs the task from the second day according to the configuration attributes.



Subsequent steps

Now, you know how to create a synchronization task and export data to different data sources. Continue to the next topic to learn how to set scheduling attributes and dependencies for a synchronization task. For more information, see *setting schedule properties and dependencies* for tasks.

5 Step 4: Scheduling and dependency settings

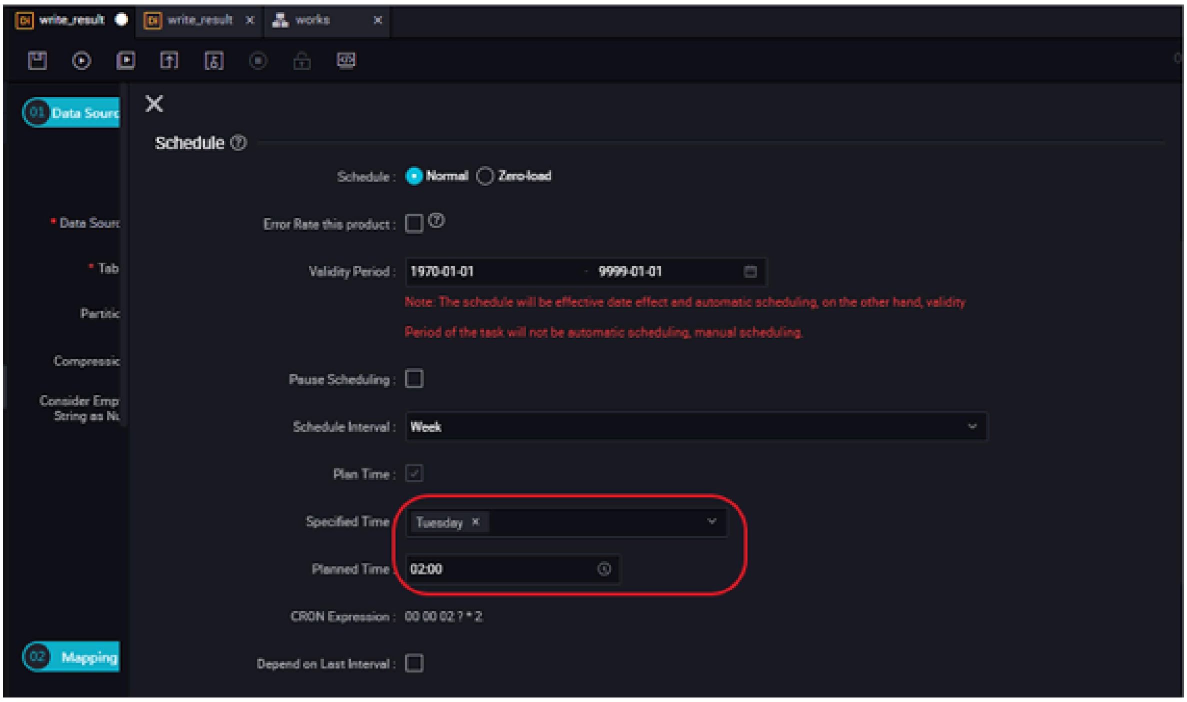
This article takes the "write_result" created in *creating synchronization tasks* as an example, configure its scheduling cycle as weekly scheduling, introduces the scheduling configuration and task operations features of DataWorks.

DataWorks provides powerful scheduling capabilities including time-based or dependency-based task trigger functions to perform **tens of millions** of tasks accurately and timely each day, based on DAG relationships. It supports scheduling by minute, hour, day, week, and month. For more information, see *Create a synchronization task*.

Procedure

Configure the scheduling attribute of a synchronization task

1. Select **data development** > **task Development** page.
2. Double-click the synchronization task (write_result) that you want to configure).
3. Click **schedule configuration** on the right to configure scheduling properties for the task.



Schedule

Schedule : ● Normal ○ Zero-load

Error Rate this product : □

Validity Period : 1970-01-01 - 9999-01-01

Note: The schedule will be effective date effect and automatic scheduling, on the other hand, validity Period of the task will not be automatic scheduling, manual scheduling.

Pause Scheduling : □

Schedule Interval : Week

Plan Time : ☑

Specified Time : Tuesday ×

Planned Time : 02:00

CRON Expression : 00 00 02 ? * 2

Depend on Last Interval : □

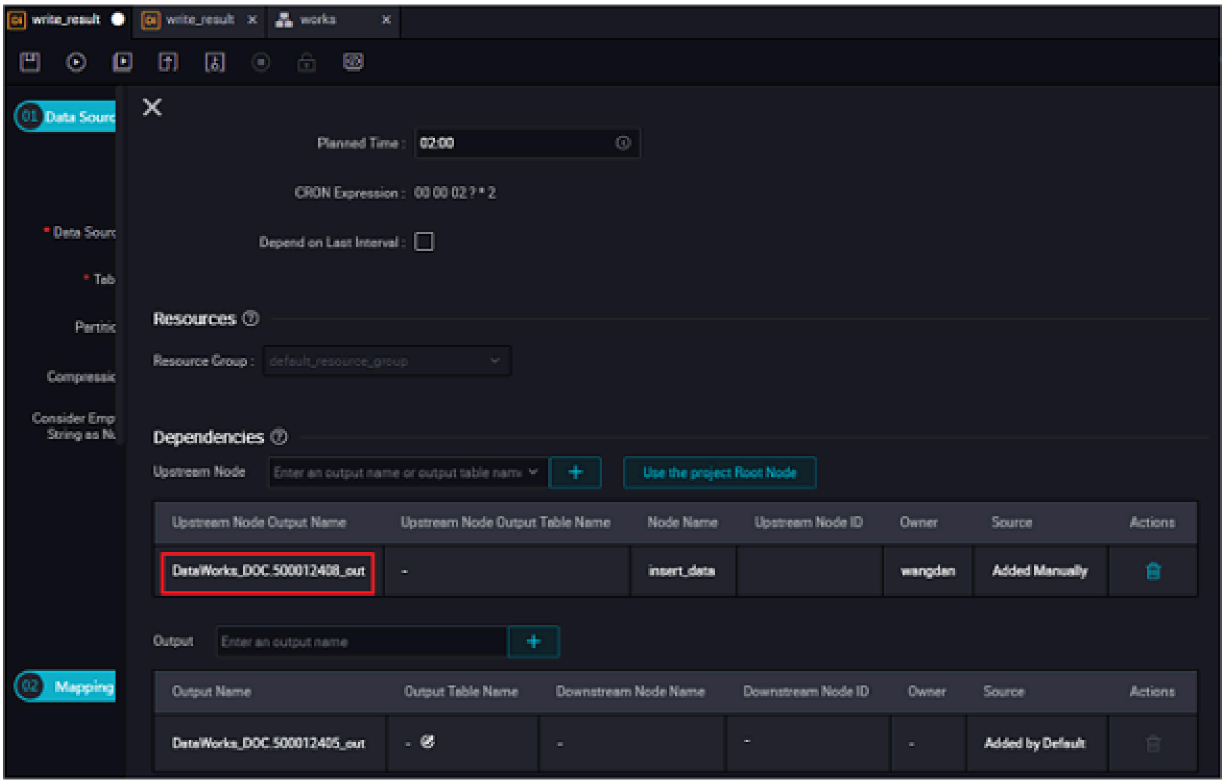
Parameters:

- Scheduling status: The task is paused when this parameter is selected.
- Error retry: Error retry is enabled when this parameter is selected.
- Start date: The date that the task takes effect can be set based on requirements.

- Scheduling period: The operating cycle of the task can be set by month, week, day, hour, and minute. For example, a task can be scheduled weekly.
- Specific time: The specific task operating time. For example, you can set up the task to run at 02:00 every Tuesday.

Configure dependency properties for a synchronization task

After completing the synchronization task schedule properties configuration , you can configure its deployment dependency properties.



Planned Time : 02:00

CRON Expression : 00 00 02 ? * 2

Depend on Last Interval : □

Resources

Resource Group : default_resource_group

Dependencies

Upstream Node: Enter an output name or output table name

Upstream Node Output Name	Upstream Node Output Table Name	Node Name	Upstream Node ID	Owner	Source	Actions
DataWorks_DOC.500012408_out	-	insert_data		wangdan	Added Manually	🗑

Output: Enter an output name

Output Name	Output Table Name	Downstream Node Name	Downstream Node ID	Owner	Source	Actions
DataWorks_DOC.500012405_out	-	-	-	-	Added by Default	🗑

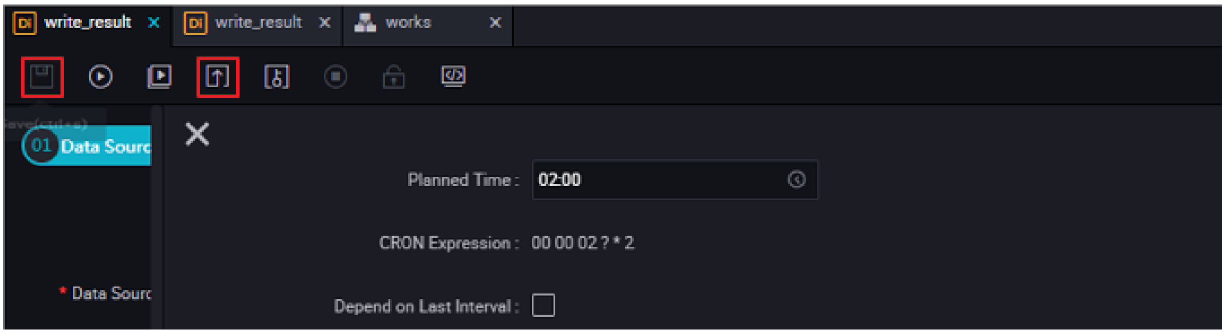
You can configure an upstream dependency for a task. In this way, even if the current task instance reaches the scheduled time, the task only run after the instance upstream task is completed.

The configuration in the preceding figure indicates the instances of the current task are triggered only after the upstream task write_result is finished. You can enter **work** in the upstream task to configure an upstream task for write_result.

If no upstream tasks is configured then, by default the current task is triggered by the project. Therefore, by default, the upstream task of the current task is project_start in the scheduling system. By default, a project_start task is created as a root task for each project.

Submit a data synchronization task

Save the synchronization task **write_result** and click **Submit** to submit it to the scheduling system.



Planned Time : 02:00

CRON Expression : 00 00 02 ? * 2

Depend on Last Interval : □

The system automatically generates an instance for the task at each time point according to the scheduling attribute configuration and periodically runs the task from the second day only after a task is submitted to the scheduling system.

Note:

If the task is submitted after 23: 30, the scheduling system automatically cycle-generate instances from the third day and run on time.

Subsequent steps

Now you know how to set a synchronization task scheduling attribute and dependency, now you can continue to the next topic to learn how to perform periodic O&M for submitted tasks and view the log troubleshooting results. For more information, see *cycle care operations and check for log ranking errors*.

6 Step 5: O&M and view log troubleshooting results

This topic describes how to implement task operations.

In the previous operations, you set a synchronization task to run at 02:00 every Tuesday. After the task is submitted, you can view the automatic operation results in the scheduling system next day .

To check whether the instance schedule and dependency are operating as expected, DataWorks provides three triggering methods: test run, data population, and periodic running, which are described as follows:

- Test run: The task is triggered manually. If you need to check the timing and operation of a single task, test run is recommended.
- Data population: The task is triggered manually. This method applies if you need to check the timing and dependencies of multiple tasks or re-execute data analysis and computing from a root task.
- Periodic running: The task is triggered automatically. After successful submission, the scheduling system automatically generates task instances at different time points starting from 00:00 the next day. It checks whether upstream instances of each instance can run successfully according to the scheduled time. If all the upstream instances run successfully at the scheduled time, the current instance runs automatically.

Note:

The scheduling system periodically generates instances based on the same rules that apply to both manual and automatic triggering modes.

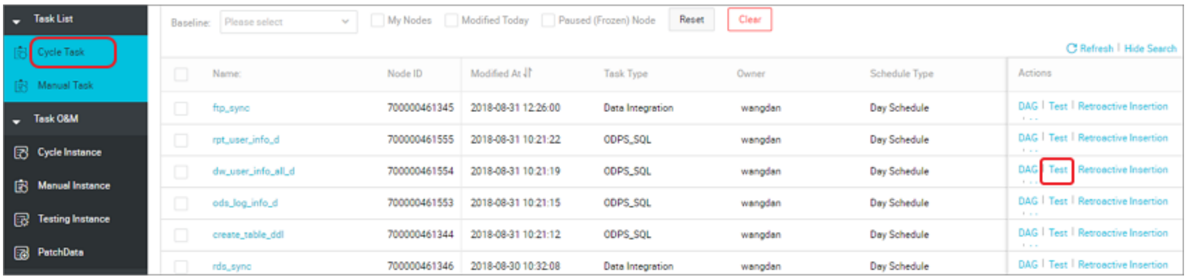
- The period can be set to monthly, weekly, daily, hourly, or even by minutes. The scheduling system always generates an instance for the task on a specified day or at a specified time.
- The scheduling system regularly runs the instance on a specified date and generates operation logs.
- Instances rather than a specified date does not run, and their statuses are directly changed to “Successful” if the running conditions are met. Therefore, no running logs are generated.

For more operational and functional instructions, see *Task operations*.

Test

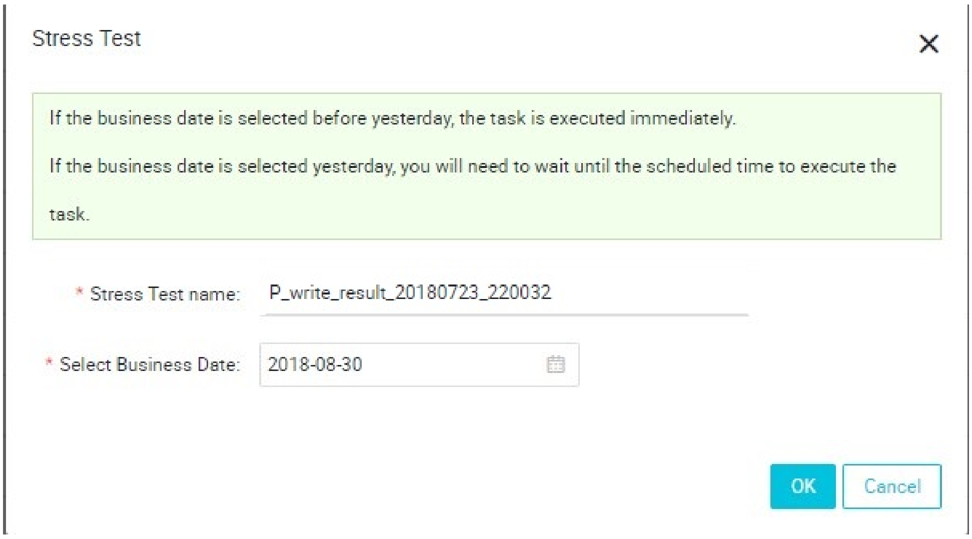
Manually trigger a test

1. On the **Cycle Task** page, locate the task that you want to run, and click **Test**.



□	Name:	Node ID	Modified At	Task Type	Owner	Schedule Type	Actions
□	ftp_sync	700000461345	2018-08-31 12:26:00	Data Integration	wangdan	Day Schedule	DAG | Test | Retroactive Insertion
□	rpt_user_info_d	700000461555	2018-08-31 10:21:22	ODPS_SQL	wangdan	Day Schedule	DAG | Test | Retroactive Insertion
□	dw_user_info_all_d	700000461554	2018-08-31 10:21:19	ODPS_SQL	wangdan	Day Schedule	DAG | Test | Retroactive Insertion
□	ods_log_info_d	700000461553	2018-08-31 10:21:15	ODPS_SQL	wangdan	Day Schedule	DAG | Test | Retroactive Insertion
□	create_table_ddl	700000461344	2018-08-31 10:21:12	ODPS_SQL	wangdan	Day Schedule	DAG | Test | Retroactive Insertion
□	rds_sync	700000461346	2018-08-30 10:32:08	Data Integration	wangdan	Day Schedule	DAG | Test | Retroactive Insertion

2. Enter the business Date and click **OK**.



Stress Test

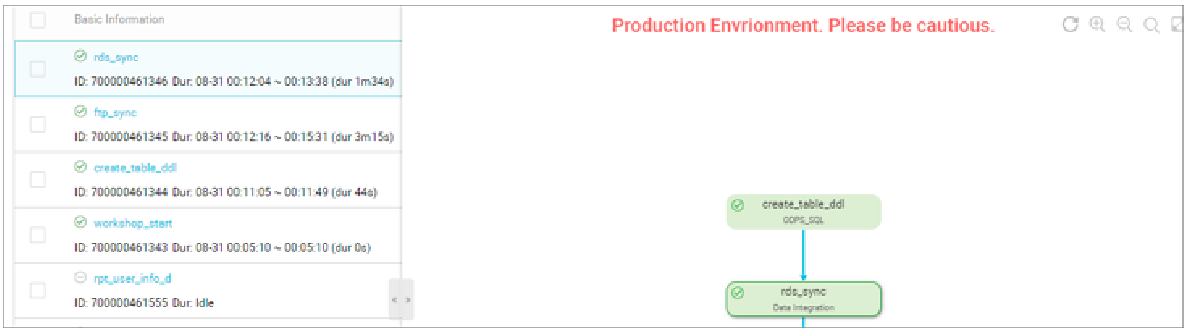
If the business date is selected before yesterday, the task is executed immediately.

If the business date is selected yesterday, you will need to wait until the scheduled time to execute the task.

* Stress Test name: P_write_result_20180723_220032

* Select Business Date: 2018-08-30

3. Go to the **Basic information** page to view the task run status.



rds_sync
ID: 700000461346 Dur: 08-31 00:12:04 ~ 00:13:38 (dur 1m34s)

ftp_sync
ID: 700000461345 Dur: 08-31 00:12:16 ~ 00:15:31 (dur 3m15s)

create_table_ddl
ID: 700000461344 Dur: 08-31 00:11:05 ~ 00:11:49 (dur 44s)

workshop_start
ID: 700000461343 Dur: 08-31 00:05:10 ~ 00:05:10 (dur 0s)

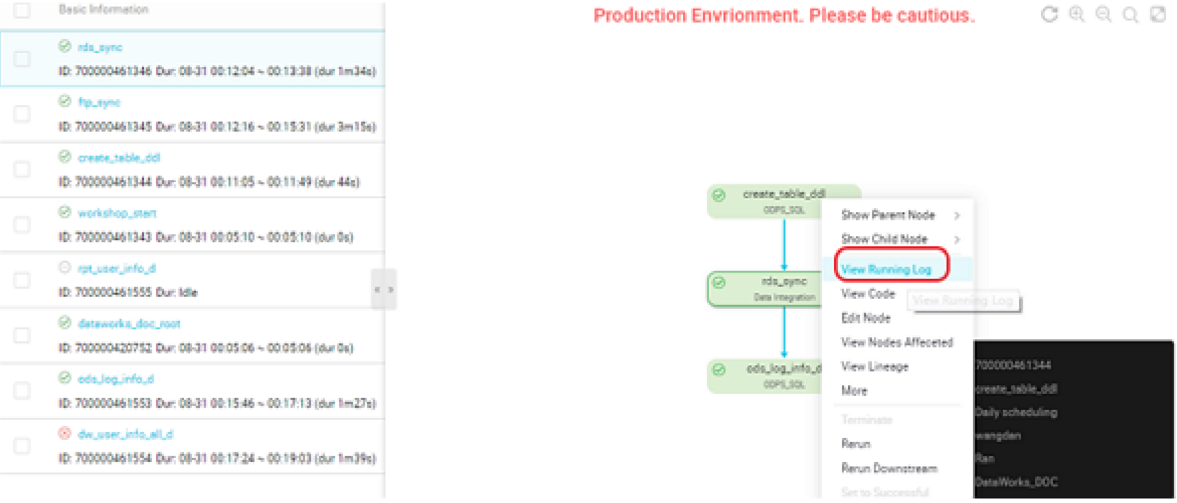
rpt_user_info_d
ID: 700000461555 Dur: Idle

View the information and operation logs of the test instance

You can see the instance DAG graph by selecting the appropriate task instance in the **test instance** page and clicking.

- Right-click an instance, you can view the instance's dependencies and details and perform specific actions such as stop, resume, and more.

- Double-click an instance to enter the pop up task properties, run log, operation log, code, and so on.



Basic Information

- rds_sync
  ID: 700000461346 Dur: 08-31 00:12:04 ~ 00:13:38 (dur 1m34s)
- ftp_sync
  ID: 700000461345 Dur: 08-31 00:12:16 ~ 00:15:31 (dur 3m15s)
- create_table_ddl
  ID: 700000461344 Dur: 08-31 00:11:05 ~ 00:11:49 (dur 44s)
- workshop_start
  ID: 700000461343 Dur: 08-31 00:05:10 ~ 00:05:10 (dur 0s)
- rpt_user_info_d
  ID: 700000461555 Dur: Idle
- dataworks_doc_root
  ID: 700000420752 Dur: 08-31 00:05:06 ~ 00:05:06 (dur 0s)
- ods_log_info_d
  ID: 700000461553 Dur: 08-31 00:15:46 ~ 00:17:13 (dur 1m27s)
- dw_user_info_all_d
  ID: 700000461554 Dur: 08-31 00:17:24 ~ 00:19:03 (dur 1m39s)

Production Envrionment. Please be cautious.

700000461344
create_table_ddl
Daily scheduling
wangdan
Ran
DataWorks_DOC

Note:

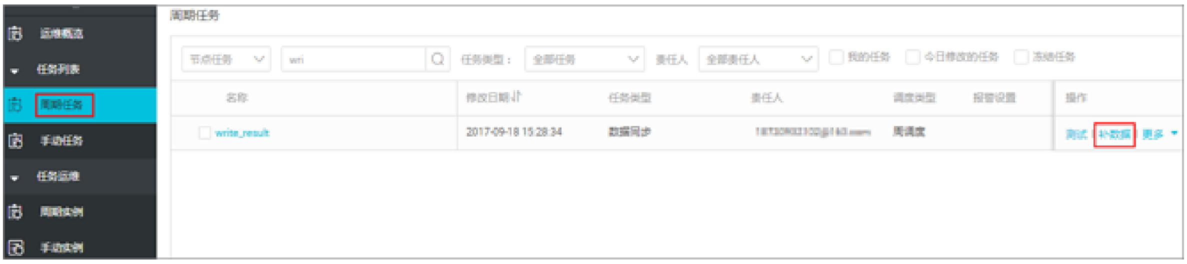
- In test run mode, the task is triggered manually. The task runs immediately as long as the set time is reached, regardless of the instance's upstream dependencies.
- The task write_result is configured to run every Tuesday morning, based on the instance generation rules described earlier in the topic. The business date selected by the test Runtime is Monday (business date = run date-1), the instance will actually run at 2. If it is not Monday, the instance is converted to a successful state at 2 points, and there is no log generation.

Replenishment data operation

Manually trigger data population

If you need to confirm the timing and interdependencies of multiple tasks, or need to re-perform the data analysis calculation from a root task, you can select the **O&M center** > **task list** > **cycle task** page and click the **replenishment data** after the task, to enter multiple tasks scheduled at a certain period of time.

1. Select the **O&M center** > **cycle task** page and enter the task name.
2. Click **replenishment data**after the query results.



周期任务

节点任务 wri 任务类型： 全部任务 责任人 全部责任人 □ 我的任务 □ 今日修改的任务 □ 冻结任务

名称	修改日期	任务类型	责任人	调度类型	报警设置	操作
□ write_result	2017-09-18 15:28:34	数据同步	[illegible]	周调度		测试 | 补数据 | 更多

3. Set the business date for the replenishment data as "to", select the write_result node task, and click **OK**.
4. Click to **view the replenishment data results**.

View the information and operation logs of the data population instance

You can see the instance DAG graph by selecting the appropriate task instance.

- Right-click an instance, you can view the dependencies and details of this instance and perform specific actions such as stop, resume, and so on..
- Double-click an instance to pop up task properties, run log, operation log, code, and so on.

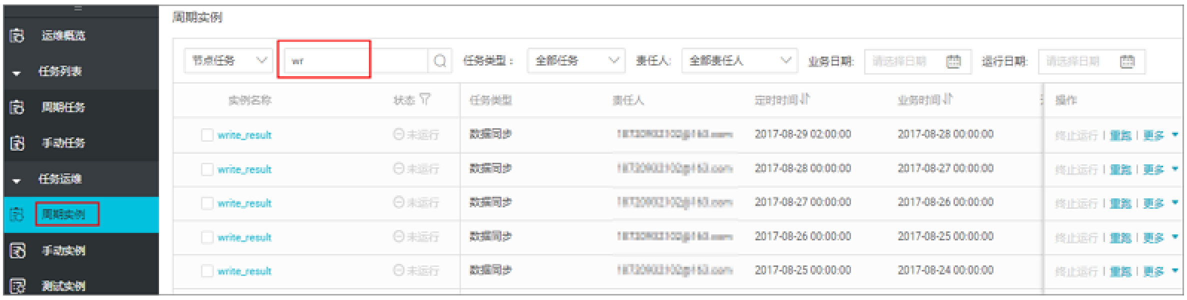
Note:

- 2017-09-18 15:56:30. 919 [job-51109647] is the job ID in the preceding figure.
- In the preceding figure, the task failed because the source does not have the partition value in the synchronized table, resulting in a read error.
- The instance of a replenishment data task is day-to-day. For example, the task runs from 2017-09-15 to 2017-09-18 , if the instance number 15 fails during this period, an instance of number 16 also will not run.
- The task write_result is configured to run every Tuesday morning, and based on the instance generation rules described earlier in the article. The business date selected by the replenishment data Runtime is Monday (business date = run date-1). The instance will run at 2 AM. If it is not Monday, the instance is converted to a successful state at 2 AM, and no log is generated.

Periodic automatic run

Under periodic automatic run mode, the scheduling system automatically triggers tasks according to all task scheduling configurations. Therefore, no operation portal is provided. You can view the instance information and operation logs by using either of the following methods.

- Select the parameters such as the business date or the running date on the **O&M center** > **cycle instance** page, search for the instance that corresponds to the write_result task, and then right-click on the instance information and the run log.

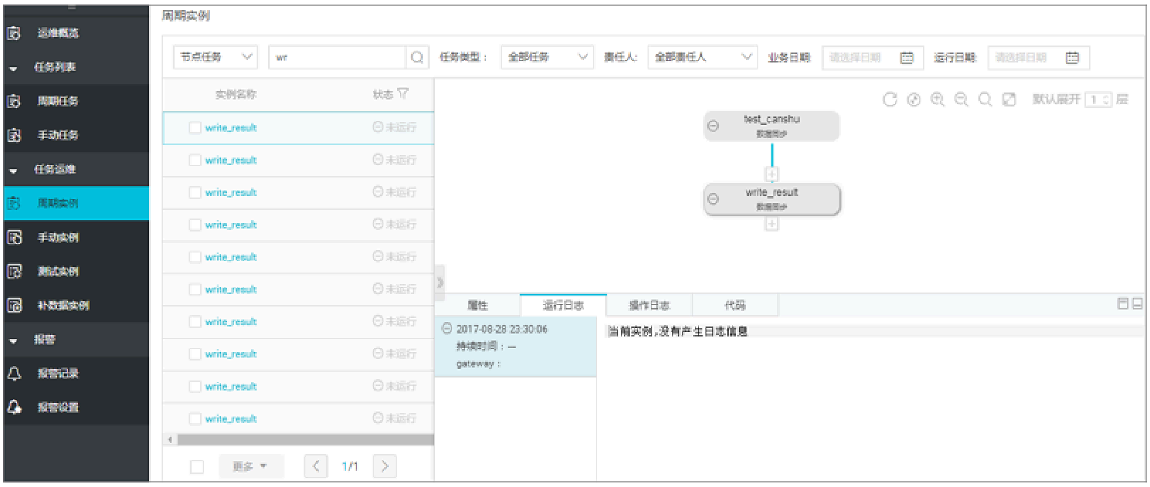


周期实例

节点任务 | wr | 任务类型： 全部任务 | 责任人: 全部责任人 | 业务日期: 请选择日期 | 运行日期: 请选择日期

实例名称	状态	任务类型	责任人	定时时间	业务时间	操作
□ write_result	⊖未运行	数据同步	[illegible]	2017-08-29 02:00:00	2017-08-28 00:00:00	终止运行 | 重跑 | 更多
□ write_result	⊖未运行	数据同步	[illegible]	2017-08-28 00:00:00	2017-08-27 00:00:00	终止运行 | 重跑 | 更多
□ write_result	⊖未运行	数据同步	[illegible]	2017-08-27 00:00:00	2017-08-26 00:00:00	终止运行 | 重跑 | 更多
□ write_result	⊖未运行	数据同步	[illegible]	2017-08-26 00:00:00	2017-08-25 00:00:00	终止运行 | 重跑 | 更多
□ write_result	⊖未运行	数据同步	[illegible]	2017-08-25 00:00:00	2017-08-24 00:00:00	终止运行 | 重跑 | 更多

- You can see the instance DAG graph by selecting the appropriate task instance in the **cycle instance** page and clicking.
  - Right-click an instance, you can view the dependencies and details of this instance and perform specific actions such as stop, resume, and so on..
  - Double-click an instance to pop up task properties, run log, operation log, code, and so on.



周期实例

节点任务 | wr | 任务类型： 全部任务 | 责任人: 全部责任人 | 业务日期 请选择日期 | 运行日期 请选择日期

实例名称	状态
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行
□ write_result	⊖未运行

⊖ 2017-08-28 23:30:06
持续时间：—
gateway：

当前实例,没有产生日志信息

Note:

- The task is not running because the upstream task is not running.
- If the initial state of an instance of a task is “Not Run”, when the scheduled time arrives, the scheduling system checks all upstream instances of this instance are running successfully.
- The instance will be triggered only when all of its upstream instances are successful and its scheduled time is reached.
- For a Not Run status instance , check all its upstream instances are successful and has reached its scheduled time.