

ALIBABA CLOUD

Alibaba Cloud

Function Compute
SDK Reference

Document Version: 20200929

 Alibaba Cloud

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 because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should 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 this document based on the "status quo", "being defective", and "existing functions" of its products and services. 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 take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, 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 contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document 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 directly contact Alibaba Cloud for any errors of this document.

Document conventions









Style	Description	Example
 Danger	A danger notice 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.
 Warning	A warning notice 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 restart an instance.
 Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: If the weight is set to 0, the server no longer receives new requests.
 Note	A note indicates supplemental instructions, best practices, tips, and other content.	 Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click Settings> Network> Set network type .
Bold	Bold formatting is used for buttons, menus, page names, and other UI elements.	Click OK .
Courier font	Courier font is used for commands	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
<i>Italic</i>	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	This format is used for a required value, where only one item can be selected.	<code>switch {active stand}</code>

Table of Contents

1. Supported SDKs	05
2. Go	06
2.1. Set the instance type for a function	06
2.2. Configure pay-as-you-go instances	07
2.3. Create custom container runtime functions	09
3. Node.js	11
3.1. Specify the instance concurrency	11

1. Supported SDKs

The following table lists the supported SDKs. SDKs in more languages are coming soon.

Language	SDK
Node.js	fc-nodejs-sdk
Java	fc-java-sdk
Python	fc-python-sdk
Go	fc-go-sdk
PHP	fc-php-sdk
C#	fc-csharp-sdk

2.Go

2.1. Set the instance type for a function

This topic describes how to use the sample code of Alibaba Cloud SDK for Go to set the instance type for a function.

When you create or modify a function, you can use Alibaba Cloud SDK for Go to specify whether to use the flexible instance or performance instance type for your function. The backend routes your request to the specified instance type.

You can set the instance type parameter to one of the following values:

- e1: flexible instance
- c1: performance instance

If you do not specify an instance type, your function uses the flexible instance by default.

Sample code

```
// Go sdk
package main

import (
    "github.com/aliyun/fc-go-sdk"
)

func main() {
    // new fc client
    client, _ := fc.NewClient("endpoint", "2016-08-15",
        "accessKeyID", "accessKeySecret")

    // CreateService
    serviceName := "TestService"
    _, err := client.CreateService(fc.NewCreateServiceInput().
        WithServiceName(serviceName))
    if err != nil {
        panic(err)
    }

    // CreateFunction
    functionName := "TestFunction"
    createFunctionInput := fc.NewCreateFunctionInput(serviceName).
        WithFunctionName(functionName).
        WithHandler("index.handler").
        WithRuntime("python2.7").
        WithCode(fc.NewCode().WithFiles("./code/index.py")).
        WithTimeout(5).
        WithInstanceType("c1") // Specifies that the function uses the performance instance type.
    _, err = client.CreateFunction(createFunctionInput)
    if err != nil {
        panic(err)
    }

    return
}
```

2.2. Configure pay-as-you-go instances

This topic describes how to use the SDK to create a pay-as-you-go instance for a function.

SDK code samples

This topic describes how to specify a pay-as-go instance for a function in Go language by using the following methods:

```
package main

import (
    "fmt"
    fc "https://github.com/aliyun/fc-go-sdk"
)

const (
    endpoint = "your-endpoint"
    fcAPIVersion = "2016-08-15"
    AccessKeyID = "your-ak-id"
    AccessKeySecret = "your-ak-secret"
)

var (
    serviceName = "service-name"
    qualifier = "LATEST"
    functionName = "function-name"
)

func main() {
    fcClient, _ := fc.NewClient(endpoint, fcAPIVersion, AccessKeyID, AccessKeySecret)

    // set
    putInput := fc.NewPutOnDemandConfigInput(serviceName, qualifier, functionName)
    putInput.WithMaximumInstanceCount(10)
    putResp, err := fcClient.PutOnDemandConfig(putInput)
    fmt.Println("put on-demand config", putResp, err)

    // list
    listInput := fc.NewListOnDemandConfigsInput()
    listInput.WithPrefix(fmt.Sprintf("services/%s", serviceName))
    listResp, err := fcClient.ListOnDemandConfigs(listInput)
    fmt.Println("list on-demand configs", listResp, err)

    // get
    getInput := fc.NewGetOnDemandConfigInput(serviceName, qualifier, functionName)
```



```
getResp, err := fcClient.GetOnDemandConfig(getInput)
fmt.Println("get on-demand configs", getResp, err)

// delete
deleteInput := NewDeleteOnDemandConfigInput(serviceName, qualifier, functionName)
deleteResp, err := fcClient.DeleteOnDemandConfig(deleteInput)
fmt.Println("delete on-demand configs", deleteResp, err)
}
```

2.3. Create custom container runtime functions

This topic describes how to use the SDK to create a custom container runtime function.

SDK code samples

The following code shows how to use SDK for Go to create a custom container runtime function:

```
import (
    "fmt"

    fc "github.com/aliyun/fc-go-sdk"
)

const (
    AK_ID    = "your ak id"
    AK_SECRET = "your ak secret"
    ROLE     = "your role arn" // role with AliyunContainerRegistryReadOnlyAccess/AliyunContainerRegistryFullAccess, for example: acs:ram::123456:role/service-role
    IMAGE    = "your ACR image" // for example: registry-vpc.cn-shanghai.aliyuncs.com/fc-test/hello-world:v1
)

func main() {
    serviceName := "fc-demo-ccr"
    funcName := "test"
    client, _ := fc.NewClient("http://123456.cn-shanghai.fc.aliyuncs.com", "2016-08-15", AK_ID, AK_SECRET)

    _, err := client.CreateService(fc.NewCreateServiceInput().
        WithServiceName(serviceName).
        WithName(funcName).
        WithImage(IMAGE).
        WithRole(ROLE))
}
```

```
withRole(RULE).
    WithDescription("this is a test service for docker image"))
if err != nil {
    panic(err)
}

// CreateFunction
createFunctionInput := fc.NewCreateFunctionInput(serviceName).WithFunctionName(funcName).
    WithDescription("test function").
    WithHandler("just-a-string").WithRuntime("custom-container").
    WithCustomContainerConfig(fc.NewCustomContainerConfig().WithImage(IMAGE)).
    WithTimeout(30)

_, err = client.CreateFunction(createFunctionInput)
if err != nil {
    panic(err)
}

invokeInput := fc.NewInvokeFunctionInput(serviceName, funcName).WithPayload([]byte("hello world
"))
invokeOutput, err := client.InvokeFunction(invokeInput)
if err != nil {
    panic(err)
} else {
    fmt.Printf("InvokeFunction response: %s \n", string(invokeOutput.Payload))
}
}
```

3. Node.js

3.1. Specify the instance concurrency

This topic describes how to use the SDK to specify the instance concurrency for a specified function.

SDK code samples

The following sample code uses SDK for Node.js as an example to specify the instance concurrency for a function:

```
// create function
var resp = await client.createFunction(serviceName, {
  functionName: funcName,
  handler: 'counter.handler',
  memorySize: 512,
  runtime: 'nodejs10',
  code: {
    zipFile: fs.readFileSync('/tmp/counter.zip', 'base64'),
  },
  instanceConcurrency: 10,
});

// update function
var resp = await client.updateFunction(serviceName, funcName, {
  instanceConcurrency: 20,
});

// get function
var resp = await client.getFunction(serviceName, funcName);
```