

Serverless HTTP 服务

操作指南







【版权声明】

©2013-2025 腾讯云版权所有

本文档(含所有文字、数据、图片等内容)完整的著作权归腾讯云计算(北京)有限责任公司单独所有,未经腾讯云事先明确书面许可,任何主体不得以任何 形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯,腾讯云将依法采取措施追究法律责任。

【商标声明】

🔗 腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算(北京)有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标,依法由权利人所有。未经腾讯云及 有关权利人书面许可,任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为,否则将构成对腾讯云及有关权利人商标权的侵犯,腾 讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况,部分产品、服务的内容可能不时有所调整。 您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定,除非双方另有约定,否则,腾讯云对本文档内容不做任何明示或默示 的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务,及相应的技术售后服务,任何问题请联系 4009100100或95716。





文档目录

操作指南

通过命令行完成框架部署 快速部署 Egg 框架 快速部署 Express 框架 快速部署 Flask 框架 快速部署 Koa 框架 快速部署 Laravel 框架 快速部署 Nestjs 框架 快速部署 Nextjs 框架 快速部署 Nuxtjs 框架



操作指南 通过命令行完成框架部署

最近更新时间: 2025-06-12 16:38:52

除了控制台之外,您也可以通过命令行快速部署 Web 框架,本篇文档将具体为您介绍,如何通过 Serverless 应用中心的 HTTP 组件,完成 Web 应用的 本地部署。

前提条件

已开通服务并完成 Serverless 应用中心的 权限配置。

支持框架

支持框架	相关文档
Express	快速部署 Express 框架
Коа	快速部署 Koa 框架
Egg	快速部署 Egg 框架
Next.js	快速部署 Nextjs 框架
Nuxt.js	快速部署 Nuxtjs 框架
Nest.js	快速部署 Nestjs 框架
Flask	快速部署 Flask 框架
Django	快速部署 Django 框架
Laravel	快速部署 Laravel 框架

操作步骤

1. 本地开发应用

根据您实际业务场景,本地完成开发,详情可参考 支持框架 开发文档。

2. 配置 yml 文件

在项目根目录下,新建 serverless.yml 文件,按照以下示例进行配置编写。全量配置请参见 配置文档。

```
# serverless.yml
component: http # (必选) 组件名称
name: webDemo # 必选) 组件实例名称.
inputs:
    region: ap-guangzhou # 云函数所在区域
    src: # 部署src下的文件代码,并打包成zip上传到bucket上
    src: ./ # 本地需要打包的文件目录
    exclude: # 被排除的文件或目录
        - .env
        - 'node_modules/**'
faas: # 函数配置相关
    framework: express #选择框架,此处以 express 为例
    runtime: Nodejs12.16
    name: webDemo # 云函数名称
    timeout: 10 # 超时时间,单位秒
    memorySize: 512 # 内存大小,默认 512 MB
    layers:
        - name: layerName # layer名称
```



version: 1 # 版本	
apigw: # # http 组件会默认帮忙创建一个 API 网关服务 isDisabled: false # 是否禁用自动创建 API 网关功能 id: service-xxx # api网关服务ID,不填则自动新建网关 name: serverless # api网关服务ID api: # 创建的 API 相关配置 cors: true # 允许跨域 timeout: 15 # API 超时时间 name: apiName # API 名称	
qualifier: \$DEFAULT # API 关联的版本	
protocols:	
- http	
- https	
environment: test	

3. 创建完成后,在根目录下执行 scf deploy 进行部署,组件会根据选择的框架类型,自动生成 scf_bootstrap 启动文件进行部署。

△ 注意:

由于启动文件逻辑与用户业务逻辑强关联,默认生成的启动文件可能导致框架无法正常启动,建议您根据实际业务需求,手动配置启动文件,详情 参考各框架的部署指引文档。

示例 scf_bootstrap

express

#!/usr/bin/env bash

/var/lang/node12/bin/node app.js

koa

#!/usr/bin/env bash

/var/lang/node12/bin/node app.js

• egg

```
#!/var/lang/nodel2/bin/node
/**
 * docker 中 node 路径: /var/lang/nodel2/bin/node
 * 由于 serverless 函数只有 /tmp 读写权限,所以在启动时需要修改两个环境变量
 * NODE_LOG_DIR 是为了改写 egg-scripts 默认 node 写入路径 (~/logs) -> /tmp
 * EGG_APP_CONFIG 是为了修改 egg 应有的默认当前目录 -> /tmp
 */
process.env.EGG_SERVER_ENV = 'prod';
process.env.NODE_LOG_DIR = '/tmp';
process.env.NODE_LOG_DIR = '/tmp';
process.env.EGG_APP_CONFIG = '{"rundir":"/tmp","logger":{"dir":"/tmp"}}';
const { Application } = require('egg');
// 如果通过层部署 node_modules 就需要修改 eggPath
Object.defineProperty(Application.prototype, Symbol.for('egg#eggPath'), {
```



```
value. /opt ,
});
const app = new Application({
    mode: 'single',
    env: 'prod',
});
app.listen(9000, '0.0.0.0', () => {
    console.log('Server start on http://0.0.0.0:9000')
});
```

nextjs

```
/*
# HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 9000
*/
const { nextStart } = require('next/dist/cli/next-start');
nextStart(['--port', '9000', '--hostname', '0.0.0.0']);
```

nuxtjs

```
#!/var/lang/node12/bin/node
/*
# HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 9000
*/
require('@nuxt/cli')
   .run(['start', '--port', '9000', '--hostname', '0.0.0.0'])
   .catch((error) => {
    require('consola').fatal(error);
    require('exit')(2);
   });
```

nestjs

```
#!/bin/bash
# SERVERLESS=1 /var/lang/node12/bin/npm run start -- -e /var/lang/node12/bin/node
SERVERLESS=1 /var/lang/node12/bin/node ./dist/main.js
• flask
```

#!/bin/bash

```
# HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 9000
/var/lang/python3/bin/python3 app.py
```

django

#!/bin/bash

HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 900



/var/lang/python3/bin/python3 manage.py runse

laravel

#!/bin/bash

- # **汪入** serverless **圿琼卜旳圿琼受重**
- ****
- # 注入 SERVERLESS 标识
- export SERVERLESS=1
- # 修改模板编译缓存路径,云函数只有 /tmp 目录可读写
- export VIEW_COMPILED_PATH=/tmp/storage/framework/views
- # 修改 session 以内存方式(数组类型)存储
- export SESSION_DRIVER=arra
- # **日志输出到** stdern
- export LOG_CHANNEL=stderr
- # 修改应用存储路径
- export APP_STORAGE=/tmp/storage

初始化模板缓存目录

- mkdir -p /tmp/storage/framework/views
- # HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 9000
- # 云端可执行文件路径 /var/lang/php7/bin/php
- /var/lang/php7/bin/php artisan serve --host 0.0.0.0 --port 9000



快速部署 Egg 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Egg 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见 通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署:一键部署 Egg 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 Egg 筛选函数模板,选择Egg 框架模板并单击下一步。如下图所示:

✿ Web 建 创建方式	該全新体验 无改造部署, 函数直接处理 HTTP 请求,体验产品写问卷,有机会获得精整 模板创建 使用示例模版快速创建一个函数或应用	\$礼品! <u>产品文档>> № 问卷入口>> </u> №	
莫糊搜索	Egg 多个过滤标签用回车键分隔	Q 共1个	推荐排序 🔹
	Egg 框架模版 查看详情 类別 函数 描述 基于 API 网关和 Web 函数,快 速部署 Egg 示例项目 标签 WebFunc Kodejs12.16 web 作者 企 勝汛云 部署 9,220次		
下 —北	間で当		

- 4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。
- 5. 单击**完成**即可创建函数。函数创建完成后,您可在**函数管理**页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Egg 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

函数管理	触发管理			
触发管理	创建触发器			
监控信息				
日志查询	API网关触发	所属别名:默认流量		
并发配额	API服务名	SCF_API_SERVICE 🕻		
部署日志	serviceId	service-		
	apild	api-		
	请求方法	ANY		
	发布环境	发布		
	鉴权方式	免鉴权		
	启用Base64编码	未启用		
	支持CORS	否		
	后端超时	15s		
	访问路径	https://service-ro2lw	ij.apigw.tencentcs.com/release/I	

7. 单击访问路径 URL,即可访问服务 Egg 项目。如下图所示:



自定义部署:快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 参考 Egg.js 官方文档,快速初始化示例项目。示例如下:

```
mkdir egg-example && cd egg-example
npm init egg --type=simple
npm i
```

2. 在根目录下,执行以下命令在本地直接启动服务。

```
npm run dev
open http://localhost:7001
```

3. 打开浏览器访问 http://localhost:7001 ,即可在本地完成 Egg 示例项目的访问。

部署上云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下三步:

- 修改监听地址与端口为 0.0.0.0:9000 。
- 修改写入路径,serverless 环境下只有 /tmp 目录可读写。



• 新增 scf_bootstrap 启动文件。

具体步骤如下:

1. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于配置环境变量和启动服务,此处仅为示例,具体操作请以您实际业务场 景进行调整):

```
'use strict';
/**
 * docker 中 node 路径: /var/lang/node12/bin/node
 * 由于 serverless 函数只有 /tmp 读写权限, 所以在启动时需要修改两个环境变量
 * NODE_LOG_DIR 是为了改写 egg-scripts 默认 node 写入路径 (~/logs) -> /tmp
 * EGG_APP_CONFIG 是为了修改 egg 应有的默认当前目录 -> /tmp
 */
process.env.BGG_SERVER_ENV = 'prod';
process.env.NODE_LOG_DIR = '/tmp';
process.env.NODE_LOG_DIR = '/tmp';
process.env.EGG_APP_CONFIG = '{"rundir":"/tmp","logger":{"dir":"/tmp"}}';
const { Application } = require('egg');
// 如果通过层部署 node_modules 就需要修改 eggPath
Object.defineProperty(Application.prototype, Symbol.for('egg#eggPath'), {
    value: '/opt',
    ));
const app = new Application({
    mode: 'single',
    env: 'prod',
    ));
app.listen(9000, '0.0.0.0', () => {
    console.log('Server start on http://0.0.0.0:9000');
    ));
```

2. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

hmod 777 scf_bootstrap

- 3. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 4. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



5. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

创建方式			
030273320		自定义创建	
	使用示例模版快速创建一个图数或应用	使用helloworld示例目定义创建函数	
基础配置			
函数类型 *	◯ 事件函数 <mark>○</mark> Web函数 ④		
函数名称 *	ExpressDemo		
	只能包含字母、数字、下划线、连字符,以字	母开头,以数字或字母结尾,2~60个字符	
地域 *	S 广州 ·		
部署方式 *	● 代码部署 🦳 镜像部署		
运行环境 *	Nodejs12.16 v		
函数代码	 上传项目前,请修改您的项目监听端口为900 	0	
提交方法 *	🔵 在线编辑 🔹 本地上传zip包 🔹 🔵 本地	3上传文件夹 🔵 通过cos上传zip包	
函数代码:	index.zip 压缩完成 重	新上传	
	请选择文件夹,最大支持250M		

- 函数类型:选择"Web 函数"。
- 函数名称:填写您自己的函数名称。
- 地域:填写您的函数部署地域,默认为广州。
- **运行环境**:选择 "Nodejs 12.16" 。
- 部署方式:选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:** 选择函数代码在本地的具体文件夹。
- 6. 单击**完成**完成项目的部署。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。



快速部署 Express 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Express 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署:一键部署 Express 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 WebFunc 筛选所有 Web 函数模板,选择 Express 框架模板并单击下一步。如下图所示:

	模板创 使用示(建自定义创建 例模版快速创建一个函数或应用 使用helloworld示例目和	创建函数	
索	webfun	S 多个过滤标签用回车键分隔	Q 共14个	推荐排序
	Node	12 Web 模版 查看详情	Express 框架模版 查看详情 Node10 Web 模版	查看详
	类别	函数	类别 函数 类别 函数	
	描述	基于 Node.js 12, 使用 API 网关和 Web 函数, 快 速部署 Hello world 示例项目	描述 基于 API 网关和 Web 函数,快速部署 Express 示 描述 基于 Node.js 10, 使用 A 递邮署 Hello world 示例II,	PI 网关和 Web 函数,快 旧
	标签	WebFunc helloworld Nodejs12.16 web	标签 WebFunc Express Nodejs12.16 标签 WebFunc hellowork web web	Nodejs10.15
	作者	🔗 腾讯云	作者 🔗 勝讯云 作者 🔗 腾讯云	
	部署	8,753次	部署 8,181次 部署 7,991次	
	Koa Ħ	E架模版 查看详情	Php5 Web 函数模版 查看详情 Nextjs 框架模版	查看详
	类别	函数	类别 函数 类别 函数	
	描述	基于 API 网关和 Web 函数,快速部署 Koa 示例项 目	描述 基于 Php5,使用 API 网关和 Web 函数,快速部 描述 基于 API 网关和 Web 函数 署 Hello world 示例项目 项目	t,快速部署 Nextjs 示例
	标签	WebFunc Koa Nodejs12.16 web	标签 WebFunc helloworld Php5 web 标签 WebFunc Nextjs	Nodejs12.16 SSR
	作者	🔗 腾讯云	作者 🔗 勝讯云 web	
			作者 🔿 勝田云	

4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。

5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL, 访问您部署的 Express 项目。单击左侧菜单栏中的触发管理, 查看访问路径。如下图所示:

← ExpressDemo	正常		
函数管理	触发管理		
触发管理	创建触发器		
监控信息 日志查询	API网关触发	所属别名: 默认流量	
并发配额	API服务名	SCF_API_SERVICE	
部署日志	serviceld	service	
	apild	api-	
	请求方法	ANY	
	发布环境	发布	
	鉴权方式	免鉴权	
	启用Base64编码	未启用	
	支持CORS	否	
	后端超时	15s	
	访问路径	https://service	.gz.apigw.tencentcs.com/release/1

7. 单击访问路径 URL,即可访问服务 Express 项目。如下图所示:



自定义部署: 快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 执行以下命令安装 Express 框架和 express-generator 脚手架,并初始化 Express 示例项目。

2. 执行以下命令,进入项目目录并安装依赖包。

cd WebApp npm install

3. 安装完成后,执行以下命令在本地直接启动服务。

ıpm start

4. 打开浏览器访问 http://localhost:3000 ,即可在本地完成 Express 示例项目的访问。



部署上云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下两步:

- 修改监听地址与端口为 0.0.0.0:9000 。
- 新增 scf_bootstrap 启动文件。

具体步骤如下:

1. 在 Express 示例项目中,可通过 ./bin/www 设置监听地址与端口,打开该文件可以发现,通过环境变量可以设置指定监听端口,否则将自动监听 3000端口。如下图所示:



2. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于配置环境变量和启动服务):



3. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

chmod 777 scf_bootstrap

4. 本地配置完成后,执行以下命令启动文件(如下命令为在该文件 scf_bootstrap 目录下执行时示例),确保您的服务在本地可以正常启动。

./scf_bootstrap

- 5. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 6. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



7. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

创建方式	模板创建 使用示例模版快速创建一个函数或应用	自定义创建 使用helloworld示例自定义创建函数	
基础配置			
函数类型 *	◯ 事件函数 ○ Web函数 (j)		
函数名称 *	ExpressDemo		
	只能包含字母、数字、下划线、连字符,以3	字母开头,以数字或字母结尾,2~60个字符	
地域 *	◎ 广州		
部署方式 *	○代码部署 ○镜像部署		
运行环境 *	Nodejs12.16 v		
函数代码 (〕上传项目前,请修改您的项目监听端口为90	100	Ŧ
提交方法 *	○ 在线编辑 ○ 本地上传zip包 ○ 本	地上传文件夹 🦳 通过cos上传zip包	
○○※ケノンズⅡ ◆			
ELAXI (N-) "	index.zip 压缩完成		
	I用A2I并又计大,取入又存230WI		

- **函数名称:**填写您自己的函数名称。
- 地域:填写您的函数部署地域,默认为广州。
- 运行环境:选择 "Nodejs 12.16"。
- **部署方式:**选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:**选择函数代码在本地的具体文件夹。

8. 单击完成完成 Express 项目的部署。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。如下图所示:



访问路径	nttps://service-	cd.apigw.tencentcs.com/release/ 🗹		测试
测试模板 请求方式 path	GET /	▼	返回结果 说 返回码 响应延时	明文档 🖸 200 780ms
headers	key 请输入key	value 请输入value	ni <u>niv</u> bouy	html <head> <title>Express</title> <link href="/stylesheets/style.css" rel="stylesheet"/> </head> <body><h1>Express</h1> Welcome to Express</body>
params	key 请输入key	value 请输入value	响应Headers	X-Powered-By:Express Content-Type:text/html; charset=utf-8 X-Api- Requestid:ddddf292c153113e4d5425 Cache-Control:max-age=3600 Expires:Tue, 15 Jun 2021 02:21:30 GMT Vary:Accept-Encoding Content-Length:170 Connection:keep-alive Etag:W/"aa-z+ebXSEdArbZ+EXIN/V



快速部署 Flask 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 SCF Web Function,快速部署您的 Flask 业务上云。

 说明: 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署: 一键部署 Flask 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 WebFunc 筛选所有 Web 函数模板,选择 Flask 框架模板并单击下一步。如下图所示:

模板创建 使用示例模版	从头开始 快速创建一个函数或应用	使用容器镜像 基于容器镜像来创建函数	
模糊搜索	WebFunc 多个过滤标签用回车键分隔	Q 共32个	推荐排序. *
	Express 框架模版 查看详情 类别 函数	Flask 框架模版 查看详情 类别 函数	Nextjs 框架模版 查看详情 类則 函数
	描述 基于 API 网关和 Web 函数,快速部署 Express 示例 项目	描述 基于 API 网关和 Web 函数,快速部署 Flask 示例项 目	描述 基于 API 网关和 Web 函数,快速邮署 Nextjs 示例项 目
	标签 WebFunc Express Nodejs12.16 web 作者 分 藤讯云 部署 20.108次	标签 WebFunc Flask Python3.6 web 作者 仑 膳讯云 部署 13.931次	标签 WebFunc Nextjs Nodejs12.16 SSR web 作者 全 腾讯云
			部署 10,007次

- 4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。
- 5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过自动生成的访问路径 URL,访问您部署的 Flask 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← flask_demo	正常	函数服务帮助
函数管理	触发管理	
触发管理		
监控信息		
日志查询	的建始发播	
并发配额	默认触发器 触发别名:默认流量	11
部署日志	访问路径 ① 公网 https://servicegz.apigw.tencentcs.com/release/Г	
	协议支持 HTTP&HTTPS	
	请求方法 ANY	
	发布环境 发布	
	鉴权方式 免鉴权	
	标签 未自用	
	您可以通过升级至API网关标准版,享受更多API网关高级能力,该升级操作不可回退,详情参考文档 🗹	

7. 单击访问路径 URL,即可访问服务 Flask 项目。如下图所示:

https://service-	.gz.apigw.tencentcs.com/release/	₿ ✿
	欢迎访问 Flask 应用	
	<u>腾讯云 Serverless</u> 为您提供服务	

自定义部署:快速迁移本地项目上云

本地开发

1. 执行以下命令,确认您本地的环境已安装好 Flask。

pip install Flask

2. 在本地创建 Hello World 示例项目。

在项目目录下,新建 app.py 文件,用于实现 Hello World 应用,示例代码如下:



3. 在本地执行 python3 app.py 命令运行 app.py 文件。示例如下:





4. 打开浏览器访问 http://127.0.0.1:5000 ,即可在本地完成 Flask 示例项目的访问。如下图所示:



部署上云

接下来执行以下步骤,对本地已创建完成的项目进行简单修改,使其可以通过 Web Function 快速部署,对于 Flask,具体修改步骤如下: 1. **安装依赖包**

1.1 由于 SCF 云上标准环境内未提供 Flask 依赖库,此处您必须将依赖文件安装完成后,与项目代码一起打包上传。请先在本地新建 requirements.txt 文件,文件内容如下:



由于 SCF 内置运行环境版本 (Python 3.6) 限制,werkzeug 只能使用低版本(<=1.0.x),高版本可能无法正常运行,函数运行环境版本 升级已在规划中,敬请期待。

1.2 执行以下命令进行安装:

pip install -r requirements.txt

2. 修改监听地址与端口

在 Web 函数内,限制了监听端口必须为9000,因此需要修改监听地址端口为 0.0.0.0:9000 ,如下图所示:



 ① 说明:

 您也可以在 scf_bootstrap

 中,通过环境变量配置监听端口。

3. 新增 scf_bootstrap 启动文件



在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于完成环境变量配置,指定服务启动命令等自定义操作,确保您的服务可 以通过该文件正常启动):



9. 单击**完成**完成 Flask 项目的部署。

开发管理



部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。如下图所示:

访问路径	https://service	.cd.apigw.tencentcs.com/release/ 🗳			I	测试
测试模板 请求方式	GET	▼	返 应	回结果 说明 回码	文档 ピ 200 23ms	
path headers	/ key	value	ца); Да),	应Weby	Hello World	
params	请输入key	请输入value	响,	应Headers	Cache-Control:max-age=3600 Expires:Tue, 22 Jun 2021 02:57:18 GMT Vary:Accept-Encoding	
	key 请输入key	ii输入value			Content-Type:text/html; charset=utf-8 Content-Length:11 Connection:keep-alive X-Api- Requestid:a93727b2052db61537b70	



快速部署 Koa 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Koa 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见 通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署:一键部署 Koa 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 koa 筛选函数模板,选择Koa 框架模板并单击下一步。如下图所示:

र्मेंटर	模板创建 使用示例模版快速创建一个函数或应用 使历	定义创建 Bhelloworld示例自定义创建函数		
要素	koa 多个过滤标签用回车键分隔	Q	共1个	推荐排序 🔻
	ROUAL LE未代集AX 直目伴悄 美別 函数 描述 基于 API 网关和 Web 函数,快 遠部署 Koa 示例项目 标签 WebFunc Koa Nodejs12.16 web			

- 4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。
- 5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Koa 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← koa_demo-	正常			函数服务帮助文档 🗹
函数管理	触发管理			
触发管理	创建缺发器			
监控信息				
日志查询	API购关触友 所属别名:	默认流量		
并发配额	API服务名 SCF_A	PI_SERVICE		
部署日志	serviceld service			
	apild api-			
	请求方法 ANY			
	发布环境 发布			
	鉴权方式 免鉴权			
	启用Base64编码 未启用			
	支持CORS 否			
	后端超时 15s			
	访问路径 https://	service .gz.	apigw.tencentcs.com/release/ 🗗	

7. 单击访问路径 URL,即可访问服务 Koa 项目。如下图所示:



自定义部署:快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 参考 Koa.js 官方文档,安装 Koa 环境并初始化您的 Koa 项目,此处以 hello world 为例, app.js 内容如下:

<pre>const Koa = require('koa');</pre>
const app = new Koa();
const main = ctx => {
<pre>ctx.response.body = 'Hello World',</pre>
app.use(main);
<pre>app.listen(3000);</pre>

2. 在根目录下,执行以下命令在本地直接启动服务。

ode app.js



3. 打开浏览器访问 http://localhost:3000 ,即可在本地完成 Koa 示例项目的访问。

部署上云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下两步:

- 修改监听地址与端口为 0.0.0.0:9000 。
- 新增 scf_bootstrap 启动文件。

具体步骤如下:

1. 在 Koa 示例项目中,修改监听端口到 9000 。如下图所示:

2. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于配置环境变量和启动服务):



3. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

mod 777 scf_bootstrap

- 4. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 5. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



6. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

创建方式	模板创建	自定义创建
	使用示例模版快速创建一个函数或应用	使用helloworld示例自定义创建函数
基础配置		
函数类型 *	◯ 事件函数 <mark>●</mark> Web函数 ④	
函数名称 *	ExpressDemo	
		母开头,以数字或字母结尾,2~60个字符
地域 *	⑤ 广州 ▼	
部睪方式 *	● 代码部署 ○ 镜像部署	
运行环境*	Nodejs12.16 V	
函数代码	① 上传项目前,请修改您的项目监听端口为900	0
提交方法 *	○ 在线编辑 ○ 本地上传zip包 ○ 本地	3上传文件夹 🔵 通过cos上传zip包
函数代码 *	index.zip 压缩完成 重	新上传
	请选择文件夹,最大支持250M	
○ 函数类型: →	选择"Web 函数"。	

- **函数名称:**填写您自己的函数名称。
- 地域:填写您的函数部署地域,默认为广州。
- 运行环境:选择 "Nodejs 12.16"。
- **部署方式:**选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:**选择函数代码在本地的具体文件夹。

7. 单击完成完成 Koa 项目的部署。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。如下图所示:



访问路径	https://servic	:6.bj.apigw.tencentcs.com	/release/ 🖸			测试
测试模板				返回结果 说明	文档 🖸	
请求方式	GET		•	返回码	200	
path	1			响应延时	143ms	
headers	key	value		响应Body	Hello World	
	请输入key	请输入value		响应Headers	X-Api-Requestid:906acebc95e4c9e4 Cache-Control:max-age=3600	
params	key	value			Expires:Mon, 19 Jul 2021 11:36:54 GMT Vary:Accept-Encoding	
	请输入key	请输入value			Content-Type:text/plain; charset=utf-8 Content-Length:11 Connection:keep-alive	



快速部署 Laravel 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文档指导您如何通过 Web 函数,快速迁移本地的 Laravel 服务上云。

 说明: 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署:一键部署 Laravel 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里筛选 WebFunc ,筛选所有 Web 函数模板,选择 Laravel 框架模板并单击下一步。如下图所示:

使用示例模版快速创建一个函数或应用使用helloworl	id示例自定义创建函数	
WebFunc 多个过滤标签用回车键分隔	Q 共15个	推荐排序
Express 框架模版 查看详情	Flask 框架模版 查看详情	Nextjs 框架模版 查看详
类别 函数	类别 國数	类别 函数
描述 基于 API 网关和 Web 函数,快速部署 Express 示例项目	描述 基于 API 网关和 Web 函数,快速部署 Flask 示例项目	描述 基于 API 网关和 Web 函数,快速部署 Nextjs 示例项目
标签 WebFunc Express	标签 WebFunc Flask Python3.6	标签 WebFunc Nextjs
Nodejs12.16 web	web	Nodejs12.16 SSR web
作者 🛛 🔗 購讯云	作者 🛛 🏠 購訊云	作者 🛛 修讯云
部署 11,203次	部署 8,678次	部署 8,957次
Koa 框架模版 查看详情	Node12 Web 模版 查看详情	Laravel 框架模版 查看详
美别 國數	类别 函数	类别 函数
描述 基于 API 网关和 Web 函数,快速部署	描述 基于 Node.js 12,使用 API 网关和 Web 函数,快速部署 Hello world 示例项目	描述 基于 API 网关和 Web 函数,快速部署 Laravel 示例项目
Koa 示例坝目		
Koa 示例项目 标签 WebFunc Koa Nodejs12.16	标签 WebFunc helloworld	标签 WebFunc Laravel Php7
Koa 示例项目 标签 WebFunc Koa Nodejs12.16 web	标签 WebFunc helloworld Nodejs12.16 web	标签 WebFunc Laravel Php7 web
Koa 示例项目 标签 WebFunc Koa Nodejs12.16 web	标签 WebFunc helloworld Nodejs12.16 web 作者 ☎ 購讯云	标签 WebFunc Laravel Php7 web 作者 仑 腾讯云

4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。

5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Laravel 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← laravel_demo	正常			函数服务帮助文档 🗹
函数管理	触发管理			
触发管理	创建触发器			
监控信息				
日志查询	API网关触发	所属别名: 默认流量		删除
并发配额	API服务名	SCF_API_SERVICE		
部署日志	serviceId	service-		
	apild	api		
	请求方法	ANY		
	发布环境	发布		
	鉴权方式	免鉴权		
	启用Base64编码	未启用		
	支持CORS	否		
	后端超时	15s		
	访问路径	https://service	.gz.apigw.tencentcs.com/release/ 🗗	

7. 单击访问路径 URL,即可访问服务 Laravel 项目。如下图所示:



自定义部署:快速迁移本地项目上云

本地开发

- 1. 参考 Laravel 官方文档,在本地环境中完成 Laravel 的开发环境搭建。
- 2. 在本地创建 Laravel 示例项目。进入项目目录下,执行以下命令,初始化 Laravel 示例应用:

oser create-project --prefer-dist laravel/laravel blog

3. 执行以下命令,在本地启动示例项目。示例如下:

```
$ php artisan serve --host 0.0.0.0 --port 9000
Laravel development server started: <http://0.0.0.0:9000>
[Wed Jul 7 11:22:05 2021] 127.0.0.1:54350 [200]: /favicon.icc
```

4. 打开浏览器访问 http://0.0.0.0.9000 ,即可在本地完成 Laravel 示例项目的访问。如下图所示:

← → C ▲ 不安全 0.0.0.0:9000					🖈 🗟 🖗 O 🏞 🕢 i
		Ldld\	/et		
				CITHUR	
	DOCS LARACAST	S NEWS BLOG	NOVA TONGE	GIIIOB	

部署上云

腾讯云

在该文件添加如下内容(用于配置环境变量和启动服务,此处仅为示例,具体操作请以您实际业务场景来调整):

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,具体修改步骤如下:

1. 新增 scf_bootstrap 启动文件

```
在项目根目录下新建 scf_bootstrap 启动文件,在该文件完成环境变量配置,指定服务启动命令等自定义操作,确保您的服务可以通过该文件正常启动。
```

```
    ⚠ 注意:

            scf_bootstrap 必须有 755 或者 777 的可执行权限。
            如需在日志中输出环境变量,需在启动命令前需要加 -u 参数,例如 python -u app.py 。
```

2. 修改文件读写路径

由于在 SCF 环境内,只有 /tmp 文件可读写,其他目录会由于缺少权限而写入失败,因此需要在 scf_bootstrap 里,以环境变量的方式注入,调 整 Laravel 框架的输出目录:

```
#!/bin/bash
# 注入 SERVERLESS 标识
export SERVERLESS=1
# 修改模板编译缓存路径,云函数只有 /tmp 目录可读写
export VIEW_COMPILED_PATH=/tmp/storage/framework/views
# 修改 session 以内存方式(数组类型)存储
export SESSION_DRIVER=array
# 日志输出到 stderr
export LOG_CHANNEL=stderr
# 修改应用存储路径
export APP_STORAGE=/tmp/storage
# 初始化模板缓存目录
```



kdir -p /tmp/storage/framework/views

3. 修改监听地址与端口

在Web函数内,限制了监听端口必须为 9000 ,因此需要在 scf_bootstrap 中通过以下命令指定监听端口:

/var/lang/php7/bin/php artisan serve --host 0.0.0.0 --port 900

完整 scf_bootstrap 内容如下:

Ð	🖬 scf_	bootstrap ×
	src >	scf_bootstrap
\cap	1	#!/bin/bash
\succ	2	
	3	*****************************
90	4	# 注入 serverless 环境下的环境变量
8	5	*****************************
	6	# 注入 SERVERLESS 标识
	7	export SERVERLESS=1
	8	# 修改模板编译缓存路径, 云函数只有 /tmp 目录可读写
	9	<pre>export VIEW_COMPILED_PATH=/tmp/storage/framework/views</pre>
	10	# 修改 session 以内存方式(数组类型)存储
	11	export SESSION_DRIVER=array
	12	# 日志输出到 stderr
	13	export LOG_CHANNEL=stderr
	14	# 修改应用存储路径
	15	<pre>export APP_STORAGE=/tmp/storage</pre>
	16	
	17	# 初始化模板缓存目录
	18	mkdir -p /tmp/storage/framework/views
	19	
	20	# HTTP 直通函数由于是基于 docker 镜像运行,所以必须监听地址为 0.0.0.0,并且端口为 9000
	21	# 云端可执行文件路径 /var/lang/php7/bin/php
	22	/var/lang/php7/bin/php artisan serve ——host 0.0.0.0 ——port 9000
	23	

4. 部署上云

本地配置完成后,执行启动文件,确保您的服务可以本地正常启动。执行以下步骤部署 Laravel:

- 4.1 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 4.2 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 4.3 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

新建	
🖒 Web 建	站全新体验 无政造部署,函数直接处理 HTTP 请求,体验产品写问卷,有机会获得精善礼品! <u>产品文档>></u> 🗹 <u>问卷入口>></u> 亿
创建方式	
	使用示例模版快速创建一个函数或应用 使用helloworld示例自定义创建函数
基础配置	
函数类型 *	○事件函数 ○ Web函数 ③
函数名称 *	LaravelDemo
	只能包含字母、数字、下划线、连字符,以字母开头,以数字或字母结尾,2~60个字符
地域 *	③ 成都 *
部署方式*	●代码部署 ○ 镜像部署
运行环境 *	Php7
○ 函数类	型:选择"Web 函数"。

- **函数名称**:填写您自己的函数名称。
- **地域**:填写您的函数部署地域,例如成都。



- **部署方式:**选择"代码部署",上传您的本地项目。
- 运行环境:选择 "Php7"。

4.4 部署完成后,单击生成的 URL,即可访问您的 Laravel 应用。如下图所示:



开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能如层绑定、日志管理等,享受 Serverless 架构带来的低成 本、弹性扩缩容等优势。

访问路径	https://service-c	.apigw.tencentcs.com/n	elease/ 🗹		测试
测试模板				返回结果 说	胡文档 🖸
请求方式	GET			返回码	200
path	1			响应延时	4637ms
headers	key 请输入key	value 请输入value		响应Body	html <html lang="en"> <head> <meta charset="utf-8"/> <meta content="wid</td></tr><tr><td>params</td><td>key
请输入key</td><td>value
请输入value</td><td></td><td></td><td>th=device-width, initial-scale=1" name="viewport"/> <title>Laravel</title></head></html>
					Fonts <link href="https://fonts.googleapis.co m/css?



快速部署 Nestjs 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Nest.js 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见 通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署: 一键部署 Nest.js 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 nest 筛选函数模板,选择 Nest 框架模板并单击下一步。如下图所示:

	模板创建 使用示例模版快速创建一个的	自定义创建 函数或应用使用helloworld示例自知	定义创建函数		
素	nest 多个过滤标签用回车	罐分隔		Q 共1个	推荐排序 ▼
	描述 基于 API 网关和 W 速部署 Nest 示例项 标签 WebFunc Nes Nodejs12.16 v	leb 函数,快 1月 st web			
	作者 🛛 🔗 購讯云				

- 4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。
- 5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Nest.js 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← NestDemo-	正常		函数服务帮助文档 🗹
函数管理	触发管理		
触发管理	创建触发器		
监控信息			
日志查询	API网关触发	所属别名:默认流量	删除
并发配额	API服务名	SCF_API_SERVICE 🗹	
部署日志	serviceId	service-	
	apild	api-	
	请求方法	ANY	
	发布环境	发布	
	鉴权方式	免鉴权	
	启用Base64编码	未启用	
	支持CORS	좀	
	后端超时	15s	
	访问路径	https://servicegz.apigw.tencentcs.com/release/I	

7. 单击访问路径 URL,即可访问服务 Nest.js 项目。如下图所示:



自定义部署:快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 参考 Nest.js 官方文档,初始化您的 Nest.js 项目:

```
npm i -g @nestjs/cli
nest new nest-app
```

2. 在根目录下,执行以下命令在本地直接启动服务。

cd nest-app && npm run start

3. 打开浏览器访问 http://localhost:3000 ,即可在本地完成 Nest.js 示例项目的访问。如下图所示:

 $\leftarrow \rightarrow$ C \triangle (S http://localhost:3000

Hello World!



部署上云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下两步:

- •新增 scf_bootstrap 启动文件。
- 修改监听地址与端口为 0.0.0.0:9000 。

具体步骤如下:

1. 修改启动文件 ./dist/main.js , 监听端口改为 9000 。如下图所示:



2. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于启动服务):

```
#!/bin/bash
SERVERLESS=1 /var/lang/node12/bin/node ./dist/main.js
```

△ 注意:

- 此处仅为示例启动文件,具体请根据您的业务场景进行调整。
- 示例使用的是云函数标准 Node 环境路径,本地调试时,需修改成您的本地路径。
- 3. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

hmod 777 scf_bootstrap

- 4. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 5. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



6. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

创建方式		
B1/E/101-4	模板创建	自定义创建
	使用示例模版快速创建一个函数或应用	使用helloworld示例自定义创建函数
基础配置		
函数类型 *	🔵 事件函数 🛛 🔵 Web函数 🕠	
函数名称 *	ExpressDemo	
		4田开头,以数字或字母结尾,2~60个字符
地域 *	◎ 广州 ▼	
部書方式*		
运行环境 *	Nodejs12.16 v	
函数代码	 上传项目前,请修改您的项目监听端口为900 	• 00
提交方法 *	🔵 在线编辑 🔹 本地上传zip包 🔹 🔵 本地	8上传文件夹 🔵 通过cos上传zip包
函数代码 *	index.zip 压缩完成 重	新上传
	请选择文件夹,最大支持250M	

- O 函数名称:填写您自己的函数名称。
- **地域**:填写您的函数部署地域,默认为广州。
- 运行环境:选择 "Nodejs 12.16"。
- 部署方式:选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:**选择函数代码在本地的具体文件夹。

7. 单击完成完成 Nest.js 项目的部署。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。



快速部署 Nextjs 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Next.js SSR 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署: 一键部署 Next.js 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 webfunc 筛选函数模板,选择 Next.js 框架模板并单击下一步。如下图所示:

+0=											
送系	webfun	IC 多个过滤标签用回车键分隔					Q	共15个		推荐排序	*
	Expre	ess 框架	查看详情	Flask	框架模版		查看详情	Nextjs	框架模版	查看	洋情
	类别	函数		类别	函数			类别	函数		
	描述	基于 API 网关和 Web 函数, 部署 Express 示例项目	快速	描述	基于 API 网关 部署 Flask 示	和 Web 函数, 列项目	快速	描述	基于 API 网关和 W 部署 Nextjs 示例项	'eb 函数,快速 目	
	标签	WebFunc Express Nodejs12.16 web		标签	WebFunc Python3.6	Flask web		标签	WebFunc Net Nodejs12.16	xtjs SSR web	
	作者	∞ 腾讯云		作者	🔗 腾讯云			作者	🔗 腾讯云		
	部署	11,206次		部署	8,683次			部署	8,959次		
	Koa ħ	医架模版 1	直看详情	Node	12 Web		查看详情	Larave	el 框架模版	查看	洋情
	类别	函数		类别	函数			类别	函数		
	描述	基于 API 网关和 Web 函数, 部署 Koa 示例项目	快速	描述	基于 Node.js 1 和 Web 函数,	12, 使用 API (快速部署 Hel	网关 Io	描述	基于 API 网关和 W 部署 Laravel 示例项	'eb 函数,快速 页目	
	标签	WebFunc Koa Nodejs12.16 web		标签	WebFunc Nodejs12.16	helloworld web		标签	WebFunc Lar Php7 web	avel	
	作者	⊘ 腾讯云		作者	🔗 腾讯云			作者	🔗 腾讯云		
	部署	9,227次		部署	9,834次			部署	9,060次		

4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。

5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Next.js 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← nextjs_demo-	正常		函数服务帮助文档 2	1
函数管理	触发管理			
触发管理	创建触发器			
监控信息				
日志查询	API网关触发,	所属别名: 默认流量	删除	
并发配额	API服务名	SCF_API_SERVICE 🛛		
部署日志	serviceId	service		
	apild	api		
	请求方法	ANY		
	发布环境	发布		
	鉴权方式	免鉴权		
	启用Base64编码	未启用		
	支持CORS	否		
	后端超时	15s		
	访问路径	https://service-	.gz. apigw.tencentcs.com/release/ I	

7. 单击访问路径 URL,即可访问服务 Next.js 项目。如下图所示:

欢迎访问 Next.js 应用

() 说明:

由于 Nextjs 框架每次部署前需要重新构建,请确保本地更新代码并且重新 build 之后再进行部署。

自定义部署:快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 参考 Next.js 官方文档,安装并初始化您的 Next.js 项目:

npx create-next-app

2. 在根目录下,执行以下命令在本地直接启动服务。

cd my-app && npm run dev

3. 打开浏览器访问 http://localhost:3000 ,即可在本地完成 Next.js 示例项目的访问。如下图所示:



部署上云

腾讯云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下两步:

- 修改监听地址与端口为 0.0.0.0:9000 。
- 新增 scf_bootstrap 启动文件。

具体步骤如下:

1. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于启动服务并指定启动端口):

```
#!/var/lang/node12/bin/node
const { nextStart } = require('next/dist/cli/next-start');
nextStart([ '--port', '9000', '--hostname', '0.0.0.0' ])
```

△ 注意:

- 此处仅为示例启动文件,具体请根据您的业务场景进行调整。
- 示例使用的是云函数标准 Node 环境路径,本地调试时,需修改成您的本地路径。
- 2. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

d 777 scf_bootstrap

- 3. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 4. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



5. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

(2由			
1)建力式	模板创建	自定义创建	
	使用示例模版快速创建一个函数或应用	使用helloworld示例自定义创建函数	
基础配置			
数类型 *	🔵 事件函数 🛛 🔵 Web函数 🚯		
数名称 *	ExpressDemo		
		母开头,以数字或字母结尾,2~60个字符	
3域*	◎ 广州		
3署方式 *	● 代码部署 ○ 镜像部署		
行环境 *	Nodejs12.16 v		
5 W6 (15 T2)			
9521709	让传项目前,请修改恶的项目监听端口为90	0	
交方法 *	◯ 在线编辑 ◯ 本地上传zip包 ◯ 本ま	8上传文件夹 🔿 通过cos上传zip包	
國物件記。			
퍼섯시 가는 '	index.zip 压缩完成 重	新上传	
	请选择文件夹,最大支持250M		
函数类型:	选择"Web 函数"。		
函数名称:	填写您自己的函数名称。		

- **地域**:填写您的函数部署地域,默认为广州。
- 运行环境:选择"Nodejs 12.16"。
- 部署方式:选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:** 选择函数代码在本地的具体文件夹。
- 6. 单击**完成**完成 Next.js 项目的部署。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。



Welcome to Next.js!

Get started by editing pages/index.js

Documentation \rightarrow

apigw.tencentcs.co

Find in-depth information about Next.js features and API.

Examples →

Discover and deploy boilerplate example Next.js projects.

Learn →

Learn about Next.js in an interactive course with quizzes!

Deploy →

Instantly deploy your Next.js site to a public URL with Vercel.



快速部署 Nuxtjs 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Nuxt.js SSR 项目快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署: 一键部署 Nuxt.js 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 webfunc 筛选函数模板,选择 Nuxt.js 框架模板并单击下一步。如下图所示:

模倣回建 目定义 使用示例模版快速创建一个函数或应用 使用he	-BU建 lloworld示例自定义创建函数	
webfunc 多个过滤标签用回车键分隔	(Q 共15个 技部署次数据
Python3 Web 查看详情	Express 框架 查看详情	Node12 Web 查看
类别 图数	类别 图数	类别 國数
描述 基于 Python3,使用 API 网关和 Web 函数,快速部署 Hello worl	描述 基于 API 网关和 Web 函数, 快速 部署 Express 示例项目	描述 基于 Node.js 12,使用 API 网关 和 Web 函数,快速邮署 Hello
标签 WebFunc helloworld	标签 WebFunc Express	标签 WebFunc helloworld
Python3.6 web	Nodejs12.16 web	Nodejs12.16 web
作者 🔗 騰讯云	作者 🛛 勝讯云	作者 🔗 廣讯云
部署 11,532次	部署 11,208次	部署 9,834次
Nuxtjs 框架模版 查看详情	Php7 Web 函 查看详情	Koa 框架模版 查看
與别 函数	與別 函数	英别 函数
描述 基于 API 网关和 Web 函数,快速 部署 Nuxtjs 示例项目	描述 基于 Php7,使用 API 网关和 Web 函数,快速部署 Hello world 示	描述 基于 API 网关和 Web 函数,快速 部署 Koa 示例项目
标签 WebFunc Neuxtjs	标签 WebFunc helloworld	标签 WebFunc Koa
Nodejs12.16 SSR web	Php7 web	Nodejs12.16 web
作者 🛛 膠讯云	作者 🔗 腾讯云	作者 🔗 購讯云
部署 9.638次	部署 9,555次	部署 9,227次

4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。

5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Nuxt.js 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← nuxtjs_demo-	正常		函数服务帮助文档 忆
函数管理	触发管理		
触发管理	创建触发器		
监控信息			
日志查询	API购关触友	所属别名:默认流量	 脚除
并发配额	API服务名	SCF_API_SERVICE 🗹	
部署日志	serviceId	service-	
	apild	api-	
	请求方法	ANY	
	发布环境	发布	
	鉴权方式	免鉴权	
	启用Base64编码	未启用	
	支持CORS	否	
	后端超时	15s	
	访问路径	https://service .gz.apigw.tencentcs.com/release/lig	

7. 单击访问路径 URL,即可访问服务 Nuxt.js 项目。如下图所示:



() 说明:

由于 Nuxtjs 框架每次部署前需要重新构建,请确保本地更新代码并且重新 build 之后再进行部署。

自定义部署:快速迁移本地项目上云

前提条件

本地已安装 Node.js 运行环境。

本地开发

1. 参考 Nuxt.js 官方文档,安装并初始化您的 Nuxt.js 项目:

npx create-nuxt-app nuxt-app

2. 在根目录下,执行以下命令在本地直接启动服务。

ed nuxt-app && npm run dev



3. 打开浏览器访问 http://localhost:3000 ,即可在本地完成 Nuxt.js 示例项目的访问。如下图所示:



部署上云

接下来执行以下步骤,对已初始化的项目进行简单修改,使其可以通过 Web Function 快速部署,此处项目改造通常分为以下两步:

- •新增 scf_bootstrap 启动文件。
- 修改监听地址与端口为 0.0.0.0:9000 。

具体步骤如下:

1. 在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容(用于启动服务并指定启动端口):

```
#!/var/lang/node12/bin/node
require("@nuxt/cli")
   .run(["start", "--port", "9000", "--hostname", "0.0.0.0"])
   .catch(error => {
    require("consola").fatal(error);
    require("exit")(2);
  });
```

▲ 注意:

- 此处仅为示例启动文件,具体请根据您的业务场景进行调整。
- 示例使用的是云函数标准 node 环境路径。本地调试时,需根据您创建云函数时选择的 node 版本,修改为您的本地路径。详情请参见 标准 语言环境绝对路径 。
- 2. 新建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可正常启动。示例如下:

nod 777 scf_bootstrap

- 3. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 4. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。



5. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

建筑式 模板の設置 使用示例機械於決進创建一个函数或成用 目定义创建。函数 使用helloworld示例目定义创建函数 正報代型・ ● 事件函数 ● Web函数 ● 国际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 日际包含字母、数字、下划线、连字符、以字母开头、以数字或字母结尾、2-60个字符 「「一一」 「「一一」 「一一」 正方式・ ● 代码部署 ● 課金部署 日示式 ● で送簿項 ● 本地上传文中文 ● 本地上传文中大 ● 通过の2-L传文内包 ■ 二十代 ■ 重算上代 重弦代码 ● Index 20 正確完成 重算上代 重算上代 重弦文件来、最大支持250M			
	沙建 方式	模板创建 使用示例模版快速创建一个函数或应用	自定义创建 使用helloword示例自定义创建函数
atti (1997) Atti (1997) At	基础配置		
融密名称・ ExpressDemo 日転物会学母、数字、下划线、连学符、以字母开头、以数字或字母线尾、2~60个字符 感・ ③ 广州 ・ 躍方式・ ④ 代码部署 ● 機像部署 日环境・ Nodejs12.16 ・ 部 部 部 部 部 部 部 部 部 部 部 部 部	函数类型 ★	◯ 事件函数 🔵 Web函数 ③	
日能勉会学母、数学、下划线、途学符,以数学或学母绘展、2-60个学符 感 ・ ・ ● 代码部署 ● 操象部署 15环境・ ● 代码部署 ● 操象部署 15环境・ ● 16项目急,清修文馆的项目监听演口为9000 ● 在线编辑 ● 本地上作文件夹 ● 通过cos上作2/P包 20次行時・ ● 在线编辑 ● 本地上作文件夹 ● 通过cos上作2/P包 20次行時・ ● 回復完成 ● 重新上作 请逐择文件夹、最大支持250M	的数名称 *	ExpressDemo	
 ◎ 广州 ▼ □ 代码部署 ● 接象部署 □ 行研調 ● 「接象部署 □ 行研調 ● 「接象部署 □ 打探機・ Nodejs12.16 ▼ ■ 数代码 ● 上传项目載, 清修改造的项目监听端口为9000 ▼ ■ 数方法・ ● 在接線編 ● 本地上传动内包 ● 本地上传文件夫 ● 通过cos上传动内包 ■ 面包x 20 圧縮振成 重新上传		只能包含字母、数字、下划线、连字符,以字	母开头,以数字或字母结尾,2~60个字符
■ 時方式・ ●代朝部署 ● 機像部署 目行环境・ Nodejs12.16 ▼ ■ 動代码 ● 上传项目前, 请修改您的项目监听集日为9000 * ■ 取方法・ ● 在线编辑 ● 本地上传zip包 ● 本地上传文件夹 ● 通过cos上传zip包 ■ 認知代码・ Index zip 圧縮完成 重新上传 请您择文件夹, 最大支持250M	8域 *	◎ 广州	
E存环境・ Nodejs12.16 ▼ B数代码 ① 上传项目前,请修次您的项目监听续□为9000 ▼ E交方法・ ○ 在线编辑 ○ 本地上传文件夹 ○ 通过cos上传zip包 E欧介法・ Index.2p 圧縮完成 重新上传 请您指文件夹,最大支持250M	『署方式 *	○代码部署 ○镜像部署	
函数代码 ① 上传项目前, 请修改您的项目监听换口为9000 ▼ 認文方法・ ○ 在线编辑 ○ 本地上传文件夹 ○ 通过cos上传zip包 函数代码・ index.zp 圧縮完成 重新上传 请选择文件夹,最大支持250M	云行环境★	Nodejs12.16 v	
	函数代码	① 上传项目前,请修改您的项目监听端口为900	10
	函数代码 是交方法。 图数代码。 函数类	 ① 上传项目前,请传次您的项目监听换口为900 ○ 在线编辑 ○ 本地上传之内包 ● 本地 ■ 市场经文件关,最大支持250M ■ 市场经文件关,最大支持250M ■ 2011 <	10 8上传文件夹 ① 通过cos上传zip包 新上传
	函数代码 量交方法。 函数代码。 函数类 函数之	 上传项目前,请侍次您的项目监听换口为900 在线编辑 本地上传2p包 ● 本地 市场水之印 压缩完成 请选择文件夹,最大支持250M 型:选择 "Web 函数"。 添:填写您自己的函数名税 	00 8上传文件夹 ○ 通过cos上传zip包 新上传 下。
地域 :填写您的函数部署地域,默认为广州。	函数代码 建交方法。 函数代码。 函数类 函数之 地域:	 上传项目前,请传改您的项目监听换口为900 在线编辑 本地上传之内包 ●本地 面包の之口 医缩亮成 请选择文件夹,最大支持250M 题:选择 "Web 函数"。 添称:填写您自己的函数名和 填写您的函数部署地域,累 	¹⁰ ^{8上传文件来} ○ 通过cos上传zip包 新上传 序。 犬认为广州 。

- 部署方式:选择"代码部署",上传您的本地项目。
- 提交方法:选择"本地上传文件夹"。
- **函数代码:** 选择函数代码在本地的具体文件夹。
- 6. 单击**完成**完成 Nuxt.js 项目的部署。

△ 注意:

访问 URL 时,可能由于前端路由导致访问失败,访问时需去掉 /release 路径。

开发管理

部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势。



快速部署 Django 框架

最近更新时间: 2024-11-15 15:19:32

操作场景

本文将为您指导如何通过 Web Function,将您的本地 Django 快速部署到云端。

说明:
 本文档主要介绍控制台部署方案,您也可以通过命令行完成部署,详情请参见 通过命令行完成框架部署。

前提条件

在使用腾讯云云函数服务之前,您需要 注册腾讯云账号 并完成 实名认证。

操作步骤

模板部署: 一键部署 Django 项目

- 1. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 2. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 3. 选择使用模板创建来新建函数,在搜索框里输入 Django 选择 Django 框架模板并单击下一步。如下图所示:

方式	模板创建 自定义创建 使用helloworld示例自定义创建函数	
素 d	jango 多个过滤标签用回车键分隔	Q 共1个
I	Django 框架模版 查看详情	
11	2月 國数	
ł	鄯述 基于 API 网关和 Web 函数,快速部署 Django 示例 项目	
1	示签 WebFunc Django Python3.6 web	
	作者 🔗 購訊云	
	^第 第 7 246次	

4. 在新建页面,您可以查看模板项目的具体配置信息并进行修改。

5. 单击完成即可创建函数。函数创建完成后,您可在函数管理页面查看 Web 函数的基本信息。



6. 您可以通过 API 网关生成的访问路径 URL,访问您部署的 Django 项目。单击左侧菜单栏中的触发管理,查看访问路径。如下图所示:

← FlaskDemo-	正常		
函数管理	触发管理		
触发管理	创建触发器		
监控信息			
日志查询	API网关触发	所属别名: 默认流量	
并发配额	API服务名	SCF_API_SERVICE 🗹	
部署日志	serviceId	service-	
	apild	api-	
	请求方法	ANY	
	发布环境	发布	
	鉴权方式	免鉴权	
	启用Base64编码	未启用	
	支持CORS	否	
	后端超时	15s	
	访问路径	https://service-	.gz.apigw.tencentcs.com/release/

7. 单击访问路径 URL,即可访问服务 Django 项目。如下图所示:



本地开发

1. 执行以下命令,确认您本地的环境已安装好 Django。

2. 在本地创建 Hello World 示例项目。

目录结构如下:





3. 在本地执行 python manage.py runserver 命令运行启动文件。示例如下:

```
$ python manage.py runserver
July 27, 2021 - 11:52:20
Django version 3.2.5, using settings 'helloworld.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

4. 打开浏览器访问 http://127.0.0.1:8000 ,即可在本地完成 Django 示例项目的访问。如下图所示:

\leftrightarrow \rightarrow C (i) 127.0.0.1:8000		
	django	View <u>release notes</u> for Django 3.2
	The install worked successfully! Congratulation	5!
	You are seeing this page because <u>DEBUG=True</u> is in your settings file and you have not configured any URLs.	
	Operation Operation <t< td=""><td>ango Community onnect, get help, or contribute</td></t<>	ango Community onnect, get help, or contribute

部署上云

接下来执行以下步骤,对本地已创建完成的项目进行简单修改,使其可以通过 Web Function 快速部署,对于 Django,具体修改步骤如下:

1. 安装依赖包

1. 由于 SCF 云上标准环境内未提供 Django 依赖库,此处您必须将依赖文件安装完成后,与项目代码一起打包上传。请先新建 requirements.txt 文件,文件内容如下:



版权所有:腾讯云计算(北京)有限责任公司



由于初始化的默认项目引用了 db.sqlite3 库,请同步安装该依赖,或将项目文件内 setting.py 里 DATABASES 字段部分配置注释。

3. 新增 scf_bootstrap 启动文件

在 Web 函数内,限制了监听端口必须为9000,因此需要修改监听地址端口,在项目根目录下新建 scf_bootstrap 启动文件,在该文件添加如下内容 (用于完成环境变量配置,指定服务启动命令等自定义操作,确保您的服务可以通过该文件正常启动):



4. 创建完成后,还需执行以下命令修改文件可执行权限,默认需要 777 或 755 权限才可以正常启动。示例如下:

chmod 777 scf_bootstrap

△ 注意:

- 在 SCF 环境内,只有 /tmp 文件可读写,建议输出文件时选择 /tmp ,其他目录会由于缺少权限而写入失败。
- 如需在日志中输出环境变量,需在启动命令前加 -u 参数,例如 python -u app.py 。
- 5. 本地配置完成后,执行以下命令启动服务(如下命令为在 scf_bootstrap 目录下执行时示例),确保您的服务在本地可以正常启动。

⚠ 注意: 本地测试时注意将 python 路径改为本地路径。

./scf_bootstrap

- 6. 登录 Serverless 控制台,单击左侧导航栏的函数服务。
- 7. 在主界面上方选择期望创建函数的地域,并单击新建,进入函数创建流程。
- 8. 选择自定义创建新建函数,根据页面提示配置相关选项。如下图所示:

创建方式		
31/22/11-24	模板创建	自定义创建
	使用示例模版快速创建一个函数或应用	使用helloworld示例自定义创建函数
甚础配置		
数类型 *	🔵 事件函数 🛛 🔵 Web函数 (
数名称 *	DjangoDemo	
	只能包含字母、数字、下划线、连字符,以字	母开头,以数字或字母结尾,2~60个字符
]域 *	▶ 「注注 ●	
『署方式 *	● 代码部署 🦳 镜像部署	
	Duth and C	

- 函数名称:填写您自己的函数名称。
- 地域:填写您的函数部署地域,例如成都。
- 部署方式:选择"代码部署",上传您的本地项目。
- 运行环境:选择 "Python3.6"。
- 9. 单击完成完成 Django 项目的部署。

开发管理



部署完成后,即可在 SCF 控制台快速访问并测试您的 Web 服务,并且体验云函数多项特色功能,例如层绑定、日志管理等,享受 Serverless 架构带来的 低成本、弹性扩缩容等优势,如下图所示:

