弹性微服务

持续集成

腾讯云

【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

持续集成
使用 Coding 部署插件创建持续集成

最近更新时间：2024-02-01 10:08:21

操作场景

弹性微服务应用可以使用 CODING 构建持续集成方案，更多关于 CODING 持续集成功能和使用的说明，请参见 CODING 持续集成 。
本文以 Java 应用为例，介绍如何使用 Coding 以 镜像 及 JAR/WAR 包 方式部署应用至弹性微服务。

准备工作

准备 Demo 应用

DEMO 代码由 Spring Initialize 自动生成，使用 Gradle 构建，依赖 Spring Web、Actuator 和 Prometheus 等。

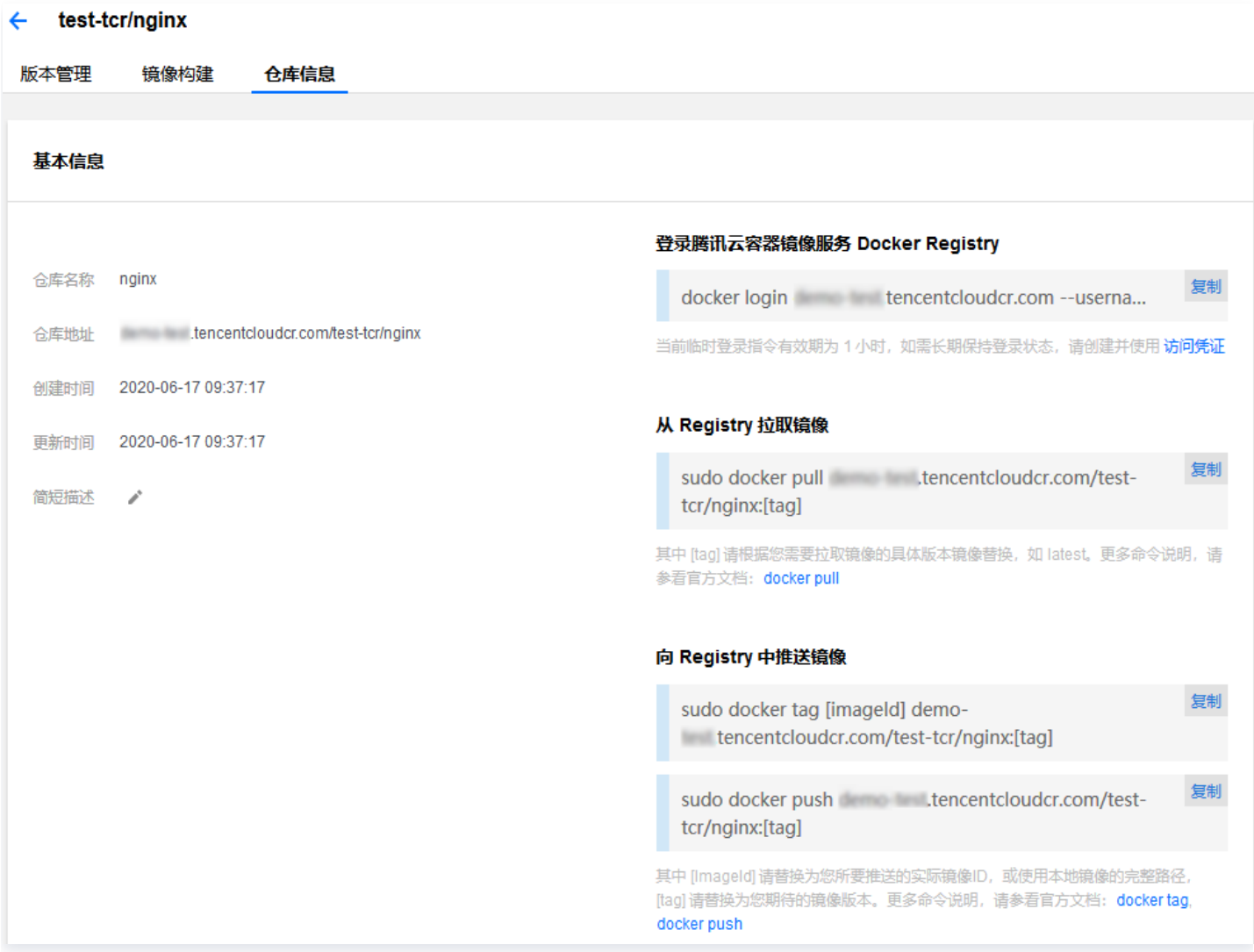
在控制台完成首次部署（可选）

首次部署时，有较多的参数需要设置。这部分信息如果全部固化到 CI/CD 流程里，就会显得特别复杂。
因此，如果您的业务涉及较多部署配置项，建议您在 弹性微服务控制台 中完成首次发布，或者通过Python部署脚本的方式配置持续集成。

> **说明：**
> Coding 构建计划里配置的发布方式应当与 TEM 保持一致，例如：如果弹性微服务应用为镜像部署方式，则 Coding 构建计划内也使用镜像部署。

确认 TCR 镜像信息

如果您需要通过镜像部署应用，需要将镜像推送至 TCR 镜像仓库并部署，因此需要获取相应 TCR 镜像仓库的 docker login 的 Registry。



Coding 平台操作

创建 Coding 项目和仓库

1. 创建全功能 DevOps 项目


STEP 1/2
选择项目模板
DEPLOY
全功能 DevOps 项目
开启一个全功能的 CODING 项目，包含代码托管、项目协同、持续集成、...
更换项目模板
STEP 2/2
填写项目基本信息
项目名称
TEM DEMO
可以使用中英文、数字、空格组合
项目标识
https://test-tem.coding.net/p
/
tem-demo
用于唯一标记项目，也可用于组织项目 URL
项目描述
DEMO for TEM CICD
完成创建
取消

2. 创建代码仓库。

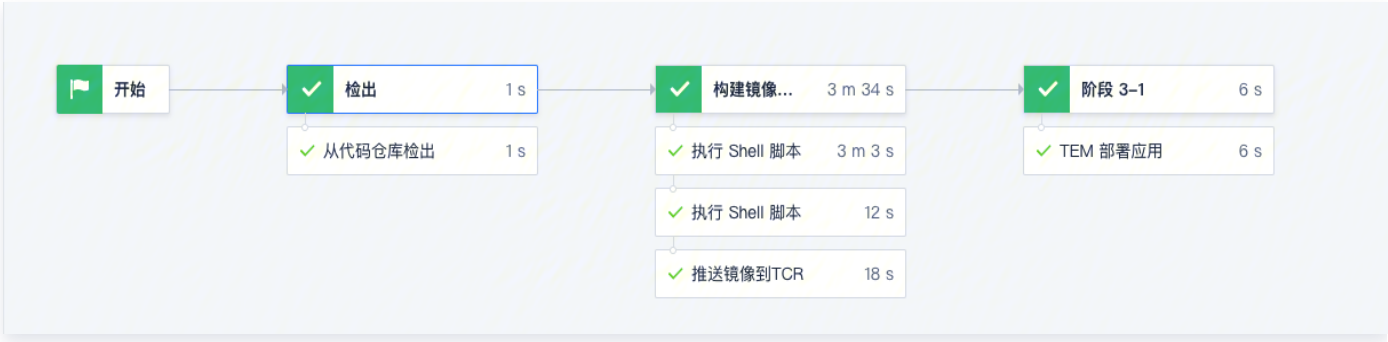
3. 添加 remote 仓库，并推送代码。

```
git init
git add .
git commit -m "init"
git remote add origin ***/tem-test/tem-demo.git
git push -u origin master
```

构建计划概览

默认 `git push origin master` 后，会触发构建计划。如您有特殊需求，可自行配置构建计划的触发规则。

- 采用镜像方式部署时：



- 采用程序包方式部署时：

配置构建计划

1. 模板选择自定义构建计划。

2. 在流程配置中，添加构建模块（执行 Shell 脚本）。由于 DEMO 中使用的 gradle 构建，可参考以下命令：

```
(cd /root/.gradle && rm init.gradle)
./gradlew clean build
```

3. **镜像部署时**在流程配置中，添加镜像构建模块（执行 Shell 脚本）

```
"docker build -t
${DOCKER_NAMESPACE}/${DOCKER_IMAGE_NAME}:${DOCKER_IMAGE_VERSION}
-f ${DOCKERFILE_PATH} ${DOCKER_BUILD_CONTEXT}"
```

您可自行在环境变量中定义相关变量：

- DOCKER_NAMESPACE： 命名空间。
- DOCKER_IMAGE_NAME：镜像仓库名称。
- DOCKER_IMAGE_VERSION：镜像版本号，e.g. ${GIT_COMMIT}。
- DOCKERFILE_PATH：dockerfile 文件位置，e.g. Dockerfile。
- DOCKER_BUILD_CONTEXT：Docker 构建目录，e.g. 使用当前目录。

4. **镜像部署时**在流程配置中，添加镜像推送模块（TCR 插件），腾讯云内网目前只支持 TCR（由于镜像拉取是大流量操作，对带宽要求比较高，不建议 NAT 接出公网）。您也可用使用 docker login && docker push 自行完成镜像推送。

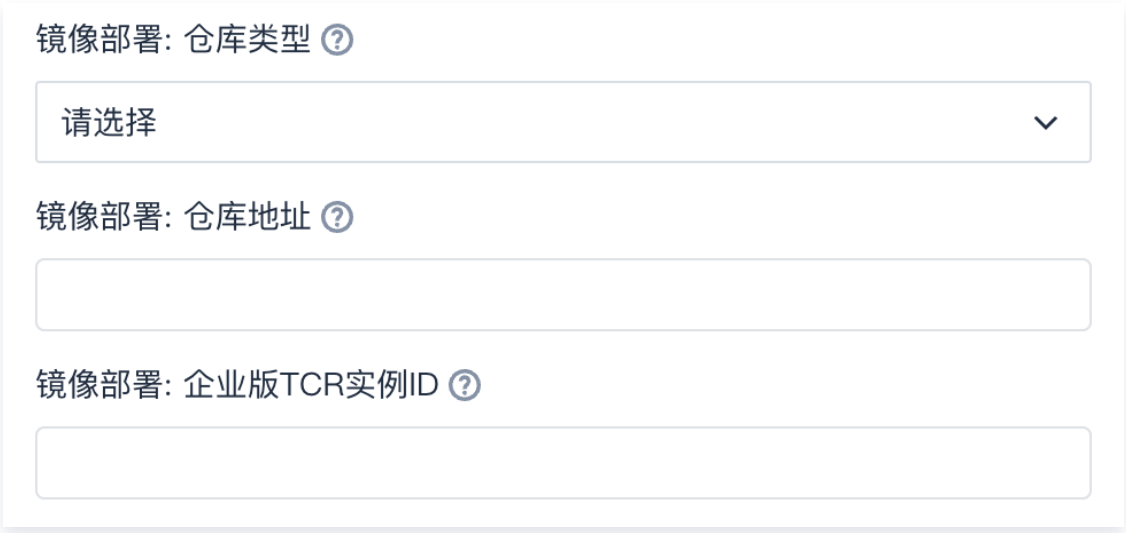
5. 在流程配置中，添加 TEM 部署模块。

TEM 部署插件配置

1. 配置通用信息与应用基本信息。

- 插件版本：选择最新。
- SecretId/SecretKey: 腾讯云的 API 密钥 。
- 地域：TEM 的服务地域。
- 应用名称：支持根据名称新建应用，或选择存量应用部署至新环境。
- 环境名称：根据环境名称指定一个存量环境，若无合适环境，请前往 创建环境 。

2. 配置镜像：配置关联的镜像 TAG，如 commitId 等。



3. 配置 JAR / WAR 包：除了配置部署版本以外，还需要额外配置 JAR/WAR 的文件路径。

4. 部署配置。

- 发布触发策略：
  - 自动触发：发布全自动执行。
  - 全手动触发：发布不同批次间，需要用户人工确认才会继续执行。
  - 小批量验证后自动触发：小批量验证完成后，经人工确认，后续批次自动触发。
- 发布批次：实例的发布批次数。
- 批次间等待时间：发布批次之间的间隔时间，以秒为单位，用来减少可能的服务抖动。
- 发布过程中保障的最小可用实例数：
  - -1 代表保证全量实例可用。
  - 0 代表不保障发布过程中一定有实例可用。
  - n 代表发布过程中保障有n个实例可用。

部署详情

部署操作是异步的，在部署操作执行完之后，会返回访问部署详情的链接。

手工确认的分批次发布场景（可选）

1. 增加 Coding 插件（人工确认）。

2. 继续下一批（执行 Shell 脚本）。

```
pip install tccli
tccli configure set secretId $SECRET_ID secretKey $SECRET_KEY region $REGION
output json
tccli tem ResumeApplication --version 2021-07-01 --ApplicationId $APPLICATION_ID
--EnvironmentId $ENVIRONMENT_ID
```

使用 Python 脚本部署应用

最近更新时间：2024-02-02 16:01:11

弹性微服务应用可以使用 Python 脚本来部署。

前提条件

在开始持续集成之前，需要完成下述的准备工作：

1. 保证机器上安装的 Python 版本不低于 3.0 版本，并已安装 PIP 等 Python 包管理工具。
2. 获取腾讯云的 访问密钥 SecretId 和 SecretKey。
3. 在弹性微服务 创建了环境 。
4. 安装脚本所需的依赖。

```
pip install tencentcloud-sdk-python cos-python-sdk-v5
```

部署准备

1. 下载 TEM 部署 Python 脚本， deploy.py 。
2. 根据脚本注释，修改您的部署脚本参数。
3. 运行脚本。

```
python3 deploy.py
```

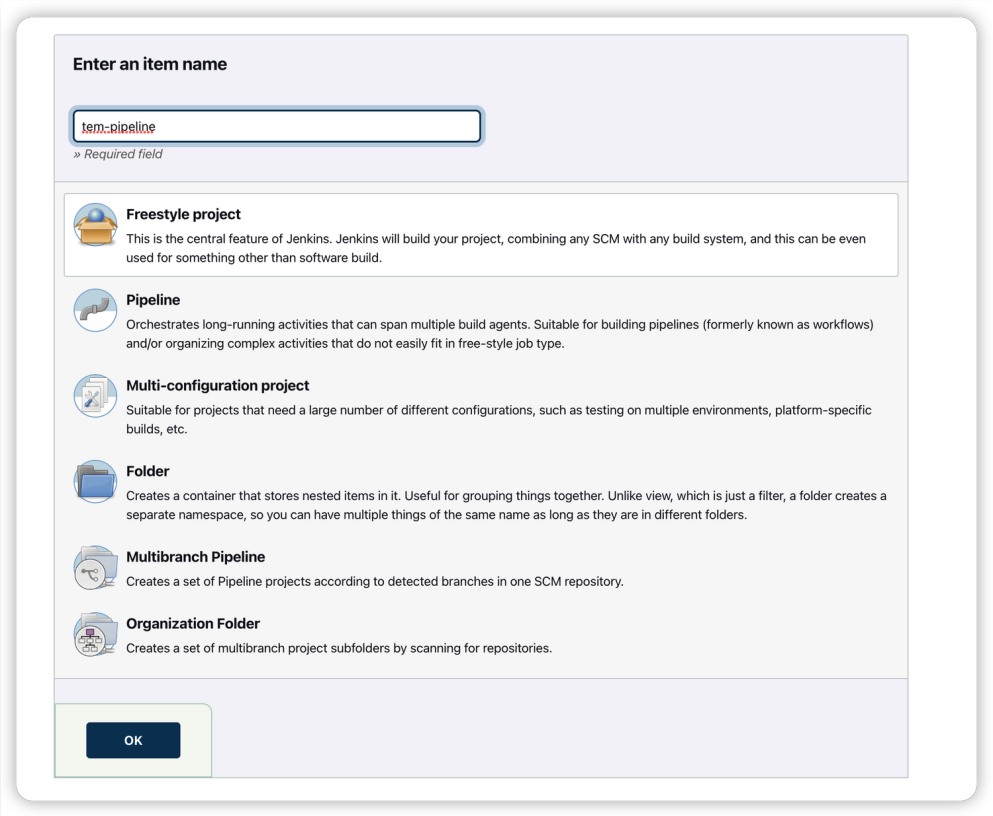
4. 登录 弹性微服务控制台 ，在左侧导航栏单击应用管理，进入应用列表页，选择目标应用，单击应用 ID，进入应用详情页，查看部署结果。

使用 Jenkins 创建持续集成

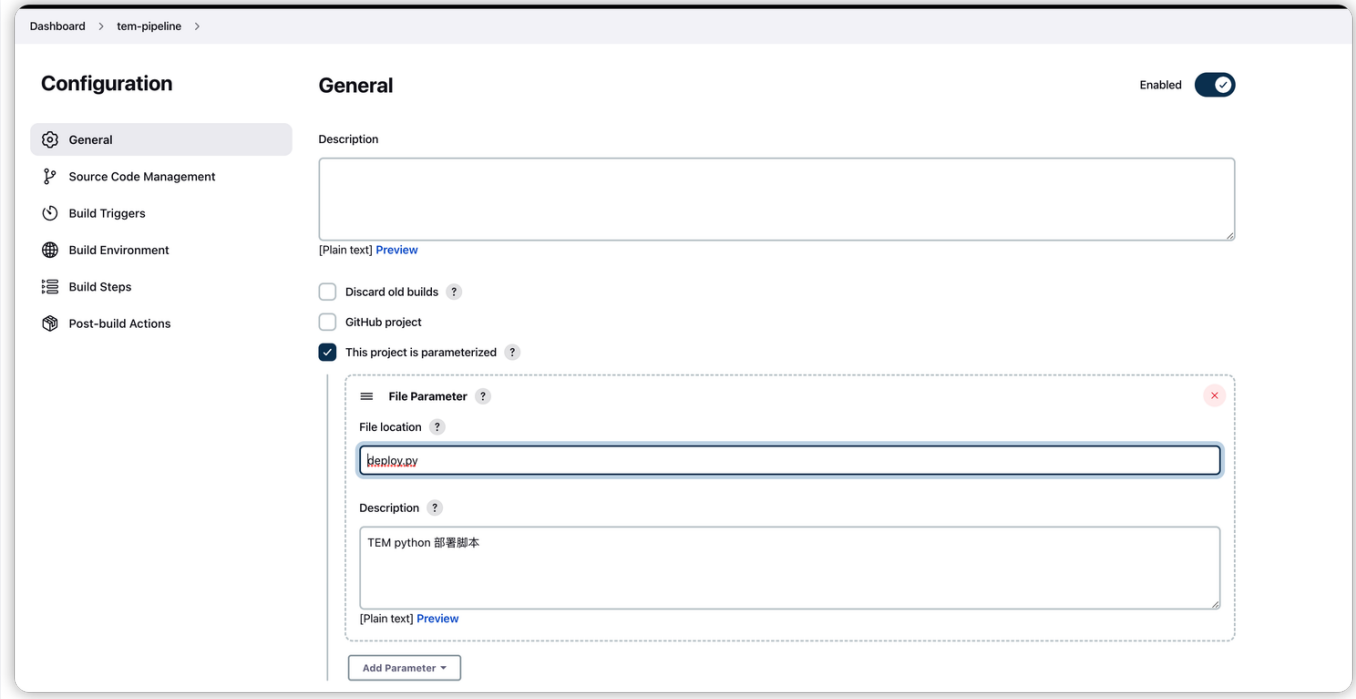
可以使用部署脚本在流水线构建持续集成方案，此处以 Jenkins 为例。

配置 Jenkins

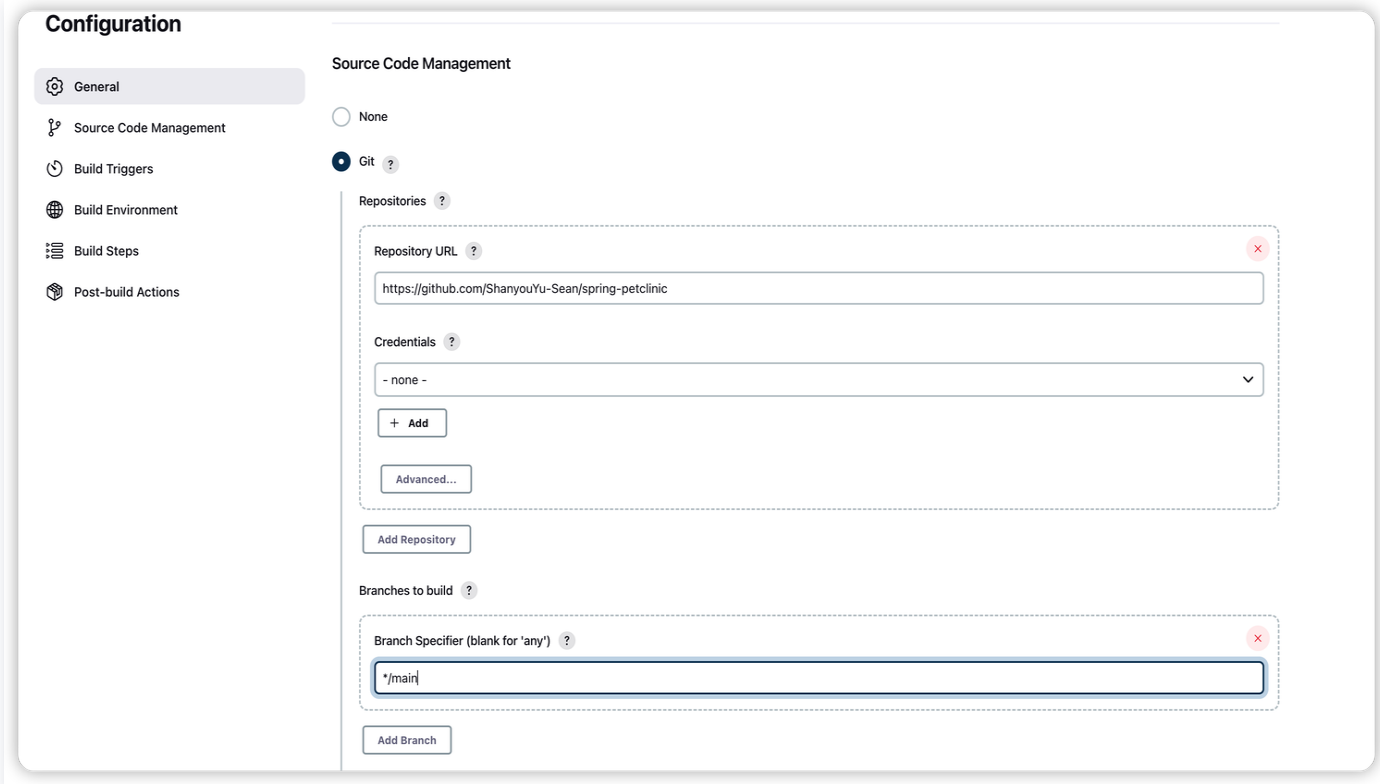
1. 在 Jenkins 首页左侧导航栏中单击新建，创建 Jenkins 任务，并选择构建一个自由风格的软件项目。



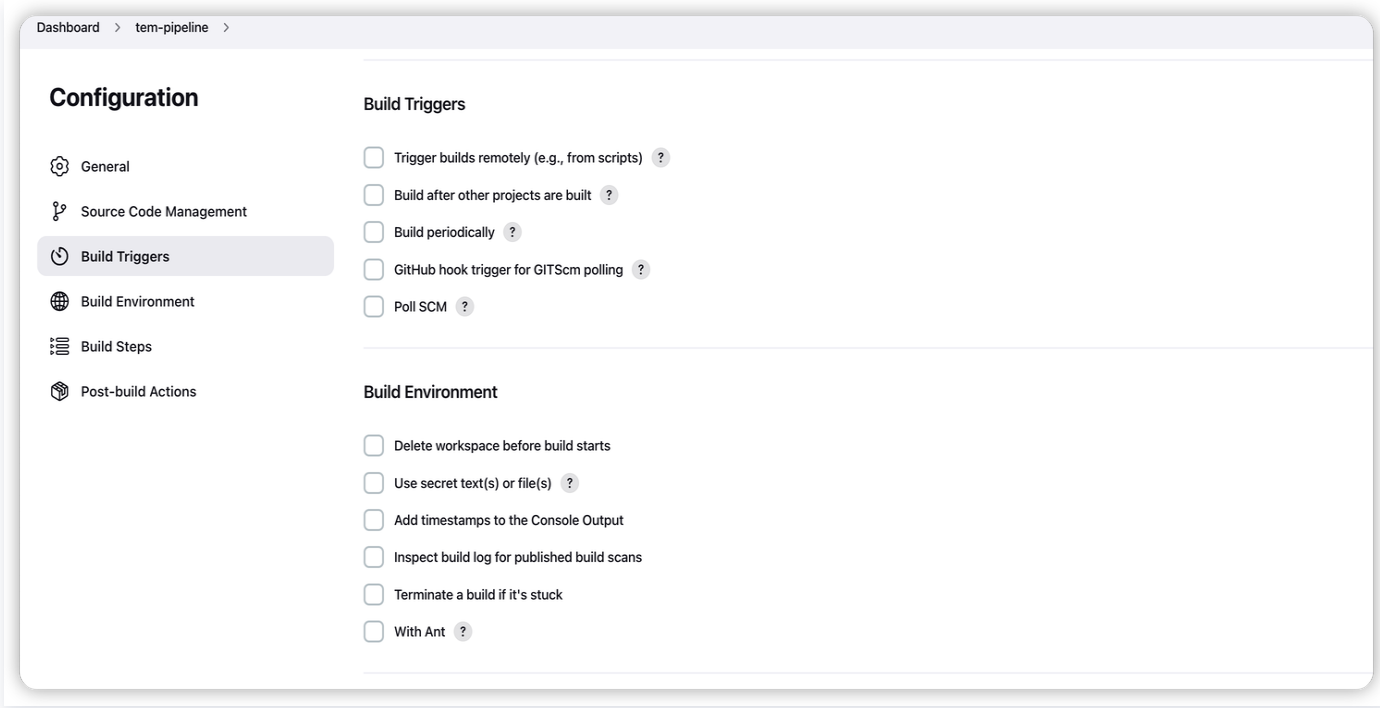
2. 在 General 中选择 `This project is parameterized -> File Parameter`，配置部署文件参数。



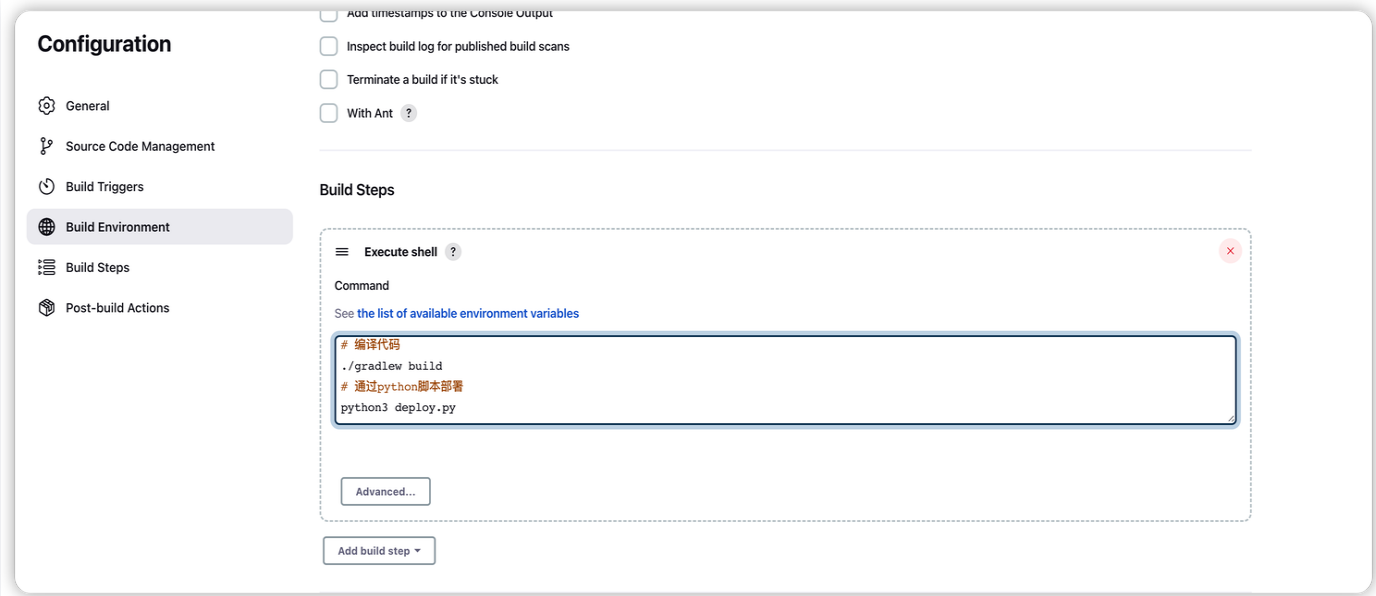
3. 配置您的项目源码。



4. 配置您的项目构建环境和触发器（本文档中暂省略，进行手动触发）。

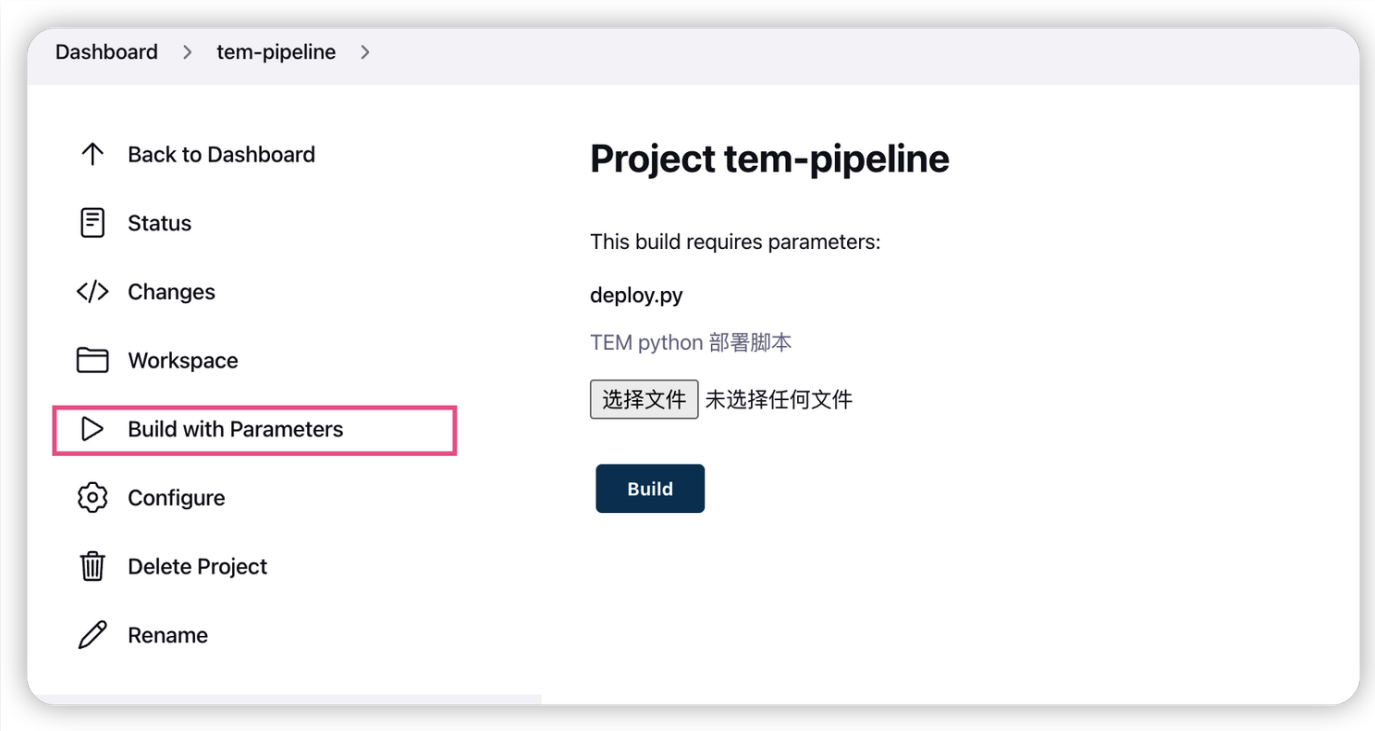


5. 添加构建步骤。

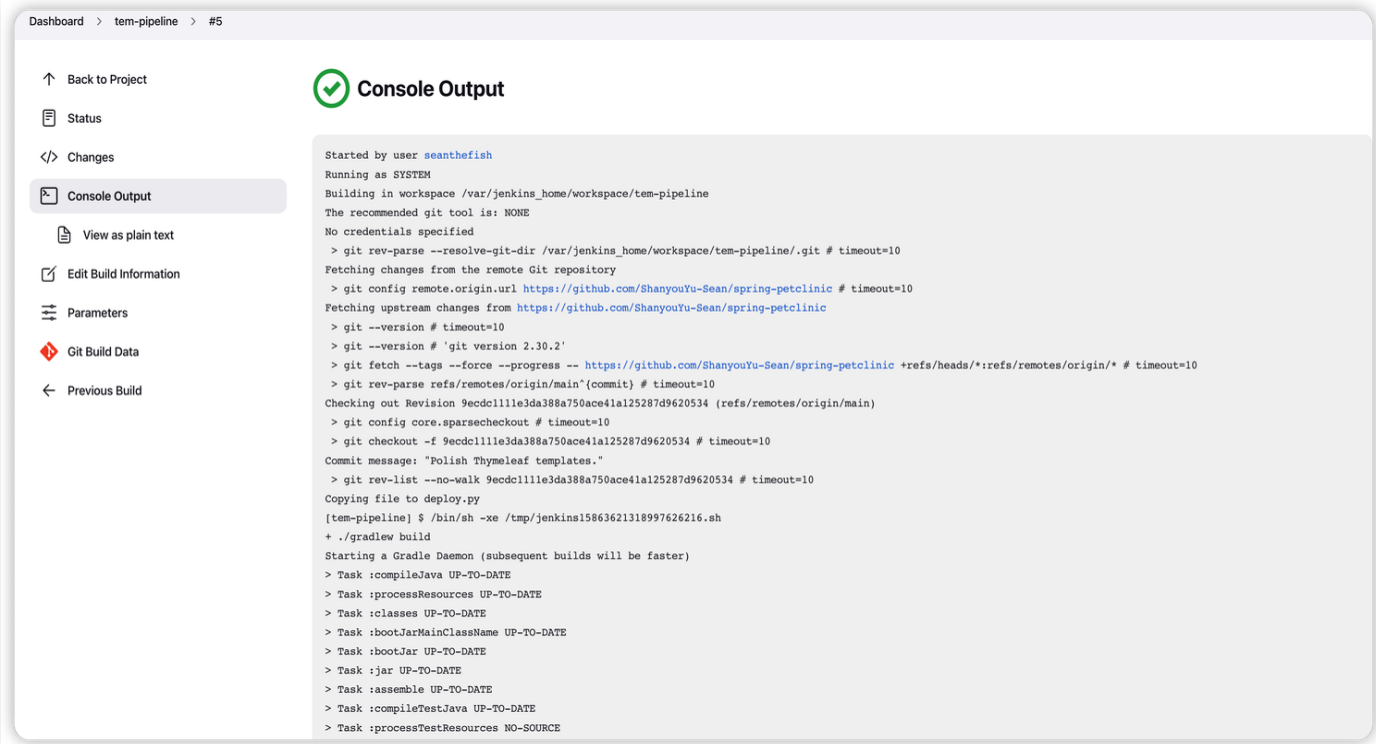


触发构建和部署

1. 选择 Build with Parameters，上传 TEM 部署脚本，触发构建和部署。



2. 在 Jenkins 中查看部署结果。



3. 在弹性微服务控制台查看部署结果。