

阿里云开发平台

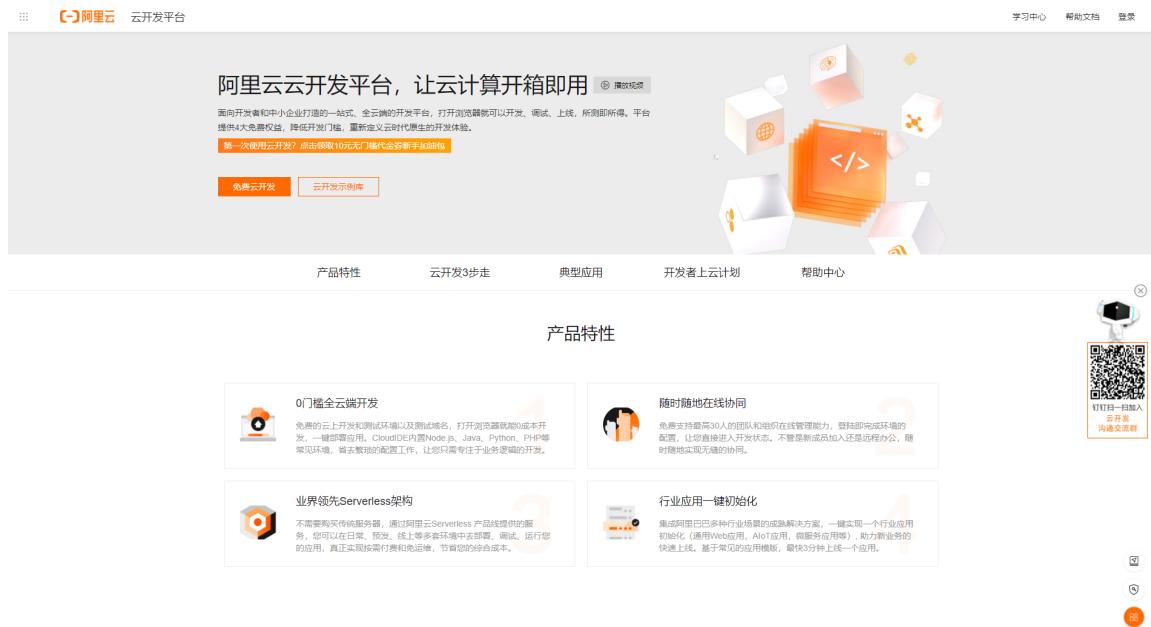
阿里云开发平台：workbench.aliyun.com

请仔细阅读每个步骤，尤其是标记为 **IMPORTANT** 的步骤。

若在“账号登录与初始化”的 **IMPORTANT** 步骤误操作，可能会导致**账号作废**，请联系助教重新为你创建一个 RAM 账号，根据实际情况扣除平时分。

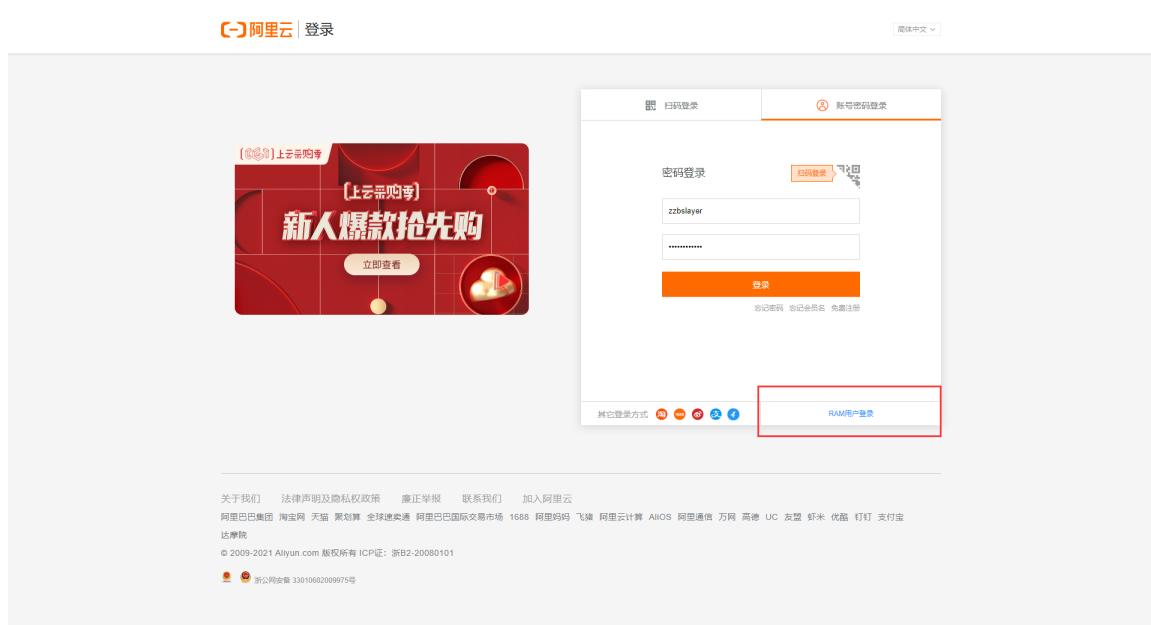
账号登录与初始化

- 访问[阿里云开发平台](#)，点击右上角的登录



The screenshot shows the main landing page of the Alibaba Cloud Dev Platform. At the top right, there is a 'Login' button. Below it, the page features a large banner with the text '阿里云云开发平台，让云计算开箱即用' (AliCloud Dev Platform, let cloud computing be open box ready). The banner includes a QR code and a link to 'Scan to log in'. The page also displays several product highlights: '0门槛全云端开发' (0-threshold full-cloud development), '随时随地在线协同' (隨時隨地在線協同), '业界领先Serverless架构' (Industry-leading Serverless architecture), and '行业应用一键初始化' (One-click initialization for industry applications). The bottom of the page includes a footer with links like 'About Us', 'Legal', 'Privacy Policy', and 'Feedback'.

- 选择 RAM 用户登录



The screenshot shows the login interface of the Alibaba Cloud Dev Platform. It features two main login options: 'Scan to log in' (扫码登录) and 'Account password login' (账号密码登录). The 'Account password login' section contains fields for 'Username' (zbslayer) and 'Password'. Below these fields is a 'Log in' (登录) button. A red box highlights the 'RAM User Login' (RAM用户登录) link located at the bottom right of the password login section. The footer of the page includes standard links such as 'About Us', 'Legal', 'Privacy Policy', and 'Feedback'.

- 使用助教发在你们邮箱内的 RAM 账户与密码来登录

阿里云 RAM 用户登录

主账号登录

RAM 用户登录

* 用户名
tttest@1414697749472774.onaliyun.com

RAM 用户名格式：<用户名><默认域名> 或 <用户名><企业别名>，例如：
username@company-alias.onaliyun.com 或 username@company-alias.

下一步

下载阿里云 App

RAM 用户登录阿里云 App，随时随地移动掌控

阿里巴巴集团 1688 全球速卖通 淘宝网 天猫 聚划算 一淘 阿里妈妈 阿里云计算 YunOS 万网 支付宝 闲逛 © 2009-2021 Aliyun.com 版权所有 增值电信业务经营许可证：浙 B2-20080101

- **IMPORTANT** 随便输入团队名称，然后创建

- 不能关闭或跳过该步骤
- 创建的团队并不会用到，但是不创建就会有问题

[〈 返回](#) [X](#)

输入企业、组织或团队的名字

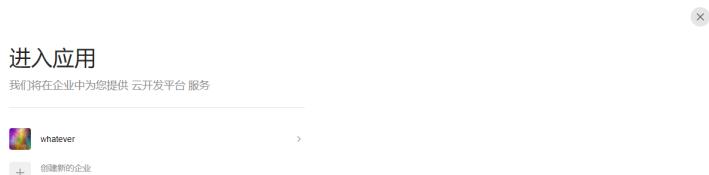
输入用户名的团队

使用规模 1 ~ 10 人

完成创建

我的企业已经完成钉钉绑定，绑定个人 [钉钉账号](#) 进入该企业

- 选择刚刚创建的团队进入



- 左上角切换团队至“zianghuang的团队”

The screenshot shows the top navigation bar of the Alibaba Cloud DevCloud platform. On the left is the Alibaba Cloud logo and '云开发平台'. In the center is a dropdown menu with the text 'whatever' and 'zianghuang的...'. A red box highlights the 'zianghuang的...' option. To the right of the dropdown are several links: '快速开始', '产品线管理', '应用管理', '开发模板', '成员管理', '学习中心', and '帮助文档'. At the top right is a user profile icon.

- 初始化成功

This screenshot shows the main dashboard of the Alibaba Cloud DevCloud after initialization. It features a large, stylized '500' icon at the top. Below it are two main sections: '创建新应用' (Create New Application) and '应用搬站' (Application Relocation). The '我的应用' (My Applications) section indicates that no applications have been created yet. A small orange button labeled '去创建>' is visible. The top navigation bar and right-side sidebar are identical to the previous screenshot.

创建 DEMO 后端应用

- 点击创建应用

This screenshot shows the 'Create Application' step in the DevCloud interface. It includes sections for 'Create New Application' and 'Application Relocation'. The 'My Applications' section shows a message: '您还没有创建过应用, 去创建>'. The top navigation bar and right-side sidebar are consistent with the previous screenshots.

- 以大作业 DEMO 的后端为例，选择 “JAVA” > “通过模板创建” > “团队模板” > “BookstoreBackend”，点击“下一步”

阿里云 云开发平台 zhanghuang的...

云开发平台 / 新建应用

新建应用

Step 1 指权管理 Step 2 选择创建方式 Step 3 填写基础信息 Step 4 云服务管理

请选择开发语言: Java NodeJS Python PHP

请选择创建方式: 直接创建 通过模板创建

请选择模板来源: 官方模版 团队模版

BookstoreBackend Bookstore Backend	se122_14_mongoandmysqlsample se122_14_mongoandmysqlsample	se122_12_springiocannotationssample se122_12_springiocannotationssample	se122_10_sshsample se122_13_sshsample
se122_12_springiocsample se122_12_springiocsample	se122_12_springarchsample se122_12_springarchsample	se122_12_hellosecuritysample se122_12_hellosecuritysample	se122_10_strutsample se122_11_strutsample
se122_10_reactdb se122_10_reactdb	se122_11_springmvcmxml se122_11_springmvcmxml	se122_11_springmvccannotation se122_11_springmvccannotation	se122_10_jsonsample se122_10_jsonsample
se122_10_ajaxsample se122_10_ajaxsample	se122_7_fileuploadsample se122_7_fileuploadsample	se122_9_ormadvancesample se122_9_ormadvancesample	se122_8_springgpaexample se122_8_springgpaexample

1 2 >

上一步 下一步

- 填写“应用名称”与“应用介绍”，“计算服务”选择“SAE”，示例项目统一放在“tutorial”“产品线”中，点击“下一步”

阿里云 云开发平台 zhanghuang的...

云开发平台 / 新建应用

新建应用

Step 1 指权管理 Step 2 选择创建方式 Step 3 填写基础信息 Step 4 云服务管理

应用模版: 书城后端 - bookstoreBackend (容器型)

应用名称: demo_bookstore_backend

应用介绍: demo_bookstore_backend

应用图标: 上传应用图标, 尺寸规格 300px

计算服务: SAE

所属产品线: tutorial

Region: 华东2 (上海)

上一步 下一步

- 点击“完成”

- 由于你们使用 RAM 子账号登录阿里云开发平台，所有费用将在阿里云主账号下结算

阿里云 云开发平台 zhanghuang的...

云服务管理 指权管理 选择创建方式 填写基础信息 云服务管理

依赖的云服务

计算服务: Serverless 应用引擎 SAE

云服务名称	服务价格	服务状态
API 网关 API Gateway	0.06元/万次, 详情	已开通
负载均衡 Server Load Balancer	0.6元/GB, 详情	已开通
Serverless 应用引擎 Serverless App Engine	3.11112元/天, 详情	已开通
专有网络 Virtual Private Cloud	免费, 详情	已开通
容器镜像服务/Alcloud Container Registry	默认免费, 详情	已开通

应用架构图

上一步 成功

- 成功完成后将会跳转至该页面

阿里云 云开发平台 zhanghuang0... 在上线部署

开发部署

环境	计算服务	状态	操作
日志环境	SAE	未部署	部署配置 域名配置 应用配置
预发环境	SAE	未部署	部署配置 域名配置 应用配置
线上环境	SAE	未部署	部署配置 域名配置 应用配置

运行 DEMO 后端代码

- 点击右上角“在线开发部署”

阿里云 云开发平台 zhanghuang0... 在上线部署

开发部署

环境	计算服务	状态	操作
日志环境	SAE	未部署	部署配置 域名配置 应用配置
预发环境	SAE	未部署	部署配置 域名配置 应用配置
线上环境	SAE	未部署	部署配置 域名配置 应用配置

- 跳转至在线编辑器页面
 - 建议阅读 README

- 编辑器下侧选择“终端”，使用命令 `mvn spring-boot:run` 运行项目

```

        mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] --- Maven Build Lifecycle Plugin: 3.6.3 (@{project.build.plugins[1].name}) @ DEMO BOOKSTORE BACKEND ---
[INFO]   <build>
[INFO]     <plugins>
[INFO]       <plugin>
[INFO]         <groupId>org.apache.maven.plugins</groupId>
[INFO]         <artifactId>maven-compiler-plugin</artifactId>
[INFO]         <version>3.8.1</version>
[INFO]         <configuration>
[INFO]           <target>1.8</target>
[INFO]           <encoding>UTF-8</encoding>
[INFO]         </configuration>
[INFO]       </plugin>
[INFO]     </plugins>
[INFO]   </build>
[INFO]
[INFO] --- Cloud-Native 配置是OK的。
[INFO] 1. Cloud-Native配置是在Cloud-Native右侧Terminal的命令行中，输入启动命令：mvn spring-boot:run，验证是否能在IDE容器中启动成功。启动成功后在IDE左下角有一个“快照”小窗，可以把云端中的服务在本地浏览器中访问到，即说明Cloud-Native配置是OK的。
[INFO] 2. 读取代码上端口：10080
[INFO] 3. 然后部署到云端：在IDE左边工具栏中打开云开发插件，然后点击部署按钮，开始部署到云端。如果部署成功，会在输出日志中，打印一个临时域名，可以直接访问。
[INFO] 到这一步，在本地用浏览器访问该临时域名。
[INFO]
[INFO] #### 5. Dockerfile 和 prepare.sh 这两个文件不要去修改
[INFO]
[INFO]
[INFO]
[INFO]

```

- 运行成功则如下图所示，可以看到项目默认运行于 8080 端口

```

        mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] --- Maven Build Lifecycle Plugin: 3.6.3 (@{project.build.plugins[1].name}) @ DEMO BOOKSTORE BACKEND ---
[INFO]   <build>
[INFO]     <plugins>
[INFO]       <plugin>
[INFO]         <groupId>org.apache.maven.plugins</groupId>
[INFO]         <artifactId>maven-compiler-plugin</artifactId>
[INFO]         <version>3.8.1</version>
[INFO]         <configuration>
[INFO]           <source>1.8</source>
[INFO]           <target>1.8</target>
[INFO]           <encoding>UTF-8</encoding>
[INFO]         </configuration>
[INFO]       </plugin>
[INFO]     </plugins>
[INFO]   </build>
[INFO]
[INFO] --- Cloud-Native 配置是OK的。
[INFO] 1. 读取代码上端口：10080
[INFO] 2. 读取代码上端口：10080
[INFO] 3. 然后部署到云端：在IDE左边工具栏中打开云开发插件，然后点击部署按钮，开始部署到云端。如果部署成功，会在输出日志中，打印一个临时域名，可以直接访问。
[INFO] 到这一步，在本地用浏览器访问该临时域名。
[INFO]
[INFO] #### 4. 到这里已经迁移完毕。需要验证一下是否迁移成功。
[INFO] 1. Cloud-Native右侧Terminal的命令行中，输入启动命令：mvn spring-boot:run，验证是否能在IDE容器中启动成功。启动成功后在IDE左下角有一个“快照”小窗，可以把云端中的服务在本地浏览器中访问到，即说明Cloud-Native配置是OK的。
[INFO] 2. 读取代码上端口：10080
[INFO] 3. 然后部署到云端：在IDE左边工具栏中打开云开发插件，然后点击部署按钮，开始部署到云端。如果部署成功，会在输出日志中，打印一个临时域名，可以直接访问。
[INFO] 到这一步，在本地用浏览器访问该临时域名。
[INFO]
[INFO] #### 5. Dockerfile 和 prepare.sh 这两个文件不要去修改
[INFO]
[INFO]
[INFO]
[INFO]

```

终端输出：

```

11:51:05.478 [main] INFO org.apache.catalina.core.StandardEngine Starting Servlet Engine: [Apache Tomcat/9.0.33]
11:51:05.496 [main] INFO org.apache.catalina.core.ContainerBase [localhost:10080], Initializing Spring embedded WebApplicationContext
11:51:05.507 [main] INFO org.springframework.web.context.ContextLoader [localhost:10080], Root WebApplicationContext: initialization completed in 2676 ms
11:51:05.609 [main] INFO org.hibernate.Version [HIBERNATE-VERSION-5.4.12.Final]
11:51:05.619 [main] INFO org.hibernate.dialect.Dialect [HIBERNATE-DIALECT-MySQL51]
11:51:05.639 [main] INFO org.hibernate.hql.ast.QueryTranslatorImpl [AST-TRANSLATOR]
11:51:05.646 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Starting...]
11:51:05.649 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Start completed.]
11:51:05.653 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Using dialect:MyQSLialect]
11:51:05.657 [main] INFO org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform [JTA Platform implementation: org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
11:51:05.576 [main] INFO org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean [Initialized JPA EntityManagerFactory for persistence unit 'default']
11:51:05.586 [main] INFO org.springframework.jdbc.datasource.DriverManagerDataSource [Explicitly configure spring.jpa.open-in-view to disable this warning]
11:51:05.619 [main] INFO org.springframework.scheduling.concurrent.ThreadsExecutor [ThreadsExecutor]
11:51:05.623 [main] INFO org.springframework.boot.web.embedded.tomcat.TomcatWebServer [Port started on port(s): 8080 [http] with context path '']
11:51:05.644 [main] INFO com.reins.bookstore.BookstoreApplication [Started BookstoreApplication in 11.471 seconds (JVM running for 11.799)]

```

- 点击左下角“预览”

- IMPORTANT** 预览有时间限制，超过 30 分钟后失效

```

        mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] --- Maven Build Lifecycle Plugin: 3.6.3 (@{project.build.plugins[1].name}) @ DEMO BOOKSTORE BACKEND ---
[INFO]   <build>
[INFO]     <plugins>
[INFO]       <plugin>
[INFO]         <groupId>org.apache.maven.plugins</groupId>
[INFO]         <artifactId>maven-compiler-plugin</artifactId>
[INFO]         <version>3.8.1</version>
[INFO]         <configuration>
[INFO]           <source>1.8</source>
[INFO]           <target>1.8</target>
[INFO]           <encoding>UTF-8</encoding>
[INFO]         </configuration>
[INFO]       </plugin>
[INFO]     </plugins>
[INFO]   </build>
[INFO]
[INFO] --- Cloud-Native 配置是OK的。
[INFO] 1. Cloud-Native右侧Terminal的命令行中，输入启动命令：mvn spring-boot:run，验证是否能在IDE容器中启动成功。启动成功后在IDE左下角有一个“快照”小窗，可以把云端中的服务在本地浏览器中访问到，即说明Cloud-Native配置是OK的。
[INFO] 2. 读取代码上端口：10080
[INFO] 3. 然后部署到云端：在IDE左边工具栏中打开云开发插件，然后点击部署按钮，开始部署到云端。如果部署成功，会在输出日志中，打印一个临时域名，可以直接访问。
[INFO] 到这一步，在本地用浏览器访问该临时域名。
[INFO]
[INFO] #### 5. Dockerfile 和 prepare.sh 这两个文件不要去修改
[INFO]
[INFO]
[INFO]
[INFO]

```

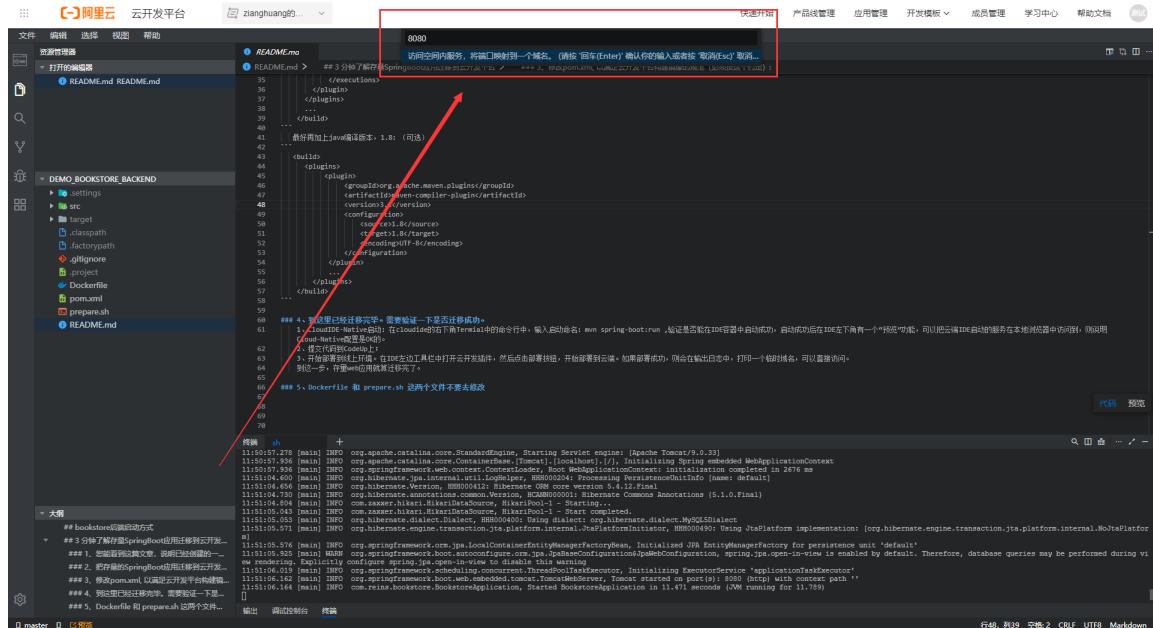
终端输出：

```

11:51:05.478 [main] INFO org.apache.catalina.core.StandardEngine Starting Servlet Engine: [Apache Tomcat/9.0.33]
11:51:05.496 [main] INFO org.apache.catalina.core.ContainerBase [localhost:10080], Initializing Spring embedded WebApplicationContext
11:51:05.507 [main] INFO org.springframework.web.context.ContextLoader [localhost:10080], Root WebApplicationContext: initialization completed in 2676 ms
11:51:05.609 [main] INFO org.hibernate.Version [HIBERNATE-VERSION-5.4.12.Final]
11:51:05.619 [main] INFO org.hibernate.dialect.Dialect [HIBERNATE-DIALECT-MySQL51]
11:51:05.639 [main] INFO org.hibernate.hql.ast.QueryTranslatorImpl [AST-TRANSLATOR]
11:51:05.646 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Starting...]
11:51:05.649 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Start completed.]
11:51:05.653 [main] INFO com.slexer.hibar.HibarDataSource [HibarPool-1 - Using dialect:MyQSLialect]
11:51:05.657 [main] INFO org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform [JTA Platform implementation: org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
11:51:05.576 [main] INFO org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean [Initialized JPA EntityManagerFactory for persistence unit 'default']
11:51:05.586 [main] INFO org.springframework.jdbc.datasource.DriverManagerDataSource [Explicitly configure spring.jpa.open-in-view to disable this warning]
11:51:05.619 [main] INFO org.springframework.scheduling.concurrent.ThreadsExecutor [ThreadsExecutor]
11:51:05.623 [main] INFO org.springframework.boot.web.embedded.tomcat.TomcatWebServer [Port started on port(s): 8080 [http] with context path '']
11:51:05.644 [main] INFO com.reins.bookstore.BookstoreApplication [Started BookstoreApplication in 11.471 seconds (JVM running for 11.799)]

```

• 编辑器最上方输入项目端口 8080



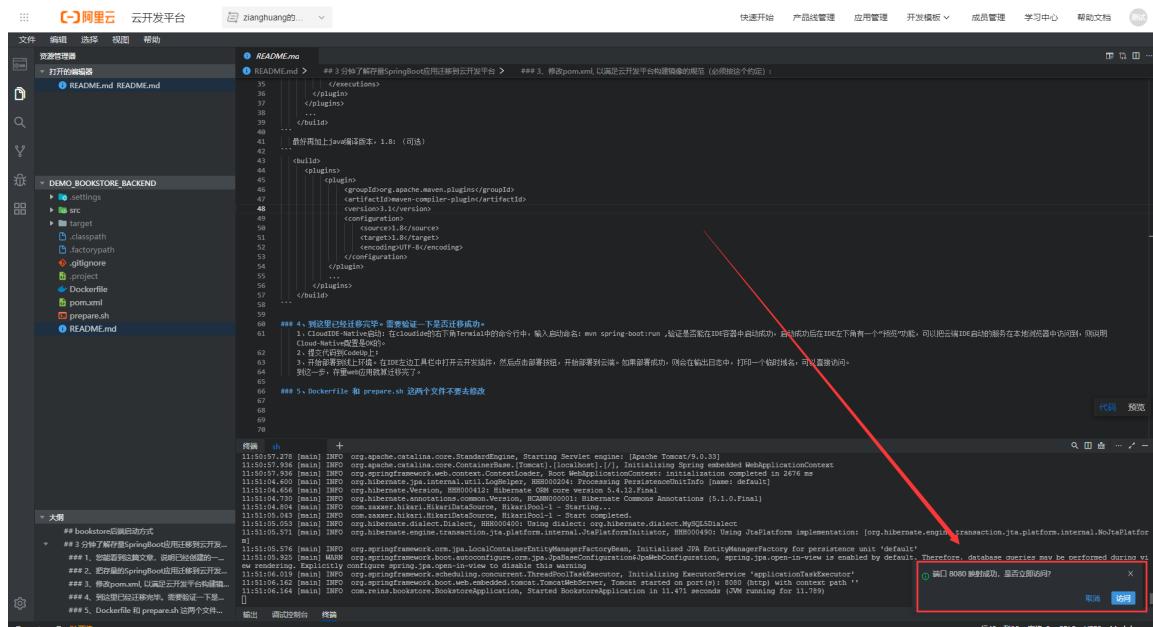
The screenshot shows an IDE interface with a red arrow pointing from the top status bar to the code editor. The status bar displays '快速开始' (Quick Start) and '端口 8080' (Port 8080). The code editor contains Java code for a Spring Boot application, specifically a Dockerfile snippet:

```

...
## 3 分钟了解如何将端口映射到一个域名。请按回车(Enter)确认你的输入或者按 取消(Esc) 取消。
...
8080
...

```

• 点击右下角“访问”



The screenshot shows an IDE interface with a red arrow pointing from the bottom right corner to the '访问' (Access) button. The code editor contains Java code for a Spring Boot application, specifically a Dockerfile snippet:

```

...
## 3 分钟了解如何将端口映射到一个域名。请按回车(Enter)确认你的输入或者按 取消(Esc) 取消。
...
8080
...

```

• 弹出的窗口中，可以使用地址栏中的域名访问后端，访问 /test 获得到 hello 则证明运行成功



创建 DEMO 前端应用

- 在应用列表页面，点击右上角“创建应用”

- 以大作业 DEMO 的前端为例，选择 NodeJS > “通过模板创建” > “团队模板” > “BookstoreFrontend”，点击“下一步”

- 填写“应用名称”与“应用介绍”，“计算服务”选择“FC”，示例项目统一放在“tutorial”“产品线”中，点击“下一步”

- 点击“完成”

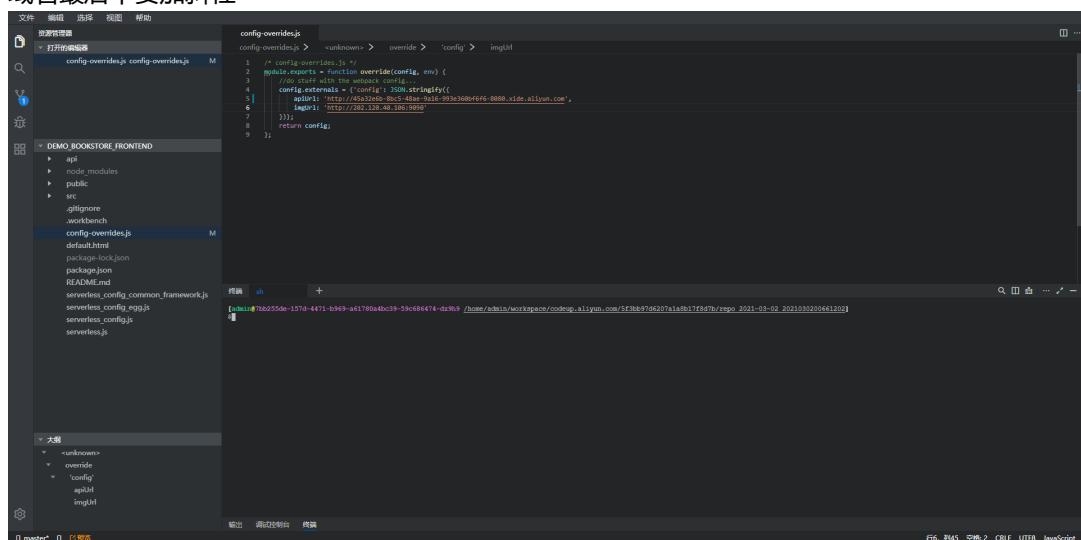
- 由于你们使用 RAM 子账号登录阿里云开发平台，所有费用将在阿里云主账号下结算

The screenshot shows the 'Create Application' wizard in progress. Step 3: Select Service Configuration. It lists several services with their prices and status:

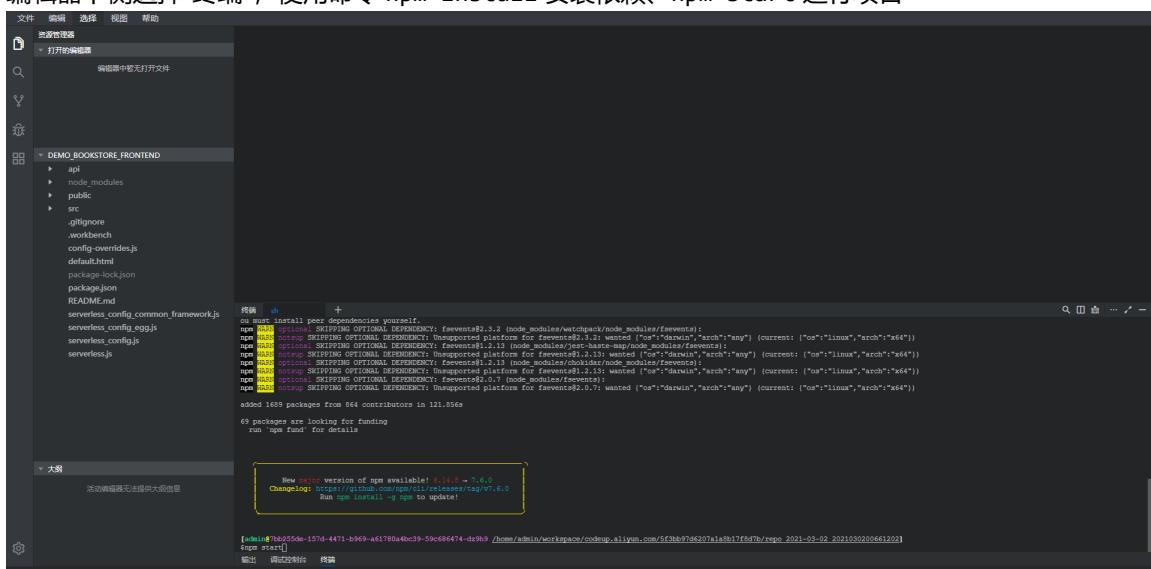
服务名称	参考价格	服务状态
API 网关 API Gateway	0.00元/万次 详情	已开通
函数计算 Function Compute	0.0133元/万次 详情	已开通
对象存储 Object Storage Service	0.125元/GB 详情	已开通
日志服务 Log Service	0.0115元/QB/天 详情	立即开通

运行 DEMO 前端代码

- 类似地，点击“在线开发部署”后跳转至编辑器页面
- IMPORTANT** 修改 config-overrides.js 中的 apiUrl 为你的刚刚启动的后端的域名
 - 必须以 https:// 为前缀
 - 域名最后不要加斜杠

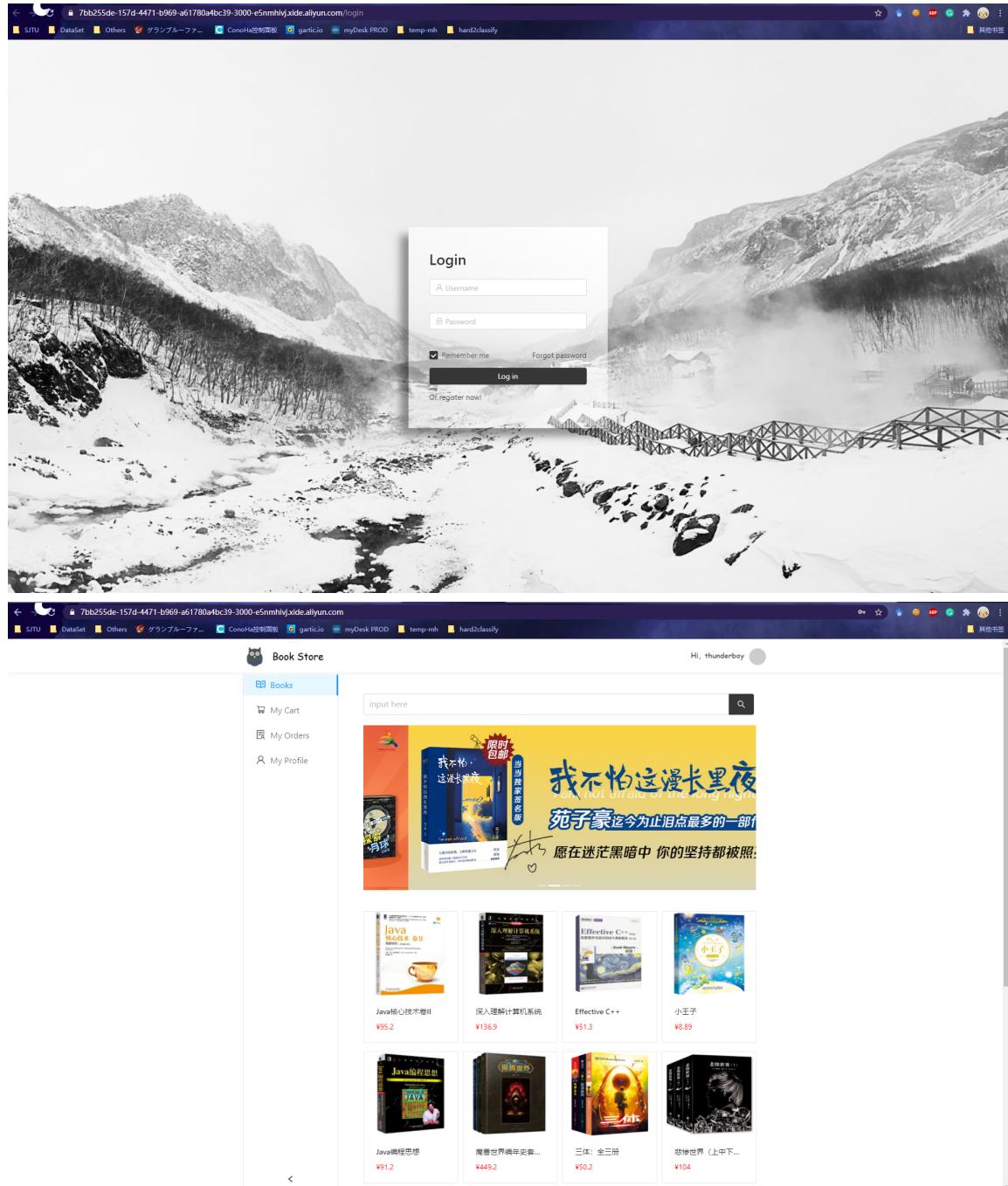


- 编辑器下侧选择“终端”，使用命令 npm install 安装依赖、npm start 运行项目



- 类似地，点击左下角“预览”，最上方输入端口号 3000，右下角点击“访问”

- 看到如下页面即运行成功



其他

- 阿里云开发平台的登录时间也有限制，如果开发过程中出现问题，可以先排查是不是登录过期
 - 访问阿里云开发平台首页，重新登录即可