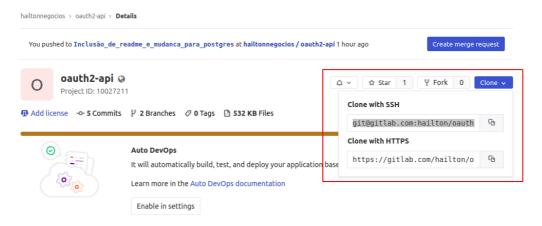
Tutorial para executar o projeto demo

1) Clonar o projeto na URL abaixo:

https://gitlab.com/hailton/oauth2-api

2) Após clonar o projeto, incluir o projeto no Eclipse ou Spring Tool Suite



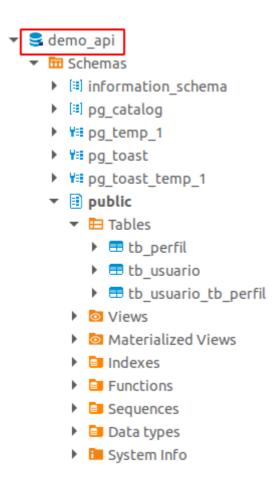


Estrutura de pastas do projeto:

- ▼ 2 > src/main/java
 - ▼ # > br.com.oauth2.exemplo
 - ▶ ⊕ config
 - ▶ 冊 > controller
 - ▶ 🖶 domain
 - ▶ # > dto
 - Repository
 - ▶ # service
 - ▶ 🛂 AppBoot.java

3) Após importar o projeto na IDE, criar um banco no postgres com o nome "demo-api"

Obs : Não precisa criar as tabelas, pois o Hibernate criará para você ao startar a aplicação



4) Após a importação do projeto e a criação do banco de dados, você já pode rodar a classe do Springboot para startar a aplicação :

```
₽ Package Explorer 🛭 🔓 Project Explorer

    AppBoot.java 
    S

                                                                                                                package br.com.oauth2.exemplo;
▼ 🛵 > demo-api [oauth2-api Inclusão_de_readme_e_mudanca_para_postgres]
   ▼ 👸 > oauth2-api [boot] [oauth2-api Inclusão_de_readme_e_mudanca_para_postgres]
                                                                                                                    \stackrel{-}{\texttt{3}} \oplus \textbf{import} \ \text{org.springframework.boot.SpringApplication;} \square
      ▼ # > src/main/java
                                                                                                                  8
9 @SpringBootApplication
10 @ComponentScan(basePackages = { "br.com.oauth2.exemplo" })
11 @EntityScan(basePackages = { "br.com.oauth2.exemplo" })
12 @EnableJpaRepositories(basePackages = { "br.com.oauth2.exemplo" })
13 public class AppBoot {
         ▼ # > br.com.oauth2.exemplo
            ▶ ∰ config
            ▶ # > controller
             🕨 🚜 domain
             ▶ # > dto
                                                                                                                            public static void main(String[] args) {
    SpringApplication.run(AppBoot.class, args);
             ▶ # repository
             ▶ ∰ service
                                                                                                                  18
19 }
20
             АррВо
      ▶ @ > src/main/resources
      src/test/java
      ▶ 

■ JRE System Library [JavaSE-1.8]
      Maven Dependencies
      > ≥ src
      b 🗁 target
         R mvnw.cmd
         № > pom.xml
         README.md
```

5) Para executar basta colocar o cursor do mouse sobre a classe e rodar o Run as ou clicar no botão superior com o ícone verde para iniciar :

```
Run Window Help
                                                       | 🐉 🔑 🔳 T
                 -

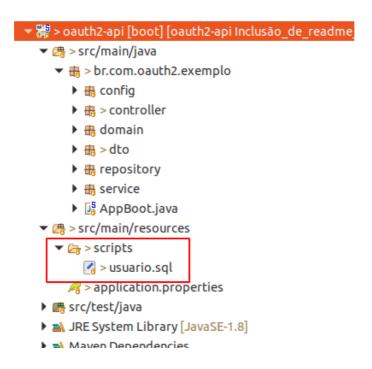
☑ AppBoot.java 
☒

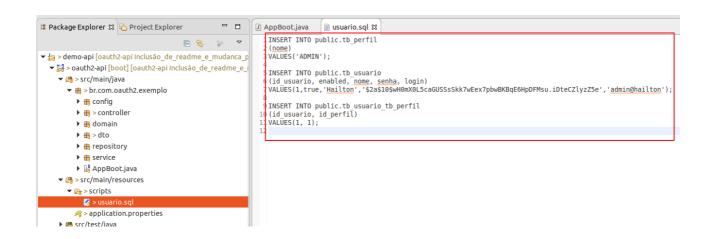
         package br.com.oauth2.exemplo;
       3⊕ import org.springframework.boot.SpringApplication;
ca p
e_1
       9 @SpringBootApplication
      10 @ComponentScan(basePackages = { "br.com.oauth2.exemplo" })
11 @EntityScan(basePackages = { "br.com.oauth2.exemplo" })
         @EnableJpaRepositories(basePackages = { "br.com.oauth2.exemplo" })
      13 public class AppBoot {
      14
      159
              public static void main(String[] args) {
      16
                  SpringApplication.run(AppBoot.class, args);
      17
      18
      19
      20
```

```
& × %
■ Console X = Progress Problems Search Prominal
oauth2-api - AppBoot [Spring Boot App] /usr/lib/jvm/java-8-oracle/bin/java (14 de mar de 2019 23:41:21)
                           (v1.5.3.RELEASE)
 :: Spring Boot ::
                         INFO 15370 ---
2019-03-14 23:41:21.586
                                                      main] br.com.oauth2.exemplo.AppBoot
                                                                                                          Starting AppBoo
2019-03-14 23:41:21.588
                          INFO 15370 ---
                                                      main] br.com.oauth2.exemplo.AppBoot
                                                                                                          No active profi
2019-03-14 23:41:21.624
                          INFO 15370 ---
                                                      {\tt main]} \ at ion {\tt ConfigEmbeddedWebApplicationContext}:
                                                                                                          Refreshing org.:
2019-03-14 23:41:22.451
                          INFO 15370 ---
                                                                                                          JSR-330 'javax.
                                                      mainl f.a.AutowiredAnnotationBeanPostProcessor
2019-03-14 23:41:22.563
                          INFO 15370 ---
                                                      main] trationDelegate$BeanPostProcessorChecker
                                                                                                          Bean 'org.spring
2019-03-14 23:41:22.599
                                                      main] trationDelegate$BeanPostProcessorChecker
                                                                                                          Bean 'security.
                          INFO 15370 ---
2019-03-14 23:41:22.601
                          INFO 15370 ---
                                                      main] trationDelegate$BeanPostProcessorChecker
                                                                                                          Bean 'org.spring
2019-03-14 23:41:22.610
                          TNFO 15370 ---
                                                      main] trationDelegate$BeanPostProcessorChecker
                                                                                                          Bean 'resourceSo
2019-03-14 23:41:22.760
                          INFO 15370 ---
                                                      mainl s.b.c.e.t.TomcatEmbeddedServletContainer
                                                                                                          Tomcat initiali:
2019-03-14 23:41:22.767
                          INFO 15370 ---
                                                      main] o.apache.catalina.core.StandardService
                                                                                                          Starting service
2019-03-14 23:41:22.768
                                                      main] org.apache.catalina.core.StandardEngine
                                                                                                          Starting Servle
                          INFO 15370 ---
2019-03-14 23:41:22.834
                          INFO 15370 ---
                                          [ost-startStop-1] o.a.c.c.C.[.[localhost].[/oauth2-api]
                                                                                                          Initializing Sp
                          INFO 15370 --- [ost-startStop-1] o.s.web.context.ContextLoader
INFO 15370 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean
2019-03-14 23:41:22.834
                                                                                                          Root WebApplica
2019-03-14 23:41:22.951
                                                                                                          Mapping filter:
2019-03-14 23:41:22.951
                          INFO 15370 ---
                                          [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean
                                                                                                          Mapping filter:
2019-03-14 23:41:22.951
                          INFO 15370 ---
                                          [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean
                                                                                                          Mapping filter:
2019-03-14 23:41:22.952
                          INFO 15370 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean
                                                                                                          Mapping filter:
                          INFO 15370 --- [ost-startStop-1] .s.DelegatingFilterProxyRegistrationBean
2019-03-14 23:41:22.953
                                                                                                          Mapping filter:
2019-03-14 23:41:22.953 INFO 15370 --- [ost-startStop-1] o.s.b.w.servlet.ServletRegistrationBean
                                                                                                          Mapping servlet
```

Execute o script dentro do projeto após o start da aplicação, nesse momento as tabelas já serão criadas no banco.

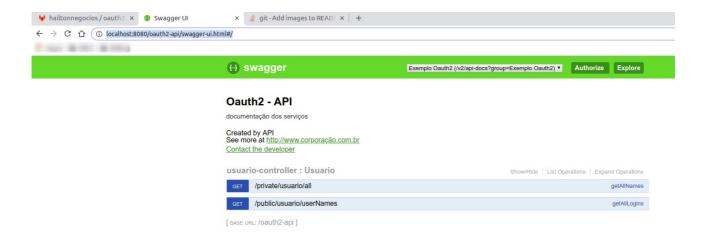
Se estiver criado, execute o script demo para ter alguma carga no banco:





6) Após executar os passos anteriores acesse a URL abaixo :

http://localhost:8080/oauth2-api/swagger-ui.html

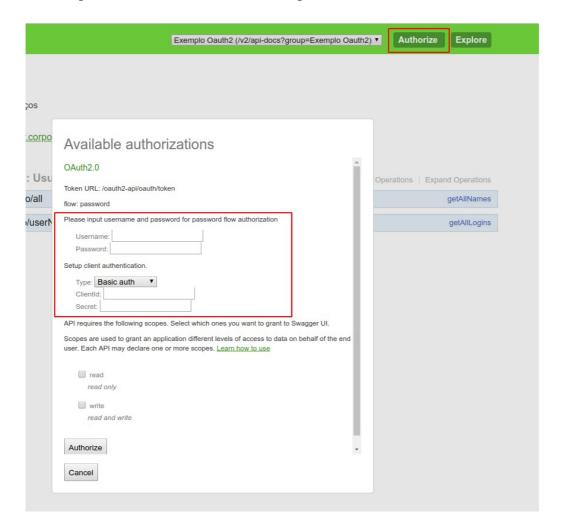


7) No projeto foi incluído dois tipos de endpoint, um privado com autenticação Oauth2 e outro publico sem a necessidade de autenticação.

Endpoint Privado:



Como é privado, click no botão Authorize para incluir as credenciais



Digite as credenciais demos abaixo:

Essa primeira credencia é o usuario criado no banco de dados

username: admin@hailton

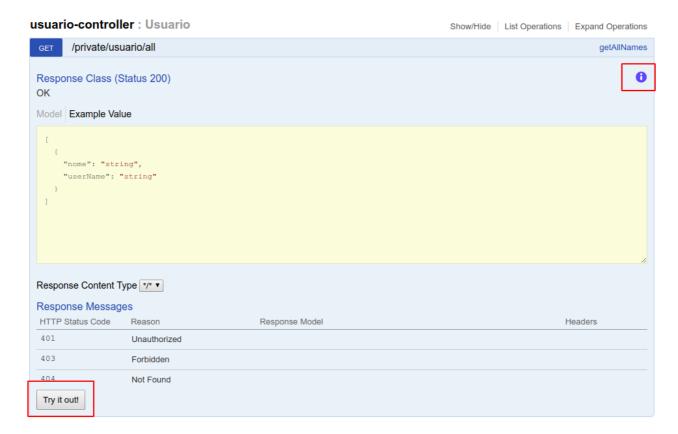
password: 123456

Essa segunda credencial esta no arquivo de propertie dentro do projeto

clientId : admin secret : 123456

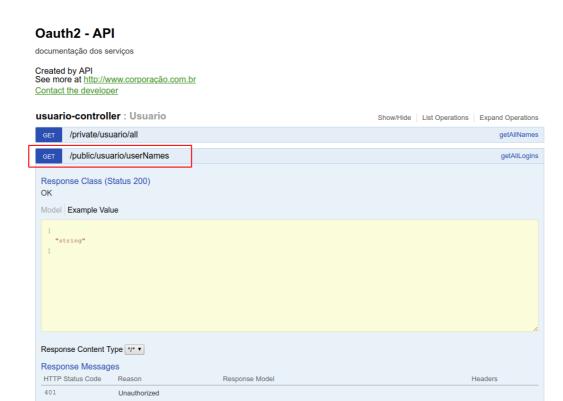
read : X write: X

Após a autenticação o botão ficará azul, assim você poderá executar o consumo :

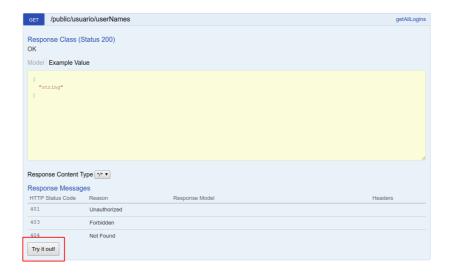


Retorno da api private / usuario / all

Endpoint Publico



Como é publico então não precisa de autenticação, basta clicar direto no botão try it out



Forbidden

Not Found

404

Try it out!

Retorno da api /public / usuario / userNames

```
Response Body

[ "admin" ]

Response Code
```