**Heroku**

A Heroku é uma plataforma nuvem que faz deploy de várias aplicações back-end seja para hospedagem ou testes em produção de aplicações. Também tem integração com o GitHub, deixando seu uso mais fácil.

**IMPORTANTE**: Para seguir esse tutorial será preciso ter uma conta no Git, assim, caso ainda não tenha, siga o tutorial para instalação e uso do Git disponível. Após continue esse tutorial.

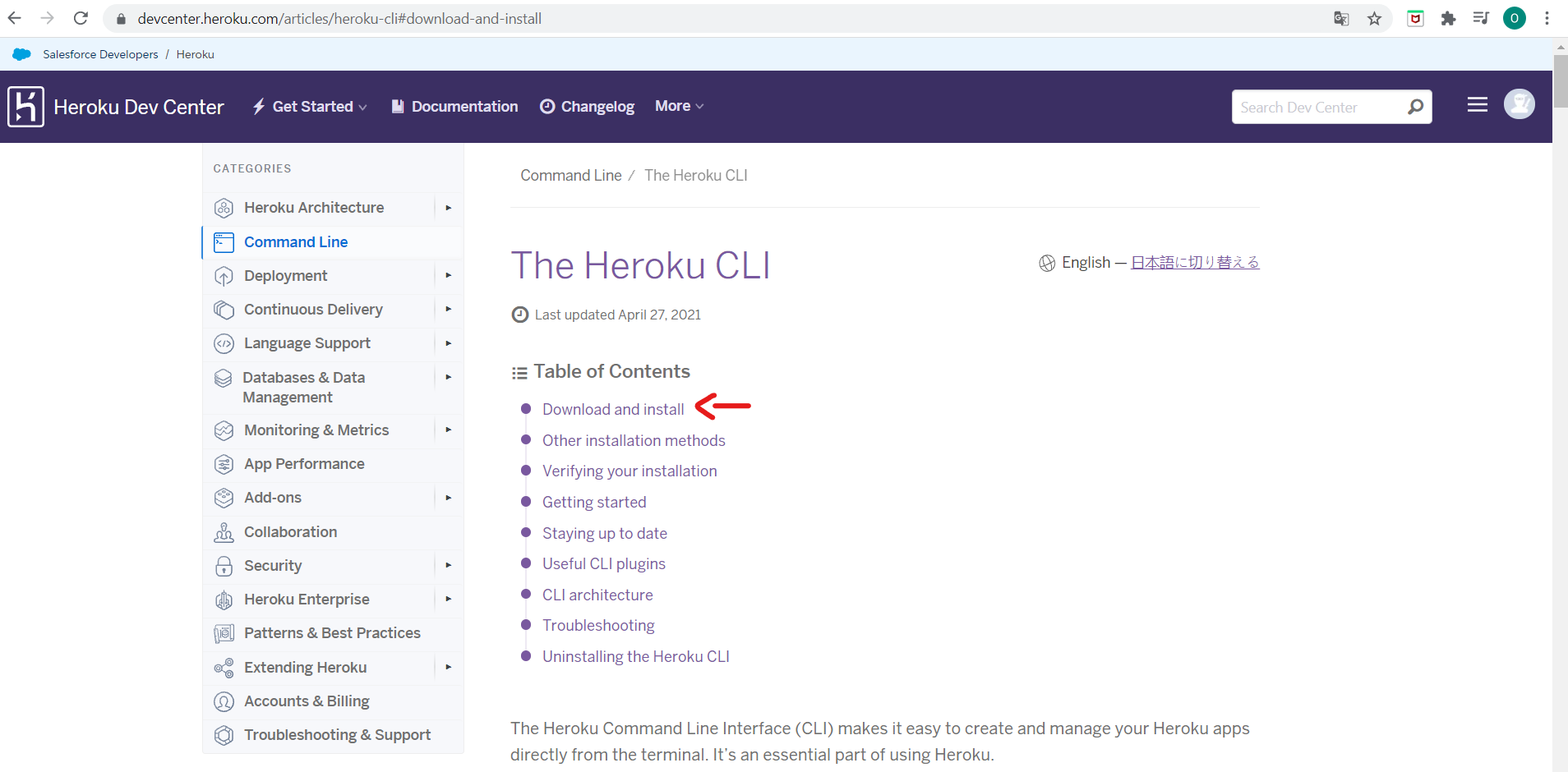
Entrar em heroku.com e inscrever-se (criar conta grátis).



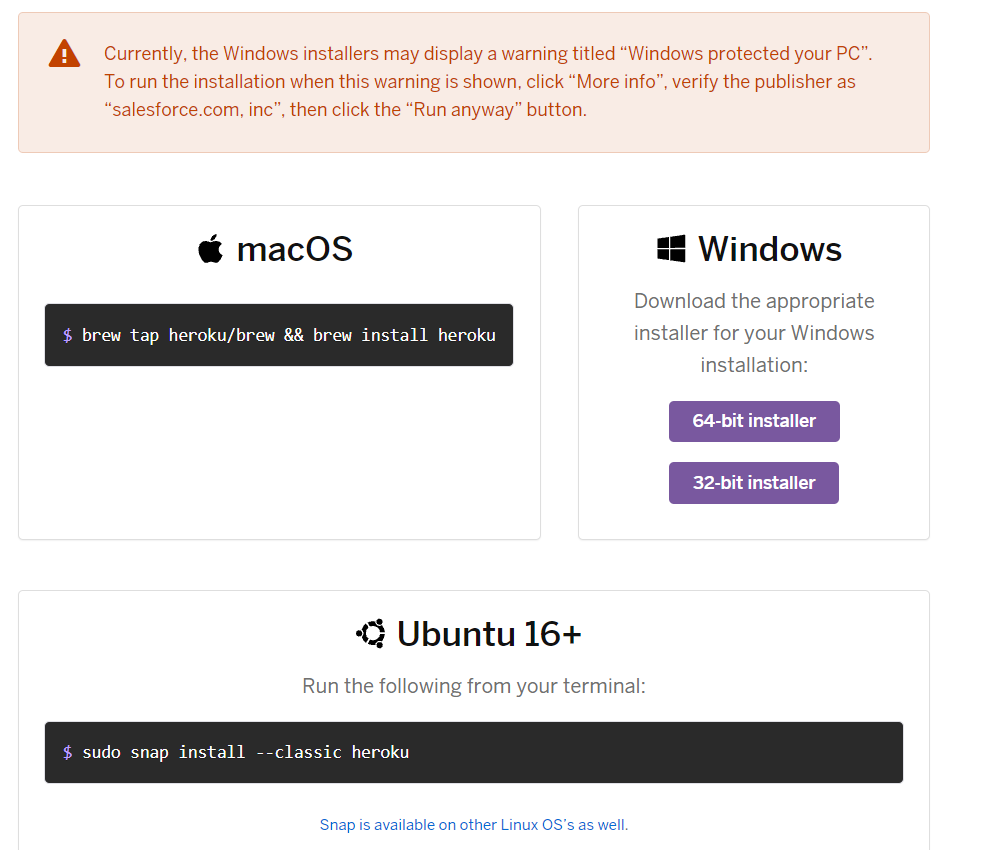
Depois de fazer a inscrição, receberá uma mensagem para confirmar a inscrição e ativar a conta. Em seguida é necessário definir a senha e concordar com os termos do serviço.

O passo seguinte é instalar o heroku cli.

Para o download entrar em: <https://devcenter.heroku.com/articles/heroku-cli> e escolher **Download and Install**, escolher a plataforma



Em seguida escolher a versão, de acordo com o SO.



Baixar o instalador e instalar.

Após terminar a instalação, no prompt pode conferir a versão instalada com

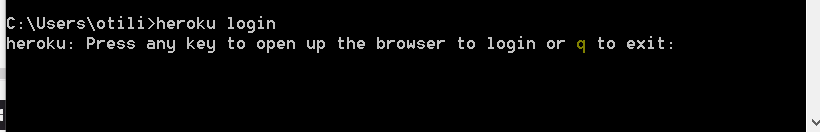
heroku --version



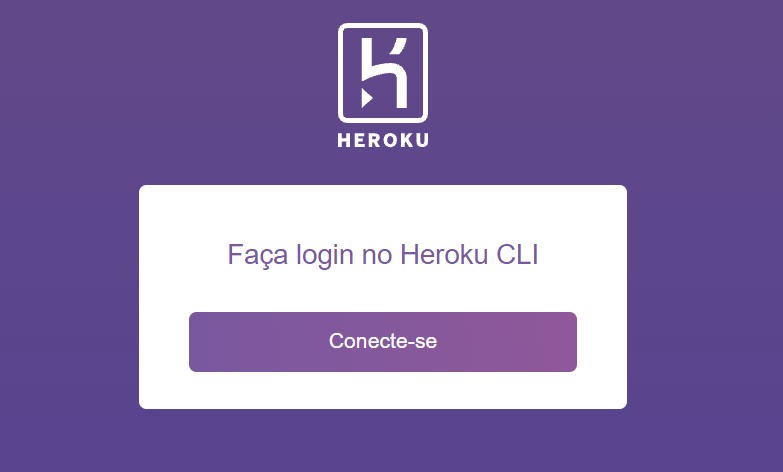
Em seguida vamos digitar

heroku login

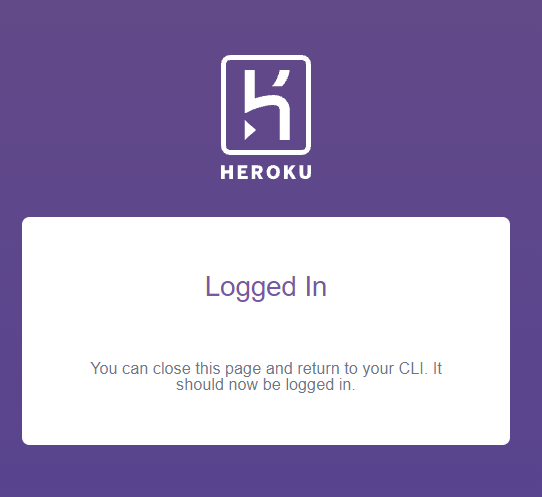
e receberemos a seguinte mensagem,



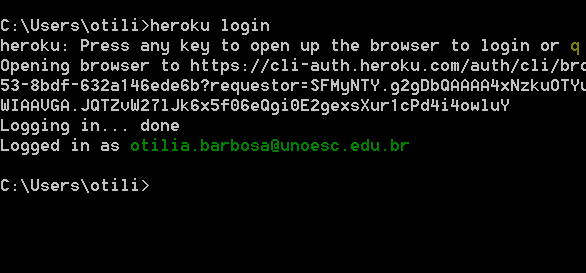
Podemos pressionar qualquer tecla para em seguida fazer o login na página do Heroku :



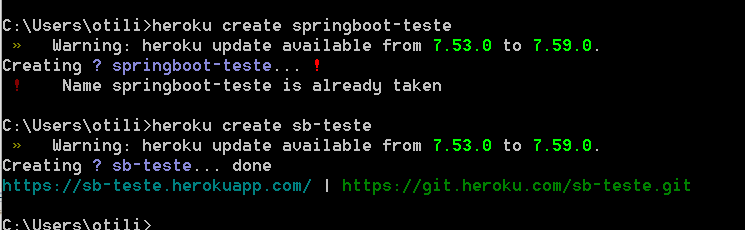
Precisamos fazer o login e em seguida receberemos a orientação de retornar para o prompt.



Ao retornar, o login estará ok.



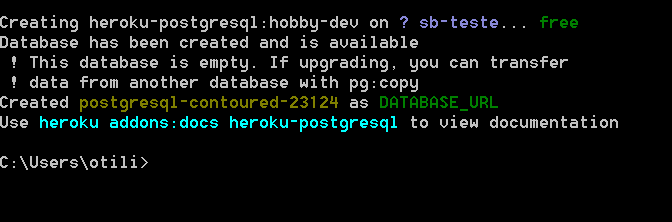
Precisamos criar um repositório com um nome disponível. Nesse caso o primeiro nome escolhido não estava disponível. Note que você deverá escolher um nome disponível (sb-teste também não estará mais disponível).



O passo seguinte é associar o banco de dados ao projeto assim (lembrar que sb-teste é o nome da aplicação que foi criada no passo anterior e no seu caso deverá ser outro nome)

heroku addons:create heroku-postgresql:hobby-dev --app **sb-teste**

E receberemos o retorno:



Nesse momento, vamos precisar modificar o projeto para informar a nova forma de acesso ao banco de dados. No Eclipse, vamos entrar no arquivo **application.properties** e modificar assim. As linhas comentadas (exceto a linha de título) devem ser utilizadas para rodar o projeto de forma local e a linha **spring.datasource.url=${DATABASE\_URL}** será utilizada pelo Heroku (conforme informado quando da criação acima).

#Spring Boot server configuration

#spring.datasource.url= jdbc:postgresql://localhost:5433/spring-banco

**spring.datasource.url=${DATABASE\_URL}**

#spring.datasource.username=postgres

#spring.datasource.password=postgres

...

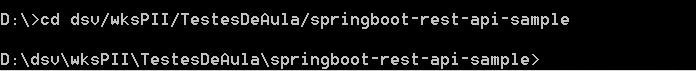
#server.address=0.0.0.0

#server.port=8081

O passo seguinte é entrar na pasta onde está o projeto (confira o seu!) com

cd\

cd pasta\subpasta\nome do projeto



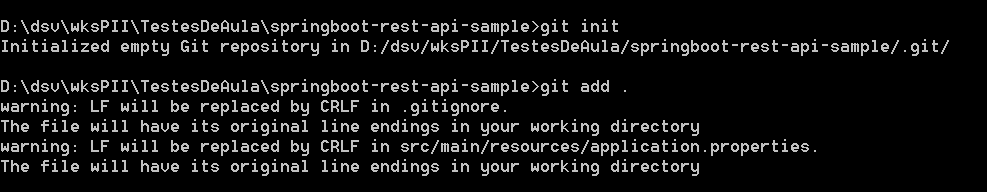
É preciso criar na raiz do projeto um arquivo chamado **system.properties** com uma linha indicando a versão do Java que foi utilizada para a construção do projeto, por exemplo:

**java.runtime.version=15**

Em seguida usamos (usar exatamente as linhas como descrito)

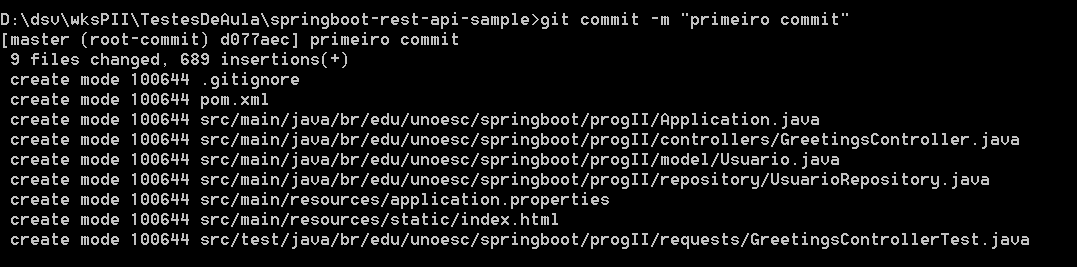
git init

git add .



Em seguida faremos o primeiro commit com:

git commit -m "primeiro commit"



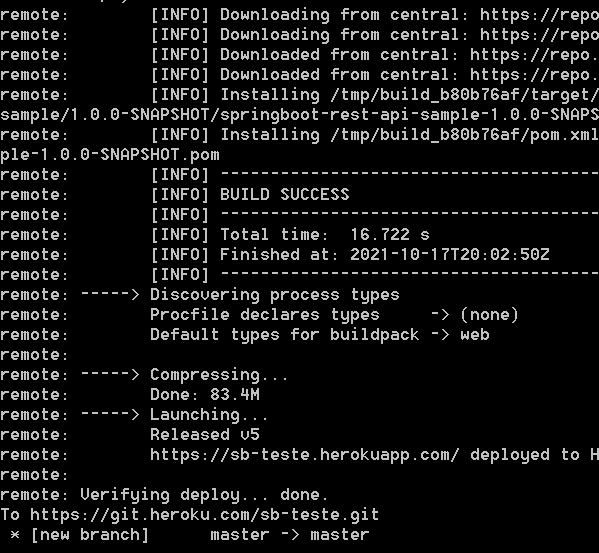
Depois usaremos a linha abaixo (lembrar de usar o nome configurado no começo):

heroku git:remote -a **sb-teste**

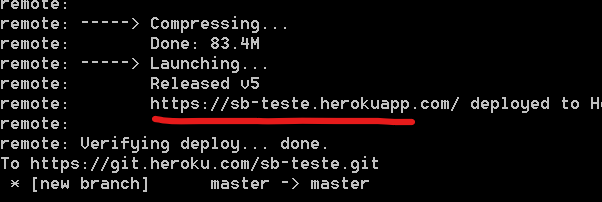
E em seguida :

git push heroku master

Depois de muitas linhas de retorno teremos:



A linha que nos interessa é

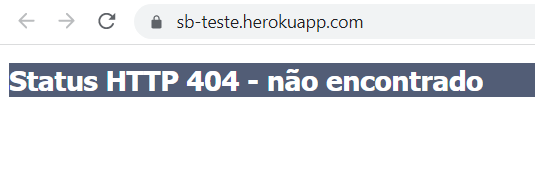


Essa linha marcada é a que usaremos para rodar a aplicação.

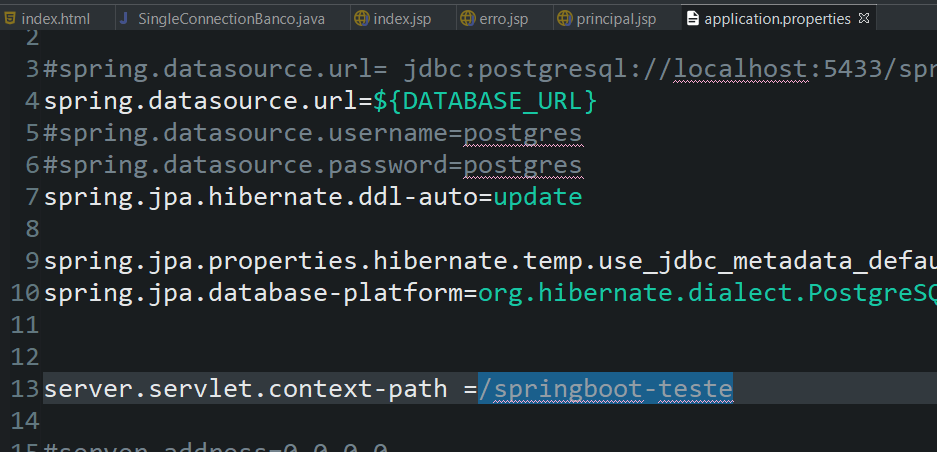
Vamos digitar:

heroku open

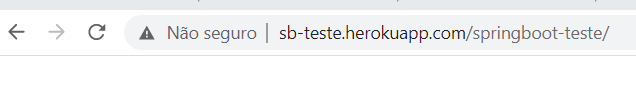
O browser será aberto assim:



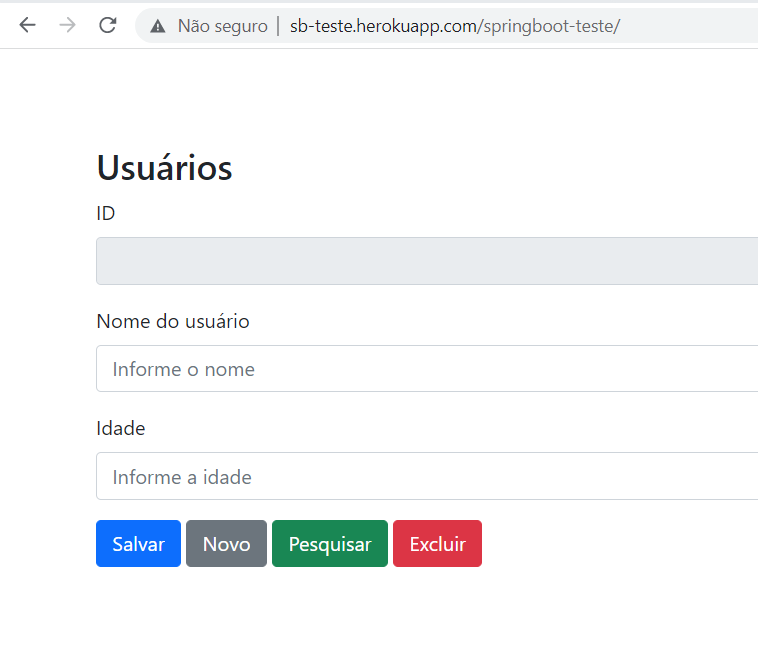
Precisamos acrescentar o nome da aplicação que está no application.properties



Assim:



E teremos a aplicação rodando no Heroku como rodava de forma local anteriormente.



http://sb-teste.herokuapp.com/springboot-teste/