Deploy aplicação com banco de dados

Deploy do banco de dados não persistente

1. Abra o terminal do Openshift e digite o comando abaixo;
   1. oc new-app -e POSTGRESQL\_USER=luke -ePOSTGRESQL\_PASSWORD=secret -ePOSTGRESQL\_DATABASE=my\_data centos/postgresql-10-centos7
2. Após criar o buildconfig da aplicação, abra a “Topology” e procure pelo aplicativo my-database.

Deploy da aplicação;

1. Abra o terminal do Openshift, e digite o comando abaixo;
   1. oc new-app <https://github.com/sclorg/nodejs-ex>
2. Após criar o buildconfig da aplicação, abra a “Topology” e procure pelo aplicativo nodejs-ex;
3. Agora no terminal do Openshift, exponha o serviço com o seguinte comando;
   1. oc expose service nodejs-ex;
4. Procure pela rota exposta, nos detalhes da aplicação;
5. Acesse o aplicativo.

Deploy do banco de dados persistente

1. Abra o terminal do Openshift, e digite o comando abaixo;
   1. oc new-app postgresql-persistent -e POSTGRESQL\_USER=luke -e POSTGRESQL\_PASSWORD=secret -e POSTGRESQL\_DATABASE=my\_data --name=postgresql
2. Após criar o buildconfig da aplicação, abra a “Topology” e procure pelo aplicativo postgresql;
3. No menu lateral procure pelo ultimo item; “Configmap”;
4. Crie um configmap com o nome “nodejs-custom-envs”;
5. Adicione a seguinte chave e valor;
   1. Chave = POSTGRESQL\_SERVICE\_HOST
   2. Valor = postgresql
6. Mude o perfil para “Administrator”;
7. Em “Workloads” acesse “Deployments”;
8. Procure pelo depoloyment “nodejs-ex”;
9. Após encontrar o deploymente, navege pela aba “Environments”;
10. Aqui podemos associar nosso configmap em nossa aplicação;
11. Clique em “Add all from ConfigMap or Secret” e procure pelo configmap criado anteriormente “nodejs-custom-envs”;
12. Salve a alteração;
13. Volte para o perfil de “Developer” e veja na “Topology” se o nodejs-ex re-criou um novo pod;
14. Caso o novo pod já tenha subido, teste novamente a aplicação.