

Bases de Dados

Lab01: Instalação

Para realizar o projeto e laboratórios da cadeira de Bases de Dados precisa de ter acesso a um servidor PostgreSQL, bem como um servidor Web.

A opção que recomendamos é a instalação do ambiente de trabalho da disciplina, baseado em *containers* Docker, no seu computador portátil.

Pode, em alternativa, utilizar o servidor PostgreSQL do IST para realizar os laboratórios baseados em SQL. No entanto, a utilização do servidor Web do IST é inviável devido à falta de bibliotecas Python necessárias para o funcionamento das aplicações desenvolvidas no âmbito do projeto da disciplina.

Instalação do Ambiente de Trabalho

1. Pré-requisitos

1.1. Instalar Docker Desktop

[Docker Desktop](#) é uma aplicação para máquinas MacOS, Linux e Windows para a construção e partilha de aplicações e microserviços em containers.

Instruções de instalação para:

- [Mac](#)
- [Windows](#)
- [Linux](#)

Nota: o Docker Desktop tem de estar aberto para instalar e utilizar o ambiente de trabalho

1.2. Instalar Git

[Git](#) é um sistema de controlo de versões distribuído, de código aberto, projetado para lidar com projetos pequenos e muito grandes com velocidade e eficiência.

Instruções de instalação para:

- [Mac](#)
- [Windows](#)
- [Linux](#)

Pode em alternativa instalar o [GitHub Desktop](#) para Windows ou Mac, caso prefira uma interface gráfica.

2. Instalação do Ambiente Base

1. Abra um terminal e altere o diretório de trabalho atual para a localização onde deseja clonar o repositório bdist-workspace

NOTA: em Windows deve abrir um terminal do Windows e não do WSL, e escolher um diretório do sistema Windows

2. Execute o seguinte comando para clonar o repositório:

```
git clone https://github.com/bdist/bdist-workspace.git
```

3. Altere o diretório de trabalho atual para a localização do diretório clonado:

```
cd bdist-workspace/
```

4. A partir do diretório, inicie o bdist-workspace executando o comando:

```
docker compose up
```

NOTA: os serviços ficam anexados ao Terminal, e portanto os seus logs serão impressos na janela do terminal. Para encerrar de forma segura, pressione CTRL+C na janela do Terminal e aguarde até que todos os serviços parem corretamente. Em alternativa, pode acrescentar a opção **-d** ao comando anterior, por forma a desanexar os serviços do terminal.

3. Jupyter Notebook

3.1. Acesso

O serviço Jupyter Notebook é executado na porta 9999, com autenticação por token, para proteger o seu acesso. É necessário o token de autenticação sempre que o bdist-workspace for iniciado.

Pode encontrar o seu token de autenticação na janela do Terminal, ou alternativamente nos logs do container bdist-workspace-notebook-1 na guia Containers da aplicação Docker Desktop (sugestão: faça CTRL+F e procure por 'token').

O URL de acesso ao Jupyter Notebook com o token de autenticação terá o seguinte formato:

```
http://127.0.0.1:9999/lab?token=SEU_TOKEN_DE_AUTENTICACAO
```

Clique no URL impresso por último, ou copie-o para a barra de endereços do seu navegador.

3.2. Serviços

Através do Jupyter Notebook tem acesso ao PostgreSQL por meio do terminal, bem como acesso ao serviço pgAdmin.

Iremos disponibilizar guiões de laboratório em formato ipynb, que pode carregar, preencher e executar diretamente no Jupyter. Deve colocar no diretório images/ as imagens que disponibilizaremos para complementar os guiões de laboratório.

4. PostgreSQL

4.1. Acesso

O PostgreSQL, como a maioria dos sistemas de gestão de base de dados, dispõe de uma interface de linha de comando através da qual é possível executar instruções SQL e também outros comandos de administração e manutenção do sistema.

1. Clique no botão New Launcher no Jupyter e abra um Terminal (secção Other).
2. Ligue-se ao PostgreSQL usando o cliente de linha de comandos psql.

```
psql -h postgres -U postgres
```

3. Introduza a password do utilizador postgres, que é também 'postgres'

4.2. Comandos

O PostgreSQL inclui comandos de administração, que começam com \, e instruções SQL, que incluem comandos como SELECT, CREATE ou INSERT, que exploraremos ao longo da disciplina, e que terminam sempre com ;

A lista de comandos de administração é a seguinte:

<code>\?</code>	Obter informação sobre todos os comandos de administração.
<code>\h</code>	Obter informação sobre todos os comandos SQL disponíveis.
<code>\q</code>	Sair da linha de comando do PostgreSQL.
<code>\l</code>	Listar todas as bases de dados existentes no sistema.
<code>\c <i>database</i></code>	Ligar à base de dados indicada.
<code>\d</code>	Listas as tabelas da base de dados atual.
<code>\d <i>table</i></code>	Obter informação sobre a estrutura de uma tabela.
<code>\timing</code>	Mostrar o tempo de execução das consultas.
<code>\i <i>file</i></code>	Executar os comandos contidos no ficheiro indicado.

4.3. Criação da Base de Dados Bank

1. Crie um utilizador bank sem privilégios:

```
CREATE USER bank WITH PASSWORD 'bank';
```

2. Crie a base de dados bank e defina o utilizador bank como dono da base de dados:

```
CREATE DATABASE bank WITH OWNER = bank ENCODING = 'UTF8';
```

Nota: Defina sempre o encoding de caracteres explicitamente como UTF-8.

3. Conceda ao utilizador bank todos os privilégios para a base de dados bank:

```
GRANT ALL ON DATABASE bank TO bank;
```

4. Saia do programa com o comando `\q` e volte a entrar com a conta bank:

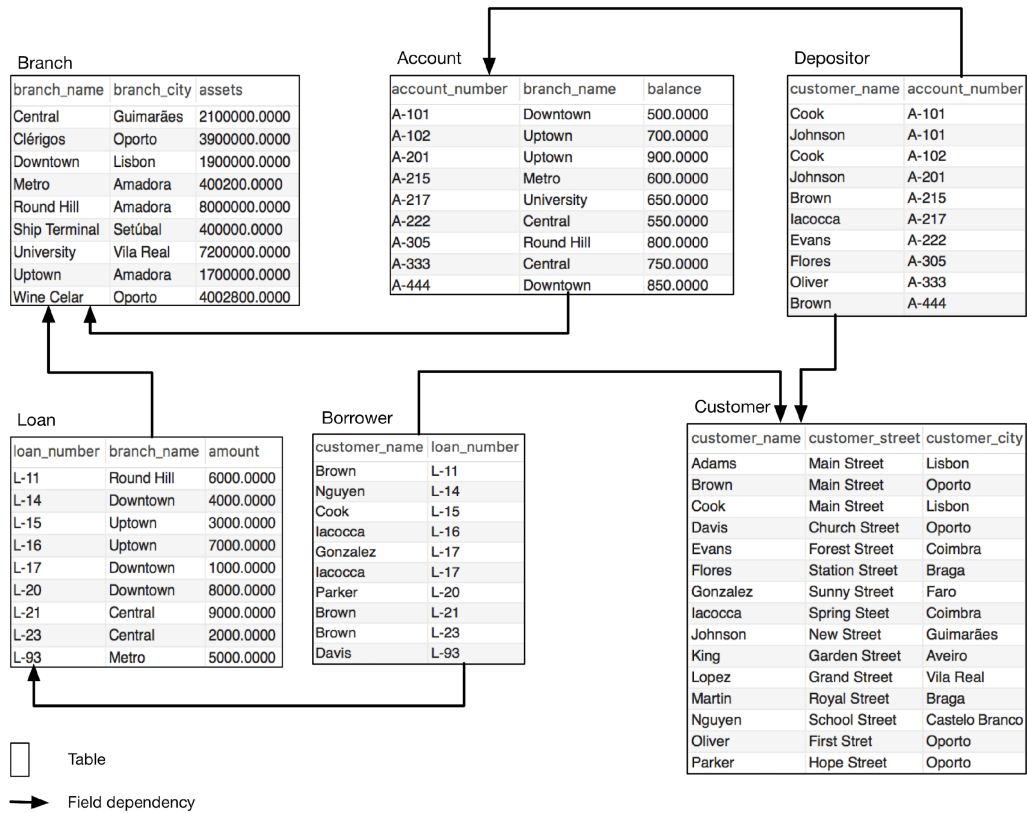
```
psql -h postgres -U bank
```

5. Introduza a password do utilizador bank, que é também 'bank'
6. Copie o ficheiro bank.sql (disponível na secção 'Laboratórios' da página da disciplina no Fénix) para o diretório data/ do Jupyter e execute-o no PostgreSQL:

```
\i ~/data/bank.sql
```

Este ficheiro consiste numa série de comandos CREATE TABLE que especificam o esquema da base de dados bank, bem como uma série de comandos INSERT, que populam as tabelas com dados.

Table Diagram for the BANK Example



NOTE: Values are only illustrative. The actual database can be different.

Uma vez executado, use o comando \d para ver a estrutura de tabelas.

Para cada tabela, pode executar o seguinte comando para visualizar os seus dados:

```
SELECT * FROM nome_da_tabela;
```

5. pgAdmin

pgAdmin é a plataforma de administração e desenvolvimento de código aberto mais popular e rica em recursos para o PostgreSQL

1. Aceda ao pgAdmin a partir do Jupyter
2. Na página de login, insira:

- Nome de utilizador: pgadmin@tecnico.pt
 - Senha: pgadmin
3. Clique no botão Adicionar Novo Servidor.
 4. Defina o Nome na guia Geral como o nome do container a correr.
 5. Defina o Nome do Host na guia de Conexão como postgres.
 6. Utilize o mesmo nome de utilizador e senha que forneceria para o psql (postgres ou bank).

6. Flask Web App

1. Execute o seguinte comando para criar o clone local da aplicação web:

```
git clone https://github.com/bdist/app.git
```

Nota: certifique-se de que coloca o diretório app/ ao lado do diretório bdist-workspace/, no mesmo diretório-pai.

2. Entre na pasta bdist-workspace:

```
cd bdist-workspace
```

3. O arquivo docker-compose..app.yml contém as instruções para iniciar o workspace com a aplicação web. Re-lance o workspace com o seguinte comando para usá-lo:

```
docker compose -f docker-compose..app.yml up
```

4. Verifique se o aplicativo está em execução colocando o seguinte URL na barra de endereços do seu navegador:

```
http://127.0.0.1:8080/ping
```

5. Deve receber uma resposta HTTP no formato JSON semelhante a esta

```
{  
  "message": "pong!"  
}
```

6. Tente modificar a mensagem em ../app/app.py enquanto ele estiver em execução. Nos logs, verifique se é acionado um reload quando guarda as alterações.

7. Resolução de Problemas

O documento [README.md](#) do repositório <https://github.com/bdist/bdist-workspace> contém uma secção de FAQs e resolução de problemas comuns.

Acesso ao PostgreSQL do IST

1. Pré-requisitos

Aceda à página de self-service da DSI em <https://selfservice.dsi.tecnico.ulisboa.pt/> e ative os serviços shell, web e cgi

2. Acesso ao Sigma

1. Para aceder à infraestrutura Cloud do IST tem de usar um programa designado genericamente por “Secure Shell” que permite estabelecer uma sessão segura com o *host*/anfitrião **sigma.tecnico.ulisboa.pt**.
 - **Windows, Linux ou Mac OS:** Abra um terminal e execute o comando

```
ssh istxxxx@sigma.tecnico.ulisboa.pt
```

Nota: A autenticação no **sigma** é sempre efetuada através as suas credenciais do Fénix (nome de utilizador e password do sistema Fénix). Por razões de segurança, ao digitar a password, os caracteres não aparecem no ecrã.

Nota: Na primeira vez que estabelece uma ligação segura com um novo servidor (host) ser-lhe-á solicitado que aceite a chave ssh. Pode responder afirmativamente. Na maior parte dos casos, isto corresponde a escrever ‘yes’ ou a pressionar o botão ‘Ok’.

2. Para obter a password da sua conta no PostgreSQL no sigma, execute o comando:

```
psql_reset
```

3. Para entrar em sessão em sessão no PostgreSQL use o comando:

```
psql -h db.tecnico.ulisboa.pt -U istxxx
```

Onde **istxxxxx** deve ser substituído pelo seu nome de utilizador no sistema Fénix, e deverá entrar com a password obtida no comando anterior.

4. Se quiser atualizar a password de acesso ao PostgreSQL pode usar o comando:

```
ALTER USER istxxxx WITH PASSWORD 'mypassxxxx';
```

5. Use o comando `\q` para sair do sistema e voltar ao terminal.
6. Faça download do ficheiro bank.sql (disponível na secção 'Laboratórios' da página da disciplina no Fénix) e copie-o para a diretoria raiz (~) da sua área no sigma
- No PC do laboratório, não precisa de transferir o ficheiro para o sigma, pois o download já o colocará lá
 - No seu PC, precisará de transferir o ficheiro para o sigma, por exemplo, por meio de um comando SCP

```
scp <file> istxxxx@sigma.tecnico.ulisboa.pt:<path-to-folder>
```

7. Entre novamente no PostgreSQL e execute o ficheiro bank.sql:

```
\i bank.sql
```