



Olá, aluno(a)!
Seja bem-vindo(a) à aula interativa!

Você entrará na reunião com a câmera e o microfone desligados.

Sua presença será computada através da enquete.
Fique atento(a) e não deixe de respondê-la!

> Módulo II - Coleta e Armazenamento de
Dados em Renda Variável

Segunda Aula Interativa

Prof. Victor A Mancini



Planejamento para hoje

- Relembrar o exemplo prático da Aula 1 -> 5 min
- Instalação PostgreSQL e DBeaver -> 20 min
- Arquitetura de código -> 20 min
- Orquestração de fluxos de dados com Airflow -> 35 min
- Apresentar o desafio prático -> 10 min



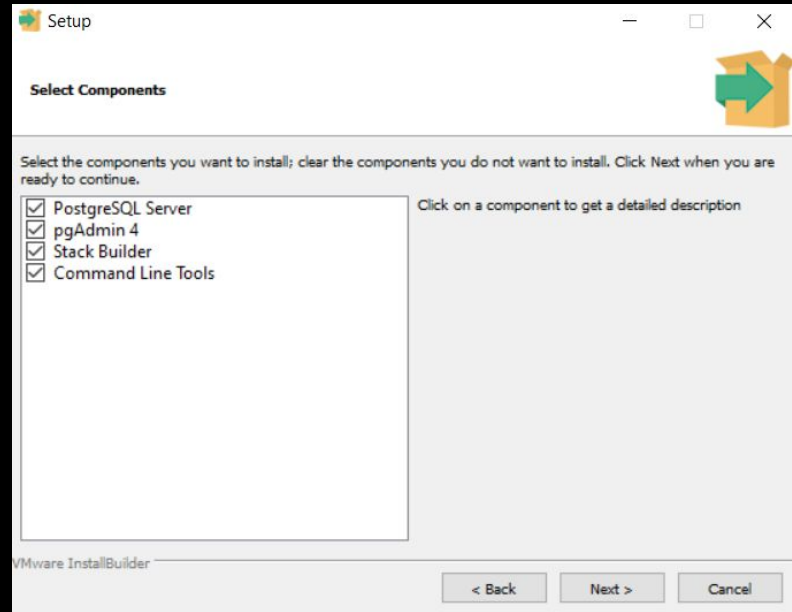
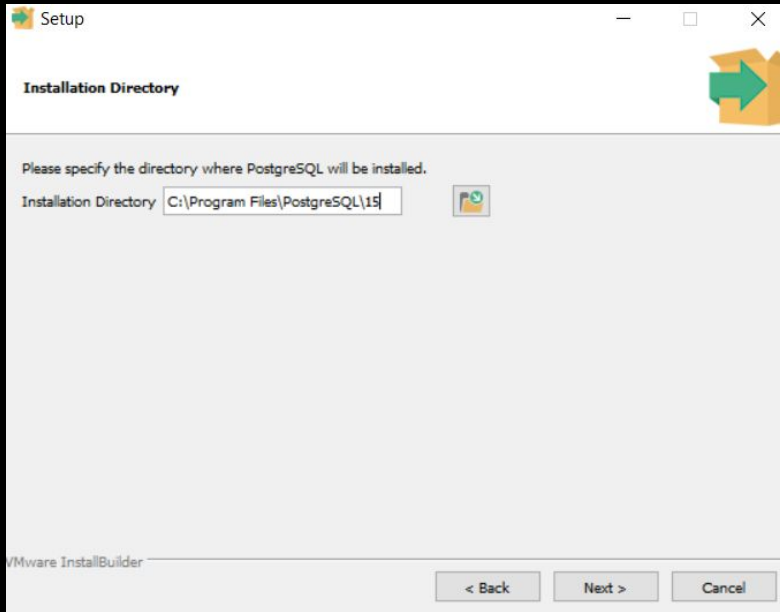
- Exemplo prático: coleta e armazenamento de dados de fundos de investimento
- 1. Coletar histórico de cotas de fundos da CVM;
- 2. Armazenar os dados brutos no data lake;
- 3. Processar os dados brutos e armazenar os dados de interesse no banco de dados MySQL;
- 4. Ler os dados processados em Python;
- 5. Rodar uma [análise de risco de fatores](#).





- Instalação PostgreSQL e DBeaver
- <https://www.postgresql.org/>







Setup

Data Directory

Please select a directory under which to store your data.

Data Directory:

VMware InstallBuilder

< Back Next > Cancel

Setup

Password

Please provide a password for the database superuser (postgres).

Password:

Retype password:

VMware InstallBuilder

< Back Next > Cancel





Setup

Port

Please select the port number the server should listen on.

Port

VMware InstallBuilder

< Back Next > Cancel

Setup

Advanced Options

Select the locale to be used by the new database cluster.

Locale

VMware InstallBuilder

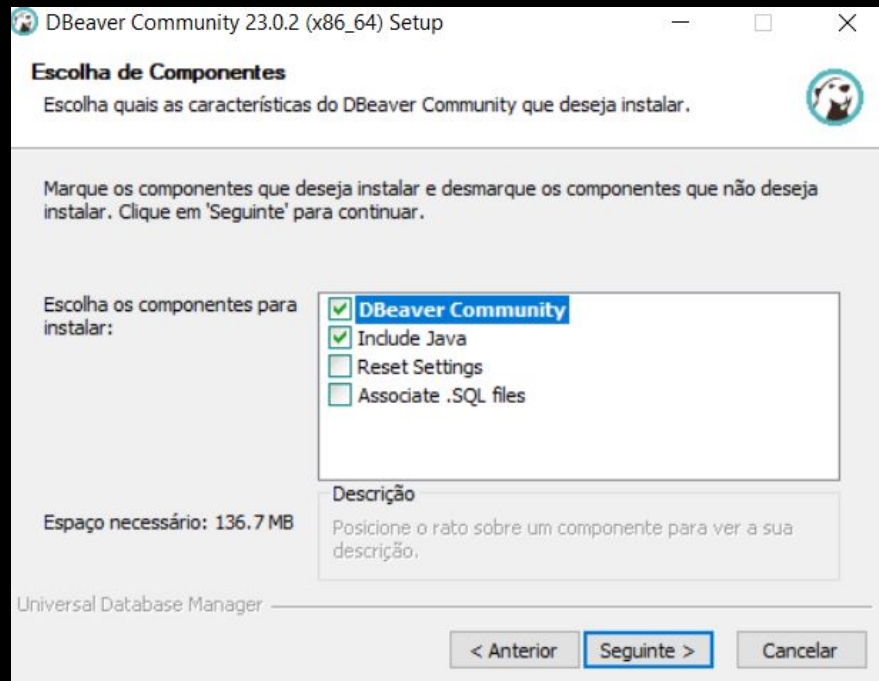
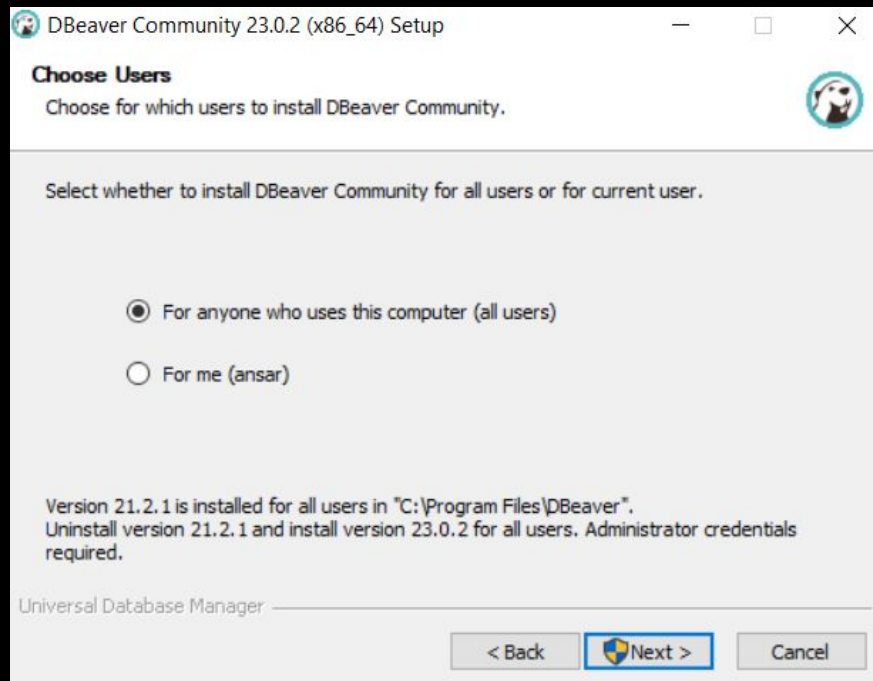
< Back Next > Cancel

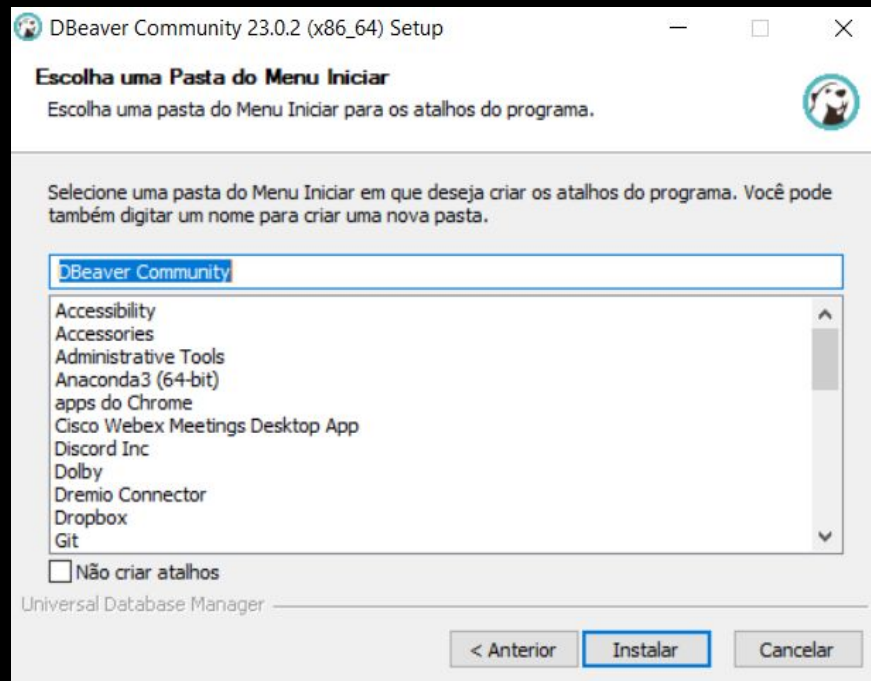
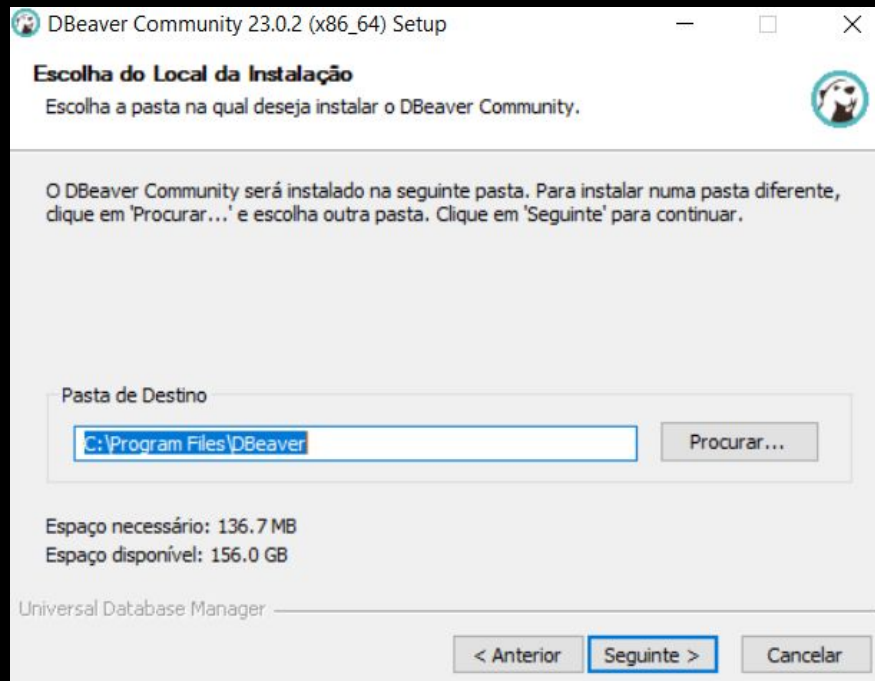




- Instalação PostgreSQL e DBeaver
- <https://dbeaver.io/>

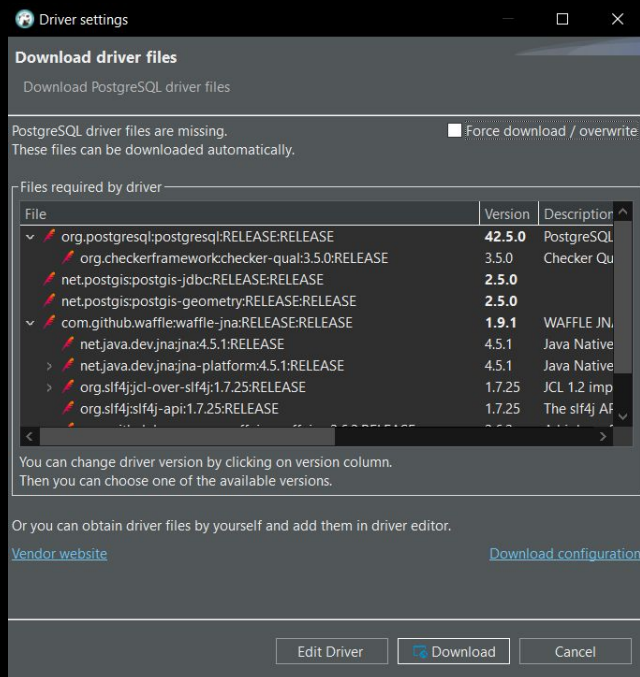








- Configurando DBeaver





- Arquitectura de código
- [Clean architecture](#)



- Orquestração de fluxos de dados com Airflow
- Pré-requisitos
 - [Instalar Linux no Windows;](#)
 - [Instalar pip e Python no Linux;](#)
 - [Instalar airflow;](#)



● Instalação: Airflow

- Create AIRFLOW_HOME variable
 - `export AIRFLOW_HOME=~/.airflow`
 - `source ~/.bashrc`
- Install Airflow package
 - `AIRFLOW_VERSION=2.2.3`
 - `PYTHON_VERSION="$(python --version | cut -d " " -f 2 | cut -d "." -f 1-2)"`
 - `CONSTRAINT_URL="https://raw.githubusercontent.com/apache/airflow/constraints-${AIRFLOW_VERSION}/constraints-${PYTHON_VERSION}.txt"`
 - `pip3 install "apache-airflow==${AIRFLOW_VERSION}" --constraint "${CONSTRAINT_URL}"`
- Initialize database
 - `airflow db init`



- Desafio prático - Enunciado

Continuando nosso exemplo do Trabalho Prático, agora o gestor de portfólio de ações nos solicitou adicionarmos alguns outros dados à nossa análise de Ambev (ABEV3).

Dessa vez, adicionalmente aos dados de mercado e aos dados macroeconômicos já coletados e armazenados, ele gostaria de trazer dados fundamentalistas para o relatório de análise.



● Desafio prático - Atividades

- Colete e processe os ITRs de Ambev para o período de 2019-2022, utilizando o sistema RAD CVM como fonte de dados (lembre-se que para realizarmos a consulta, precisamos saber o código CVM da companhia);
- Armazene esses dados fundamentalistas em um banco de dados relacional SQL local ou em nuvem;
- Leia essa série histórica do banco de dados SQL usando o Python;
- Filtre apenas os dados de “Receita de Venda de Bens e/ou Serviços” dentro da “Demonstração do Resultado” do “DFs Consolidadas”;
- Gere um pandas DataFrame final apenas com esses valores de “Receita de Venda de Bens e/ou Serviços” e a respectiva data;
- É essa tabela que estamos interessados. Salve-a localmente num arquivo Excel com formato .xlsx.





Muito obrigado!

