

UNIVERSIDADE FEDERAL DE ALAGOAS - UFAL
CAMPUS ARAPIRACA
CIÊNCIA DA COMPUTAÇÃO

Alunos:

- **Gustavo Henrique da Silva Xavier**
- **João Victor Brito da Silva**

Projeto da disciplina: Hibernate (ou DAO)

Os requisitos para o projeto são:

- O modelo conceitual deve ter ao menos 1 relacionamento M-N
- O modelo conceitual deve ter ao menos 1 relacionamento de outra cardinalidade (1-N ou 1-1)
- O modelo conceitual deve ter ao menos uma herança entre entidades
- O projeto deve contemplar o mapeamento objeto-relacional ou implementar o padrão DAO para todo o modelo conceitual
- O projeto deve implementar ao menos duas consultas SQL (ou equivalente, e.g. HQL)

Para este projeto, usamos a ORM SQLAlchemy para Python, e o banco de dados feito usando o MySQL Workbench.

Requisitos para rodar o projeto:

Ter o Python 3.10 instalado. Caso não tenha, link para download:

<https://www.python.org/downloads/>

Ter o MySQL Workbench instalado. Caso não tenha, link para download:

<https://dev.mysql.com/downloads/workbench/>

Ter o driver PyMySQL instalado. Caso não tenha, instale através do comando:

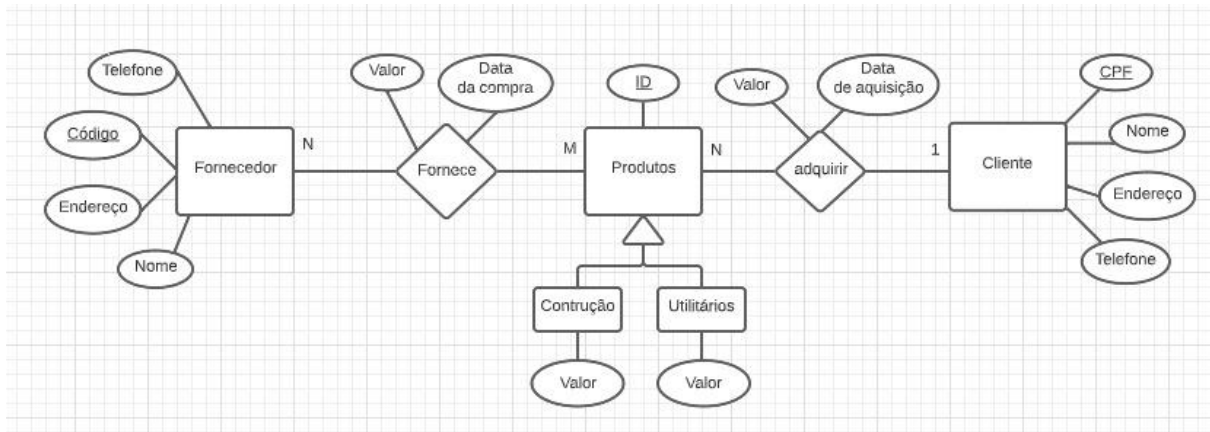
pip install PyMySQL

Ter a biblioteca e ORM SQLAlchemy instalada. Caso não tenha, instale através do comando:

pip install SQLAlchemy

Com todos os requisitos satisfeitos, precisará importar o banco de dados para o MySQL Workbench. Concluída a importação e com o serviço MySQL rodando, conseguirá acessar o Banco de Dados localmente.

Modelo Entidade Relacionamento:



Temos 3 entidades: Fornecedor, Produtos e Cliente.

Temos 2 relacionamentos: fornecer e adquirir. Iram virar tabela por terem atributos.

Na herança optamos por usar somente uma tabela, a Produtos, onde nela adicionaremos o atributo "tipo" que define se é Construção ou Utilitário.

Cliente(cpf, nome, endereco, telefone)

Produtos(id, nome, valor, tipo)

Fornecedor(codigo, nome, telefone, endereco)

fornecer(data_compra, codigo_fornecedorFK, id_produtoFK)

adquirir(data_compra, cpf_clienteFK, id_produtoFK)

Sobre o código:

Para melhor organização o código foi dividido em pastas:

classes: onde estão todas as classes que manipulam o banco de dados;

configuracoes: onde está a conexão com a ORM SQLAlchemy;