

FIAP GRADUAÇÃO

| DAO Pattern

- ▶ Introdução
- ▶ Implementação
- ▶ Generics
- ▶ Exercícios

Introdução

- ▶ O Python é uma linguagem de programação de propósito geral, de alto nível, interpretada, de tipagem dinâmica e com sintaxe fácil de ler.
- ▶ O Oracle é um sistema de gerenciamento de banco de dados relacional (RDBMS) de propriedade da Oracle Corporation.
- ▶ O Python pode ser usado para acessar bancos de dados Oracle de várias maneiras, incluindo:
 - ▶ O pacote `oracledb`
 - ▶ O pacote `cx_Oracle`

O pacote oracledb

- ▶ O pacote oracledb é um pacote Python que fornece uma interface para o Oracle Database.
- ▶ é um pacote nativo, o que significa que ele é compilado para a plataforma específica em que está sendo usado.
- ▶ segundo a documentação é um upgrade do pacote cx_Oracle
- ▶ para instalar no computador, execute o comando `pip install oracledb` ou `pip3` se você está no MAC ou Linux

Exemplo de código de consulta

O seguinte código mostra como usar o pacote `oracledb` para conectar-se ao Oracle Database e recuperar os dados da tabela `TBL_ENQUETE`:

```
1 import oracledb
2
3 conn = oracledb.connect(user='pf0313', password='professor
   #23', dsn="oracle.fiap.com.br/orcl")
4 print("Database version:", conn.version)
5
6 cur = conn.cursor() # Criar um cursor
7 cur.execute('SELECT * FROM EMPREGADO') # Executar uma
   consulta
8
9 rows = cur.fetchall() # Recuperar os resultados da consulta
10 for reg in rows:
11     print(reg)
12
13 cur.close() # Fechar o cursor
14 conn.close() # Fechar a conexão
```

Métodos fetch

- ▶ `fetchall()` - recupera todos os registros da consulta, tome cuidado se a quantidade de registros for muito grande pode faltar memória além de demorar muito tempo
- ▶ `fetchone()` - recupera um registro da consulta
- ▶ `fetchmany(size)` - recupera uma quantidade `size` de registros ou se nenhum valor for passado como parâmetro recupera um número default de registros
- ▶ veja abaixo um exemplo que recupera todos os registros da consulta usando o método `fetchmany()`

```
1 while True:
2     regs = cursor.fetchmany()
3     if not regs:
4         break
5         for reg in regs:
6             print(reg)
```

Inserir, atualizar e excluir

O seguinte código mostra como usar o pacote `oracledb` para inserir, atualizar e excluir registros em uma tabela:

```
1 conn = oracledb.connect(user='pf0313', password='professor
   #23', dsn="oracle.fiap.com.br/orcl")
2 cur = conn.cursor()
3 # Inserir um novo registro
4 cur.execute("INSERT INTO EMPREGADO (numero, nome, cargo,
   data_contratacao, salario, departamento) VALUES (:num, :
   nome, :cargo, to_date(:data, 'YYYY-MM-DD'), :sal, :depto
   )", {'num': 7839, 'nome': 'SCOTT', 'cargo': 'ANALYST', '
   data': '1987-04-19', 'sal': 3000, 'depto': 'TI'})
5
6 # Atualizar um registro existente
7 cur.execute('UPDATE EMPREGADO SET salario = :sal WHERE
   numero = :num', {'sal': 3500, 'num': 7839})
8
9 # Excluir um registro
10 cur.execute('DELETE FROM EMPREGADO WHERE numero = :num', {'
   num': 7839})
11
12 #commit dos dados no banco
13 conn.commit()
   Eduardo Gondo
```

Considerações

- ▶ não esquecer de fechar o cursor e depois fechar a conexão
- ▶ comandos de alteração, inserção e delete devem ser finalizados com o método `commit` da conexão
- ▶ caso contrário, nenhuma informação será gravada
- ▶ temos também o método `rollback()` da conexão que desfaz todas as alterações no registros do banco desde a abertura da conexão

Outra forma de abrir conexões e cursores

- ▶ podemos abrir conexões e cursores usando o comando `with`
- ▶ abrindo com `with` não precisamos fechar explicitamente eles
- ▶ no fim do bloco `with` os recursos são fechados de forma automática
- ▶ veja um exemplo:

```
1 with oracledb.connect(user='user', password='pwd', dsn='
   dbhost.example.com/orclpdb') as conn:
2     with conn.cursor() as cur:
3         cur.execute("insert into categoria values (:id, :
           nome)", {'id':1, 'nome':"têmpero"})
4     conn.commit()
```

Exercícios

1. Crie uma aplicação em Python que simula a inserção de um veículo, peça os dados necessários para o usuário e faça o insert no banco de dados
2. Agora, acrescente funções de alteração, consulta e remoção do veículo. Defina a forma de consulta e como você faz a alteração e como apaga um veículo do banco de dados.
3. Tente pensar em uma regra de negócio para inserir um novo veículo e implemente ela no exercício 1.