Otimização e Balanceamento de Banco de Dados

Aluna: Leticia Macedo Prudente de Carvalho **Turma:** Banco de Dados 2021.1

Atividade: Laboratório 04 – Parse e variáveis embutidas

- Criação da Tabela e Mock de Dados para execução do Laboratório 04:

```
create table lab04_OTB (
    id NUMBER PRIMARY KEY ,
    data_hora VARCHAR(50)
);

create sequence sq_lab04 INCREMENT BY 1 NOCACHE;

CREATE OR REPLACE PROCEDURE insere_lab04_otb IS
    cont int :=0;
    BEGIN
        LOOP

            INSERT INTO lab04_OTB (id, data_hora) values (sq_lab04.NEXTVAL,TO_CHAR(current_timestamp, 'DD/MM/YYYY HH24:MI:SS') );
            cont := cont+1;
            COMMIT;

            EXIT WHEN cont = 100001;
        END LOOP;
END insere_lab04_otb;

--user: System

SET TIMING ON
EXEC insere_lab04_otb;


select count(*) from lab04_OTB ;
```

- Conexão da aplicação Java com o Banco de Dados:

```
public class Conexao {
    public static Connection getConnection() throws SQLException{
        try {
            Class.forName ("oracle.jdbc.OracleDriver");
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
         Connection con = DriverManager.getConnection(
                 "jdbc:oracle:thin:@127.0.0.1:1521:fatec","system","fatec");
         System.out.println("Connection Success");
        return con;
    }


}
```

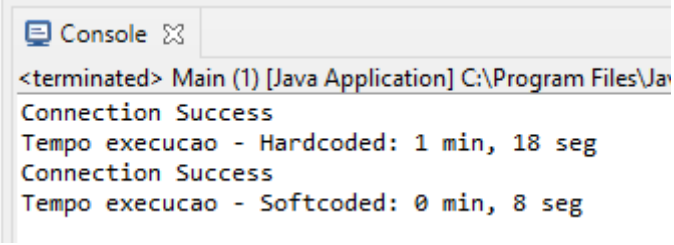
- Hardcoded:

```
public static void hardcoded() throws SQLException {
    Connection con = Conexao.getConnection();
    Long tempoInicio = System.currentTimeMillis() ;
    String query ;
    Statement s = con.createStatement() ;
    ResultSet r ;
    Long cont ;
    String data_hora;

    for (cont = (long) 1 ; cont <= (long) 100000 ; cont++) {
        query="select data_hora from lab04_otb where id=" + Long.toString(cont) ;
        r = s.executeQuery(query) ;

        if(r.next()) {
            data_hora = r.getString(1) ;
        }


        r.close() ;
    }
    s.close() ;

    Long tempoFim = System.currentTimeMillis() ;
    Long tempoExec = tempoFim - tempoInicio;

    String tempoTotal = String.format("%d min, %d seg",
            TimeUnit.MILLISECONDS.toMinutes(tempoExec),
            TimeUnit.MILLISECONDS.toSeconds(tempoExec) -
            TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes(tempoExec))
        );

    System.out.println("Tempo execucao - Hardcoded: " +tempoTotal);


}
```

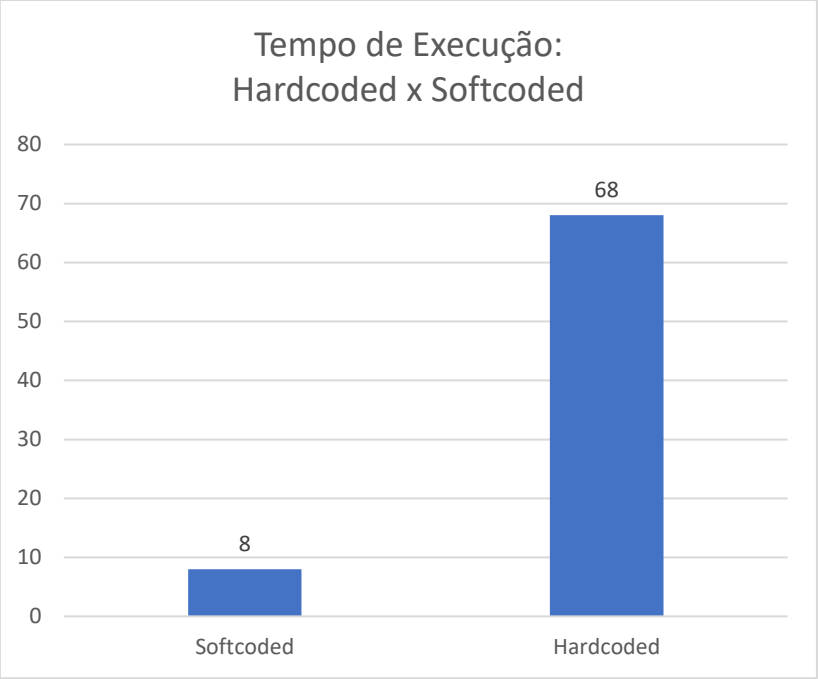
- Softcoded:

```java
public static void softcoded() throws SQLException {
    Connection con = Conexao.getConnection();
    Long tempoInicio = System.currentTimeMillis() ;
    Long cont ;
    String data_hora ;
    PreparedStatement s = con.prepareStatement("select data_hora from lab04_otb where id = ?") ;
    ResultSet r ;

    for(cont = (long) 1 ; cont <= (long) 100000 ; cont++) {
        s.setLong(1, cont);
        r = s.executeQuery();
        if(r.next()) {
            data_hora = r.getString(1) ;
            r.close();
        }
    }
    s.close() ;
    Long tempoFim = System.currentTimeMillis() ;
    Long tempoExec = tempoFim - tempoInicio;

    String tempoTotal = String.format("%d min, %d seg",
            TimeUnit.MILLISECONDS.toMinutes(tempoExec),
            TimeUnit.MILLISECONDS.toSeconds(tempoExec) -
            TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes(tempoExec))
        );

    System.out.println("Tempo execucao - Softcoded: " +tempoTotal);
}
```

- Tempo de Execução:



- Gráfico comparativo entre as duas versões implementadas:



GitHub com o código: https://github.com/leticiaprudente/lab04_OTB