

## Criar transações em diferentes linguagens de programação

(07/12 - apresentação)

O desafio aqui é implementar uma transação em diferentes linguagens de programação. Ao implementar uma transação deve-se ficar atento à questão de como iniciá-la e finalizá-la. Ou seja, algumas linguagens apenas executam a operação de commit quando for explicitamente invocado o comando de finalização. Para compararmos, iremos dividir as principais linguagens de programação entre os grupos e cada grupo irá apresentar como a determinada linguagem de programação se comporta.

Segue um código exemplo de uma transação em JAVA.

```
public class Teste {

    public static void main(String[] args) {

        final String url = "jdbc:postgresql://localhost:5432/postgres"; //indica o caminho do banco
        de dados
        final String user = "usuario"; // aqui vai o nome usuario que vc quer acessar
        final String password = "123"; // aqui a senha do seu banco

        int id = 12;
        String autor = "ddgfdgdg";
        String query = "INSERT INTO autor(id, name) VALUES(?, ?)";
        Time time_i=getTime()
        try (Connection con = DriverManager.getConnection(url, user, password);
            PreparedStatement pst = con.prepareStatement(query)) {
            con.setAutoCommit(false);
            pst.setInt(1, id);
            pst.setString(2, autor);
            pst.executeUpdate();
            con.commit();
            System.out.println("Transação efetuada com sucesso");

        } catch (SQLException ex) {
            System.out.println("Transação não efetuada");
            Logger lgr = Logger.getLogger(Teste.class.getName());
            lgr.log(Level.SEVERE, ex.getMessage(), ex);
        }
        Time time_e=getTime()

    }
}
```

O principal ponto a ser notado é que foi acionado o comando `setAutoCommit(false)` para desabilitar a operação de auto commit. Algumas linguagens implementam isso de forma automática. Após executar operações de inserção, é acionado o comando de "salvamento"

para finalizar a operação.

#### **Detalhes que devem ser atendidas:**

- 1- Inserir as tuplas em uma tabela usando o Postgres, os dados estão no [arquivo](#)
- 2- A linguagem a ser utilizada por cada grupo será definida via sorteio;
- 3- Cada grupo irá elaborar um vídeo (**máximo de 5min**) com a explicação. Cada grupo deverá estar presente em aula para responder as dúvidas. O grupo deve participar do vídeo como um todo.

#### **Funções a serem implementadas:**

1- Fazer a inserção dos dados na tabela utilizando uma função com transações implícitas vs. transações explícitas (comando insert a cada nova inserção). Compare o tempo de inserção. Explique o motivo da variação do tempo.

**PS: rodar 5 vezes e fazer a média e desvio padrão dos tempos de execução para evitar perturbações de outros processos.**

2- Implementar uma função que cause um rollback na transação (ex: um campo mal formado). Alguma tupla foi salva no caso da implícita e da explícita?

As linguagens que serão abordadas neste trabalho são:

	Linguagem	Grupo (ID)
1	C++	19,9
2	Javascript	18,8
3	Rubby-- Shell Script	17,7
4	Go	16,6
5	PHP	15,5
6	C	14,4
7	Rust	13,3
8	Clojure	12,2

9	Dart	11,1
10	C#	10,20
11	Perl	21

### Grupos (criar grupos de 3 pessoas)

ID	NOME1	NOME2	NOME3
1	Murilo Xavier	José Timm	Rodrigo Vogt
3	Linka Sofia Lunkes	Patrícia Trevisan	Vitor Hugo Bittencourt dos Santos
5	Emerson andre Silvestrin	Bianca Gabriela Fritsch	Rafaelle
7	Jeferson Krumenauer	Pietro de Souza gonçalves	Vinicius Luis Tedesco
9	Romane		Paulo Comasetto
11	Douglas Kosvoski	Guilherme Rafael Graeff	Stefani Caroline Frare Meneghetti
12	Diego Minichiello	Eduardo kreve	
14	Jovenei Lorenzi	Bárbara	Adalberto Marques da Costa
15	PIERRE		
16	Henrique Alberto de Andrade	Vinícius Mittmann	Pedro manfio
17	kevin w. gonçalves	Milton	
18	Débora R.		
19	Nicolas	Mateus lima	
20	Guilherme Smorcinski Nerling		
21	Geovano		