

## **Técnicas de Programação – Prof. Fischer**

Trabalho Final – Primeira Parte

Nome: Johnatas Félix Zuza

Matrícula: 422160

Código:

```
package apresentacao;
```

```
import negocio.Categorias;
```

```
import negocio.Participante;
```

```
import persistencia.CategoriasDAO;
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        // Insere uma nova Categoria
```

```
        Participante p1 = new Participante(1, "", 10);
```

```
        Categorias p2 = new Categorias();
```

```
        CategoriasDAO dao = new CategoriasDAO();
```

```
        p2.setPessoa("p1");
```

```
        p2.setCarro("");
```

```
        p2.setCidadeEstadoPais("");
```

```
        p2.setFruta("");
```

```
        p2.setAnimal("");
```

```
        p2.setObjeto("");
```

```
        p2.setParticipante(p1);
```

```
        dao.inserir(p2);
```

```

        // Atualiza
        dao.update(p2, 9);

        // Deletar
        dao.deletar(14);

//          Listar Categorias/Recuperar
        for (Categorias p : dao.listar()) {
            System.out.println("pessoa: " + p.getPessoa() + " carro: " +
p.getCarro() + " cidadeEstadoPais: "
                                + p.getCidadeEstadoPais() + "fruta: " +
p.getFruta() + "objeto: " + p.getObjeto() + "animal: "
                                + p.getAnimal() + "id participante: " +
p.getParticipante());
        }

    }

}

package negocio;

public class Categorias {

    private Participante participante;
    private String pessoa;
    private String carro;
    private String cidadeEstadoPais;
    private String fruta;
    private String objeto;
    private String animal;

    public Categorias(Participante p) {
        setParticipante(p);
    }

    public Categorias() {

    }

    public String getPessoa() {
        return pessoa;
    }
}

```

```

    public void setPessoa(String pessoa) {
        this.pessoa = pessoa;
    }

    public String getCarro() {
        return carro;
    }

    public void setCarro(String carro) {
        this.carro = carro.toUpperCase();
    }

    public String getCidadeEstadoPais() {
        return cidadeEstadoPais;
    }

    public void setCidadeEstadoPais(String cidadeEstadoPais) {
        this.cidadeEstadoPais = cidadeEstadoPais.toUpperCase();
    }

    public String getFruta() {
        return fruta;
    }

    public void setFruta(String fruta) {
        this.fruta = fruta.toUpperCase();
    }

    public String getObjeto() {
        return objeto;
    }

    public void setObjeto(String objeto) {
        this.objeto = objeto.toUpperCase();
    }

    public String getAnimal() {
        return animal;
    }

    public void setAnimal(String animal) {
        this.animal = animal.toUpperCase();
    }

    public Participante getParticipante() {
        return participante;
    }

    public void setParticipante(Participante participante) {
        this.participante = participante;
    }

    public int getId() {
        return participante.getId();
    }
}

package negocio;

```

```

public class Participante {

    private int id;
    private String nome;
    private int pontuacao;

    public Participante(int id, String nome, int pontuacao) {
        setId(id);
        setNome(nome);
        setPontuacao(pontuacao);
    }

    public Participante() {

    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public int getPontuacao() {
        return pontuacao;
    }

    public void setPontuacao(int pontuacao) {
        this.pontuacao = pontuacao;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

}

```

```

package negocio;

```

```

public class Participante {

    private int id;
    private String nome;
    private int pontuacao;

    public Participante(int id, String nome, int pontuacao) {
        setId(id);
        setNome(nome);
        setPontuacao(pontuacao);
    }

    public Participante() {

    }

}

```

```

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public int getPontuacao() {
        return pontuacao;
    }

    public void setPontuacao(int pontuacao) {
        this.pontuacao = pontuacao;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
}

package persistencia;

import java.sql.*;

public class Conexao {
    private String host;
    private String usuario;
    private String senha;
    private String banco;

    public Conexao() {
        this.host = "127.0.0.1";
        this.banco = "da_java";
        this.usuario = "root";
        this.senha = "";
    }

    public Connection getConexao() {
        String url = "jdbc:mysql://" + this.host + "/" + this.banco;
        try {
            return DriverManager.getConnection(url, usuario, senha);
        } catch (SQLException ex) {
            System.out.println("Conexão com MySQL não realizada");
            ex.printStackTrace();
            //
            Logger.getLogger(Conexao.class.getName()).log(Level.SEVERE, null, ex);
        }
        return null;
    }
}

package persistencia;

```

```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import negocio.Participante;

public class ParticipanteDAO {

    public Participante listarPorId(int id) {

        Participante participante = new Participante();

        try {

            Connection conexao = new Conexao().getConexao();

            PreparedStatement sql =
conexao.prepareStatement("select * from participante where id=?");

            sql.setInt(1, id);

            ResultSet result = sql.executeQuery();

            while (result.next()) {
                participante.setNome(result.getString("nome"));

                participante.setPontuacao(result.getInt("pontuacao"));
            }

            conexao.close();
```

```
    } catch (Exception e) {  
    }  
    return participante;  
}
```

```
public ArrayList<Participante> listar() {
```

```
    ArrayList<Participante> listaParticipantes = new  
    ArrayList<Participante>();
```

```
    try {
```

```
        Connection conexao = new Conexao().getConexao();
```

```
        ResultSet result = conexao
```

```
            .prepareStatement("SELECT * FROM  
participante order by dataCadastroParticipante DESC")
```

```
            .executeQuery();
```

```
        Participante participante;
```

```
        while (result.next()) {
```

```
            participante = new Participante();
```

```
            participante.setNome(result.getString("nome"));
```

```
            participante.setPontuacao(result.getInt("pontuacao"));
```

```
            listaParticipantes.add(participante);
```

```
        }
```

```
        conexao.close();
```

```
    } catch (SQLException e) {
```

```

    } finally {
    }
    return listaParticipantes;
}

```

```

public boolean inserir(Participante a) {
    int valor = 0;
    try {
        Connection conexao = new Conexao().getConexao();

        PreparedStatement result = conexao
            .prepareStatement("INSERT INTO
participante (nome, pontuacao) VALUES (?, ?);");

        result.setString(1, a.getNome());
        result.setInt(2, a.getPontuacao());

        valor = result.executeUpdate();
        conexao.close();
    } catch (Exception e) {

    }

    if (valor == 1)
        return true;
    else
        return false;
}

```

```

public void deletar(int id) {

```



```

try {

    Connection conexao = new Conexao().getConexao();

    PreparedStatement result =
conexao.prepareStatement("DELETE FROM participante WHERE id = ?;");

    result.setInt(1, id);

    result.executeUpdate();

    conexao.close();
} catch (Exception e) {

}

}

```

```

public boolean update(Participante a) {
    int valor = 0;
    try {
        Connection conexao = new Conexao().getConexao();

        PreparedStatement result = conexao
            .prepareStatement("UPDATE participante
SET nome = ?, pontuacao=? WHERE id = ?");

        result.setString(1, a.getNome());
        result.setInt(2, a.getPontuacao());
        result.setInt(3, a.getId());

        valor = result.executeUpdate();
    }
}

```

```
        conexao.close();
    } catch (Exception e) {

    }

    if (valor == 1)
        return true;
    else
        return false;

}

}
```