



PADRÃO DAO

FAPESC – DESENVOLVEDORES PARA TECNOLOGIA DA INFORMAÇÃO

HERCULANO DE BIASI

herculano.debiasi@unoesc.edu.br

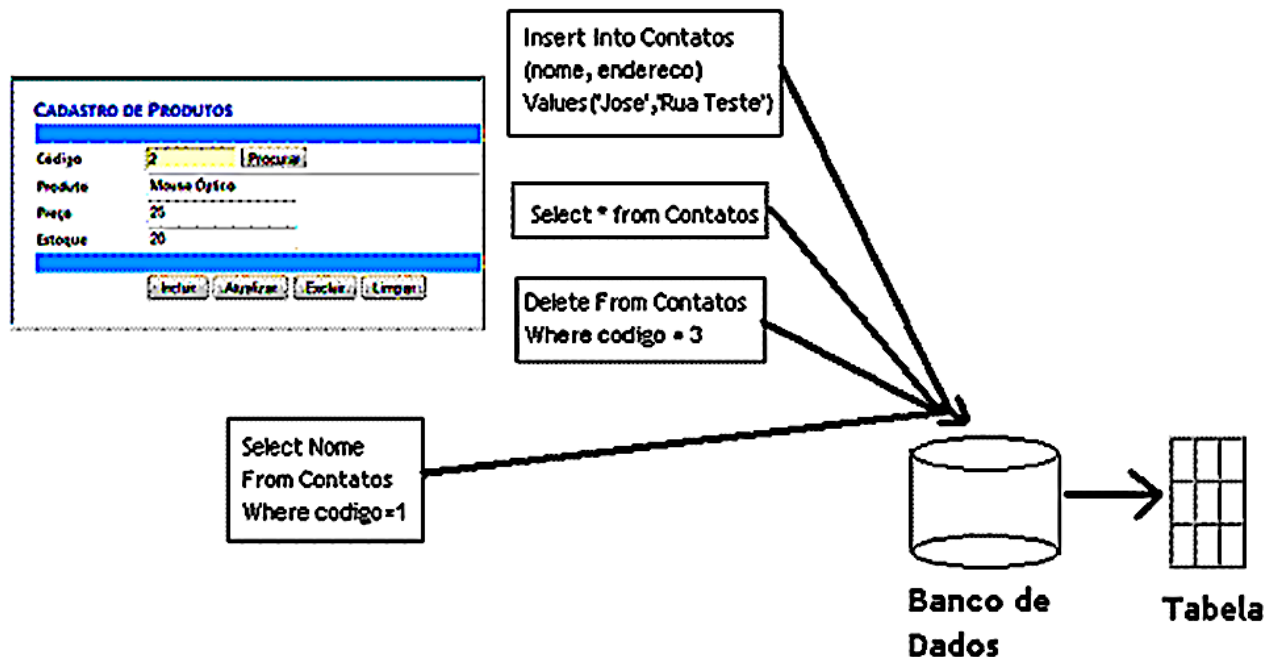


TÓPICOS

- Padrão DAO
- JavaBeans

PADRÃO DAO

- **DAO** significa *Data Access Object* (Objeto de Acesso a Dados) e representa um padrão de projeto destinado a realizar a persistência de dados fornecendo uma *interface* abstrata sem precisar expor detalhes do banco
- Este isolamento suporta o **Princípio de Responsabilidade Única** do **SOLID**
- Este padrão se originou na Sun, com o Java EE “Sun Microsystems’ best practice guidelines ‘**Core J2EE Patterns**’”
- DAO reduziu o problema de acoplamento entre GUI e BD, então muito comum



PADRÃO DAO

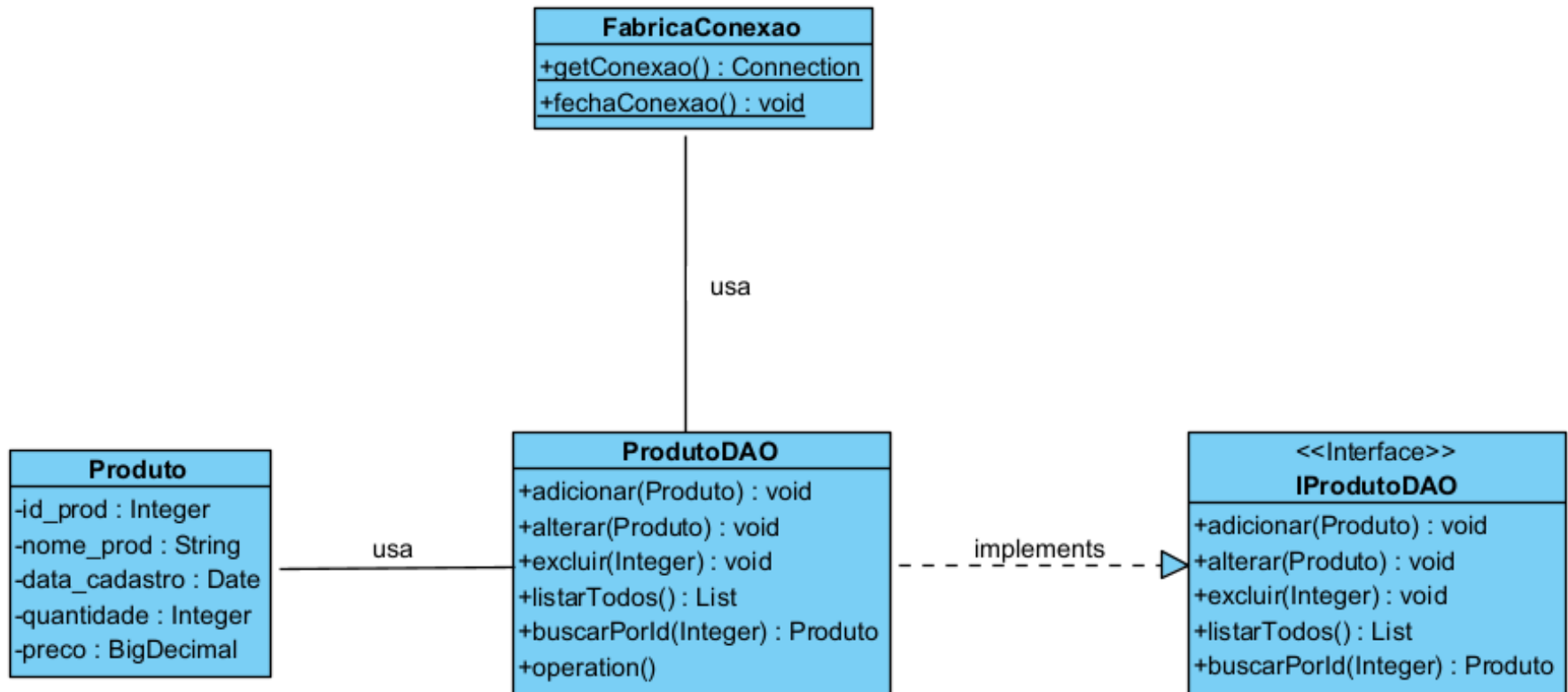
- O padrão DAO permite separar a lógicas das regras de negócio das regras de acesso a banco de dados, fornecendo algumas operações específicas sobre os dados sem expor os detalhes do sistema gerenciador de banco de dados
- Esse desacoplamento diminui a dependência das entidades de domínio em relação à tecnologia ou implementação específica de persistência dos dados, ou seja, diminui o impacto de mudar, por exemplo, de um SGBD MySQL para um Oracle, ou de armazenamento baseado em SGBD para um NoSQL ou *web service*
- DAO atua como um intermediário entre a aplicação e o banco de dados sendo responsável por todo o acesso a dados de forma a manter o encapsulamento
- Cada instância DAO é responsável por um objeto de domínio e responsável pelas operações de CRUD no domínio
- A classe DAO não deve ser a responsável por gerenciar as transações e sessões ou criar as conexões – isso deve ser feito fora do DAO

PADRÃO DAO

- Em uma aplicação que utilize a arquitetura MVC, todas as funcionalidades de bancos de dados, tais como obter as conexões, mapear objetos Java para tipos de dados SQL ou executar comandos SQL, devem ser feitas por classes DAO
- Métodos comuns em uma classe DAO
 - Adicionar
 - Alterar
 - Remover
 - Listar todos os registros
 - Buscar um registro específico pelo ID

PADRÃO DAO

- Em relação à implementação do padrão DAO pode-se encapsular operações CRUD em uma *interface* que deve ser implementada por uma classe concreta
- Por fim, a implementação concreta do DAO usa APIs tais como JDBC, ADO, JPA e Hibernate, para executar operações CRUD



JAVABEANS

- JavaBeans são classes que possuem o construtor simples, sem argumentos, e expõem suas propriedades através de métodos de acesso do tipo `get` e `set`
- JavaBeans precisam implementar a *interface* `java.io.Serializable`
- Beans podem ser utilizados em diversas situações, como em classes que representam o modelo de dados
- JavaBeans não devem ser confundidos com os Enterprise Java Beans (EJB)

```
public class Contato implements java.io.Serializable {  
    private Long id;  
    private String nome;  
    private String endereco;  
    private String telefone;  
    private String email;  
    private String genero;  
  
    public Contato() { }  
  
    public Long getId() { return id; }  
    public void setId(Long id) { this.id = id; }  
  
    public String getNome() { return nome; }  
    public void setNome(String nome) { this.nome = nome; }  
  
    public String getEndereco() { return endereco; }  
    public void setEndereco(String endereco) { this.endereco = endereco; }  
  
    public String getTelefone() { return telefone; }  
    public void setTelefone(String telefone) { this.telefone = telefone; }  
  
    public String getEmail() { return email; }  
    public void setEmail(String email) { this.email = email; }  
  
    public String getGenero() { return genero; }  
    public void setGenero(String genero) { this.genero = genero; }  
}
```