

Java Persistence API e Frameworks ORM

Mapeamento Objeto Relacional ou Object-Relational Mapping

Object-Relational Mapping (ORM) é uma técnica de programação para converter dados entre bancos de dados relacionais e linguagens de programação orientada a objeto, como Java, C #, etc.

Vantagens

- 1- Vamos comercializar objetos de acesso ao código em vez de tabelas de banco de dados.
- 2 Oculta detalhes de consultas SQL da lógica OO.
- 3 Baseado no JDBC.
- 4 Não há necessidade de lidar com a implementação do banco de dados.
- 5 Entidades baseadas em conceitos de negócios em vez de estrutura de banco de dados.
- 6 Gerenciamento de transações e geração automática de chaves.
- 7 Desenvolvimento rápido da aplicação.

Regras para mapeamento - ORM

- Todas as tabelas (ou relações) devem ter uma chave primária
- Mapeamento de atributos
- Herança
- Associações
- Mapeamento

ORM - Tabelas

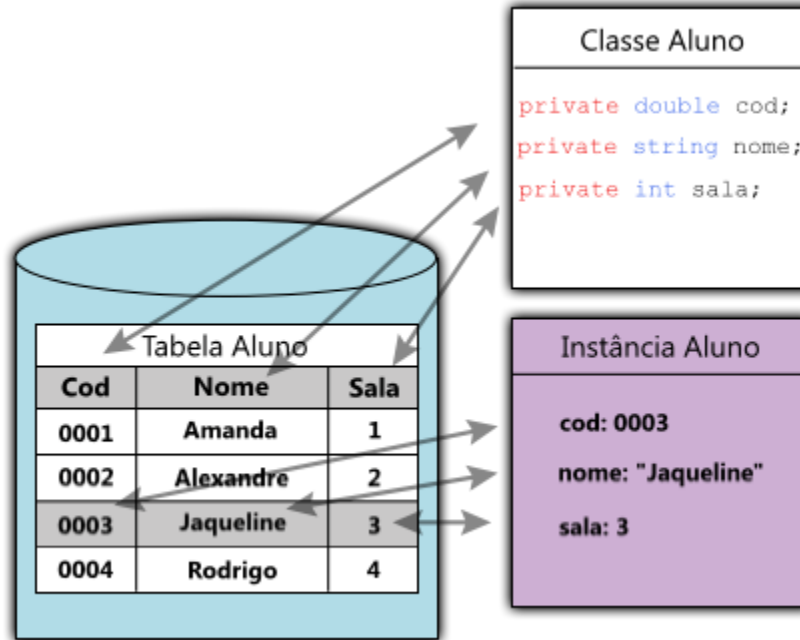
Em um sistema orientado a objetos, cada objeto é único. Essa unicidade é garantida através da introdução de um “identificador de objetos” (OID – Object IDentifier).

ORM - Mapeamento de Atributos

Existem três tipos de mapeamento de atributos:

- Atributos simples são mapeados através de colunas;
- Atributos compostos podem ser mapeados em várias colunas;
- Atributos multivalorados devem ser mapeados em tabelas.

ORM - Mapeamento de Atributos

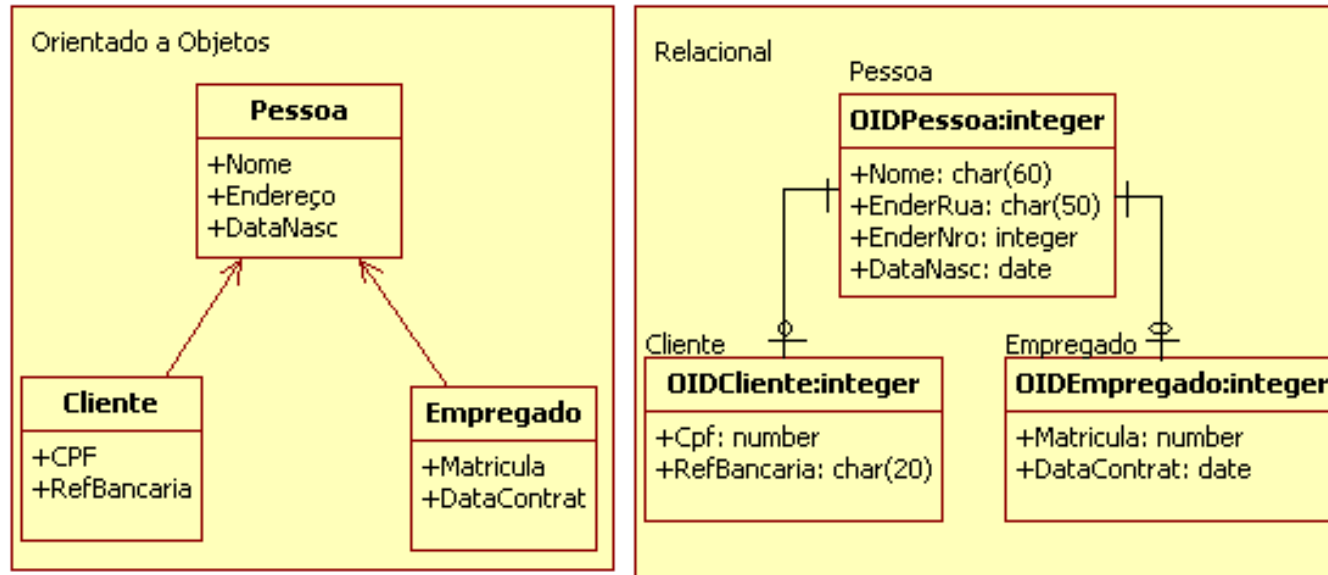


ORM - Herança

Existem três tipos de mapeamento de herança:

- Criar uma tabela para cada classe;
- Criar uma única tabela para toda a hierarquia de classes;
- Criar uma tabela para cada classe concreta.

ORM - Herança



Mapeamento de uma tabela por classe

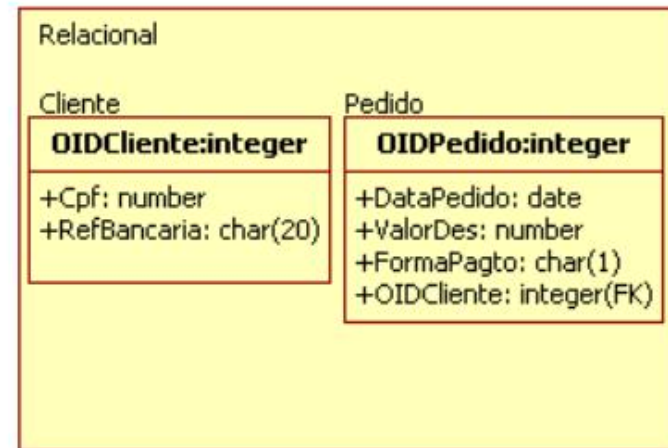
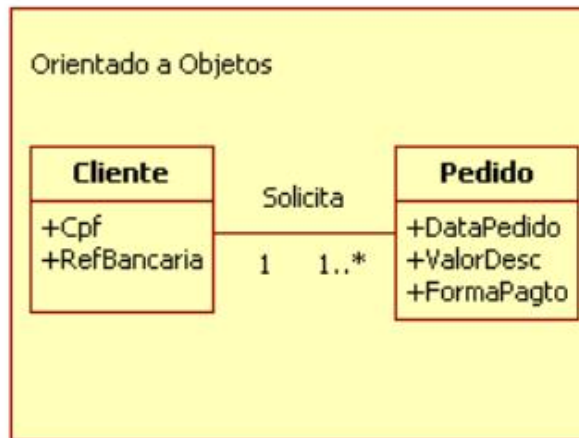
ORM - Associação

- **Associação Muitos-para-Muitos:** deve-se criar uma tabela associativa em que a chave primária é composta pelas chaves primárias das tabelas associadas ou tabela associativa.



ORM - Associação

- **Associação Um-para-Muitos:** a tabela cujos registros podem ser endereçados diversas vezes é a que herda a referência da tabela cuja correspondência é unitária



ORM - Associação

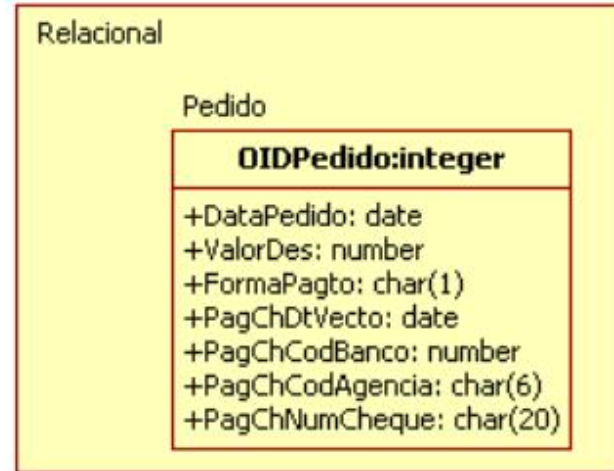
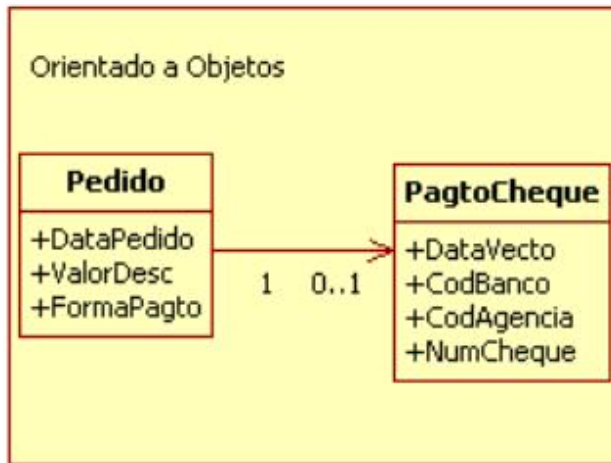
- **Associação Um-para-Um**

- **Existem dois tipos:**

- **Única tabela no modelo relacional:** os atributos da classe agregada devem ser colocados na mesma tabela da classe agregadora
 - **Gerar duas tabelas:** uma delas deve herdar como um atributo normal (chave estrangeira) a chave primária da outra tabela

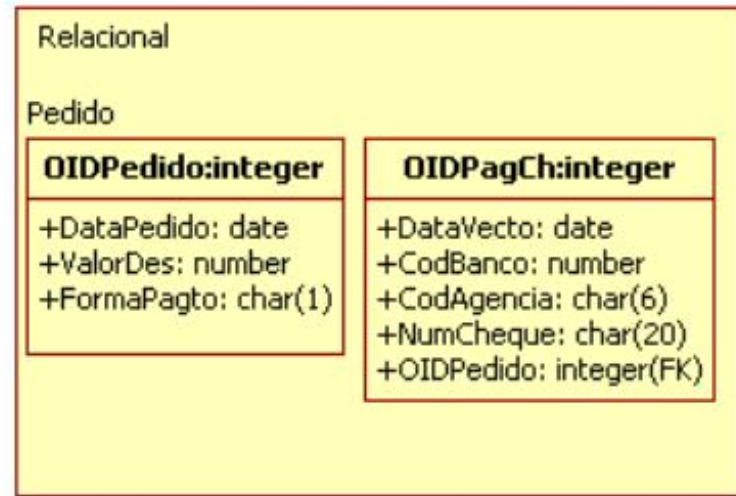
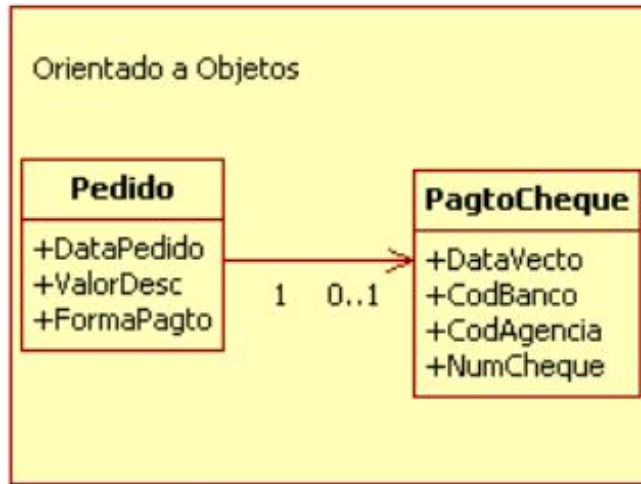
ORM - Associação

- Associação Um-para-Um
 - Única tabela no modelo relacional



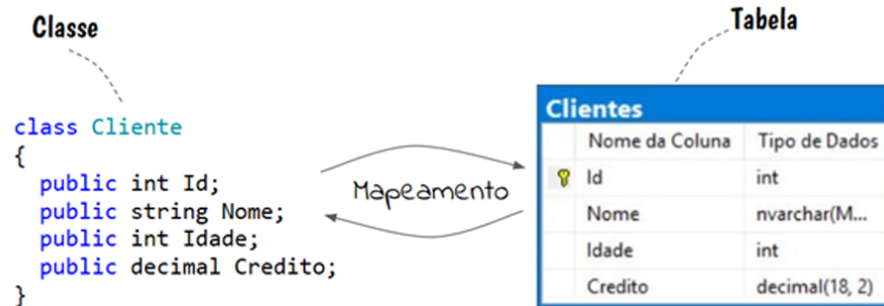
ORM - Associação

- Associação Um-para-Um
 - Gerar duas tabelas



ORM - Mapeamento

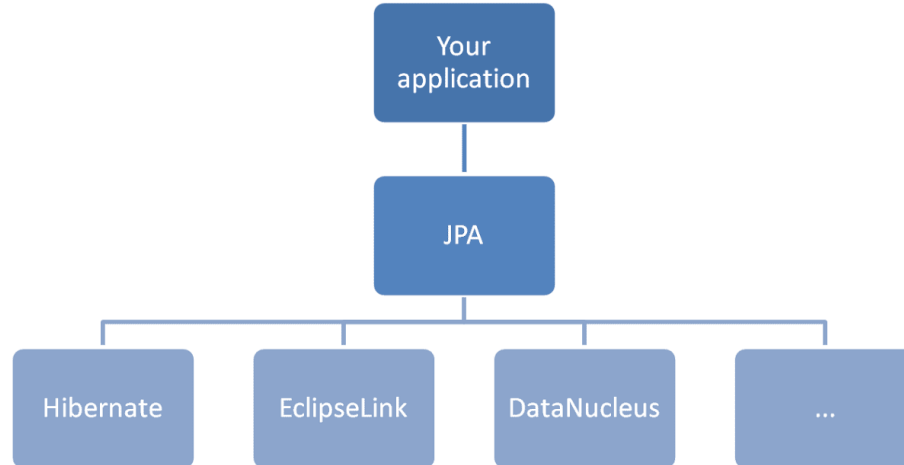
É a representação do mapeamento de um sistema



Java Persistence API

JPA - Java Persistence API

JPA (ou Java Persistence API) é uma especificação oficial que descreve como deve ser o comportamento dos frameworks de persistência Java que desejarem implementá-la



JPA - Algumas características

- POJOS Persistentes
- Consultas em Objetos
- Configurações simples
- Integração e testes

JPA - Exemplo

```
17 @Entity
18 @Table(name="VEICULOS")
19 public class Veiculo {
20
21     @Id
22     @GeneratedValue
23     private Long id;
24
25     @Column(nullable=false)
26     private Integer ano;
27
28     @Column(nullable=false)
29     private String modelo;
30
31     @Lob
32     private byte[] foto;
33
34     @Enumerated
35     @Column(nullable = false)
36     private Cor cor;
37
38     @Enumerated
39     @Column(nullable = false)
40     private Marca marca;
41
42     @OneToMany(fetch=FetchType.EAGER, mappedBy = "veiculo")
43     private Set<Locacao> locacoes;
44
45     @ManyToOne(fetch = FetchType.EAGER)
46     @JoinColumn(name="PRD_ID", nullable = false)
47     private Proprietario proprietario;
48
49     // getters e setters omitidos
50 }
```

Frameworks ORM em Java

- Enterprise JavaBeans Entity Beans
- Java Data Objects
- Castor
- TopLink
- Spring DAO
- Hibernate
- E muito mais...

Demonstração: usando o JPA

Exercícios

