

HIBERNATE: Mapeamento de Relacionamento N..N

Jean Eduardo Glazar

Programação Orientada a Objetos II

*Curso de Sistemas de Informação
Ifes Campus Colatina*

Tabela intermediária SEM atributos

@ManyToMany

- Nas duas classes
- FetchType.LAZY

@JoinTable

- **name**: nome da tabela intermediária
- **joinColumns**: ID da tabela corrente
- **inverseJoinColumns**: ID da outra tabela



Classe Pedido

```
@Entity
public class Pedido implements Serializable {
    ...
    @ManyToMany ( fetch = FetchType.LAZY )
    @JoinTable ( name="Pedido_Lanche",
                joinColumns = { @JoinColumn (name="idPedido") },
                inverseJoinColumns = { @JoinColumn(name="idLanche") }
    )
    private List<Lanche> lanches = new ArrayList();
    ...
}
```

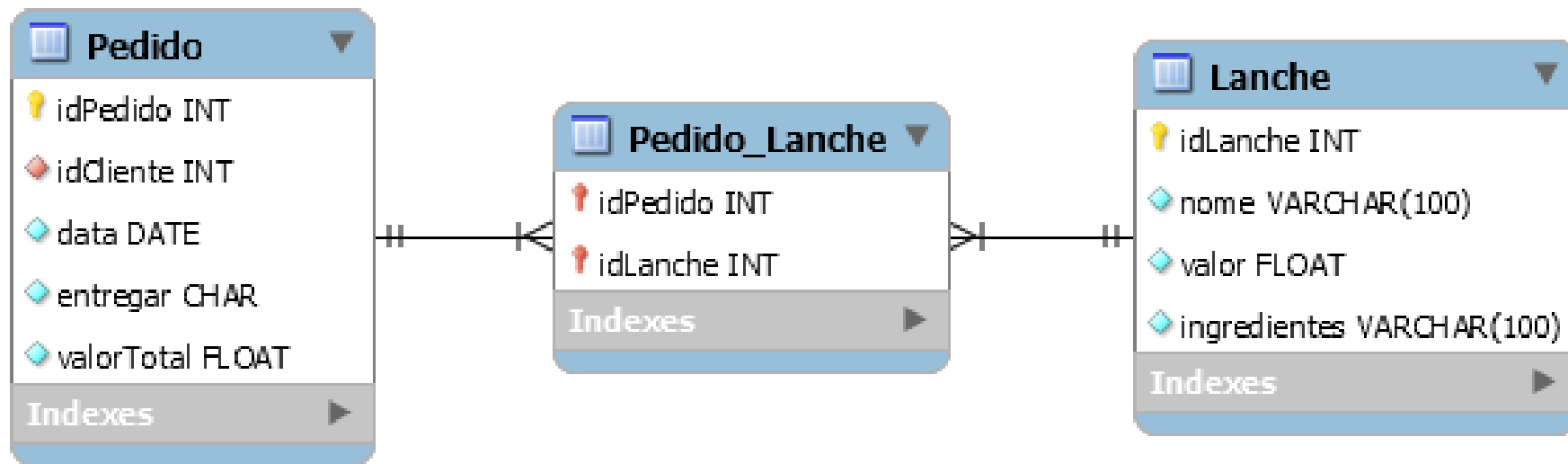


Classe Lanche

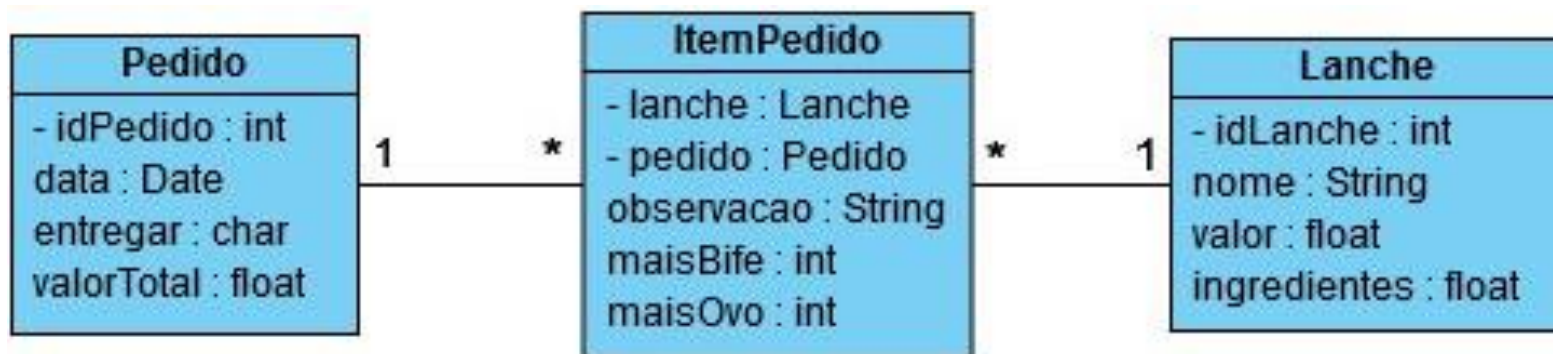
```
@Entity
public class Lanche implements Serializable {
    ...
    @ManyToMany ( fetch = FetchType.LAZY )
    @JoinTable ( name="Pedido_Lanche",
                joinColumns = { @JoinColumn (name="idLanche") },
                inverseJoinColumns = { @JoinColumn(name="idPedido") }
            )
    private List<Pedido> pedidos = new ArrayList();
    ...
}
```



Tabela intermediária SEM atributos

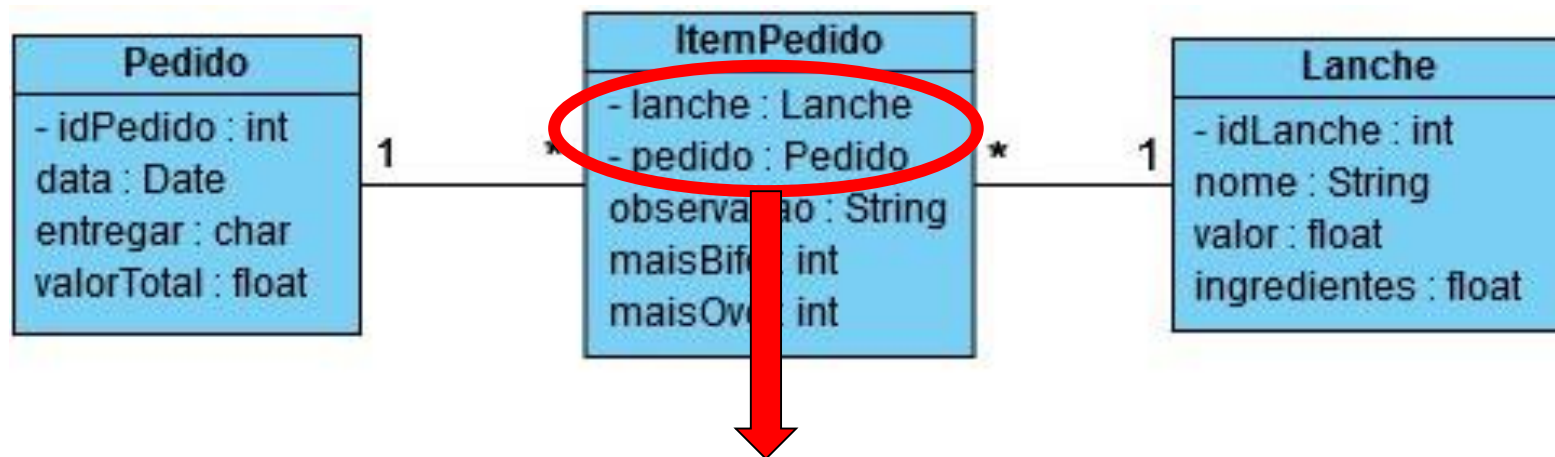


Classe intermediária COM atributos



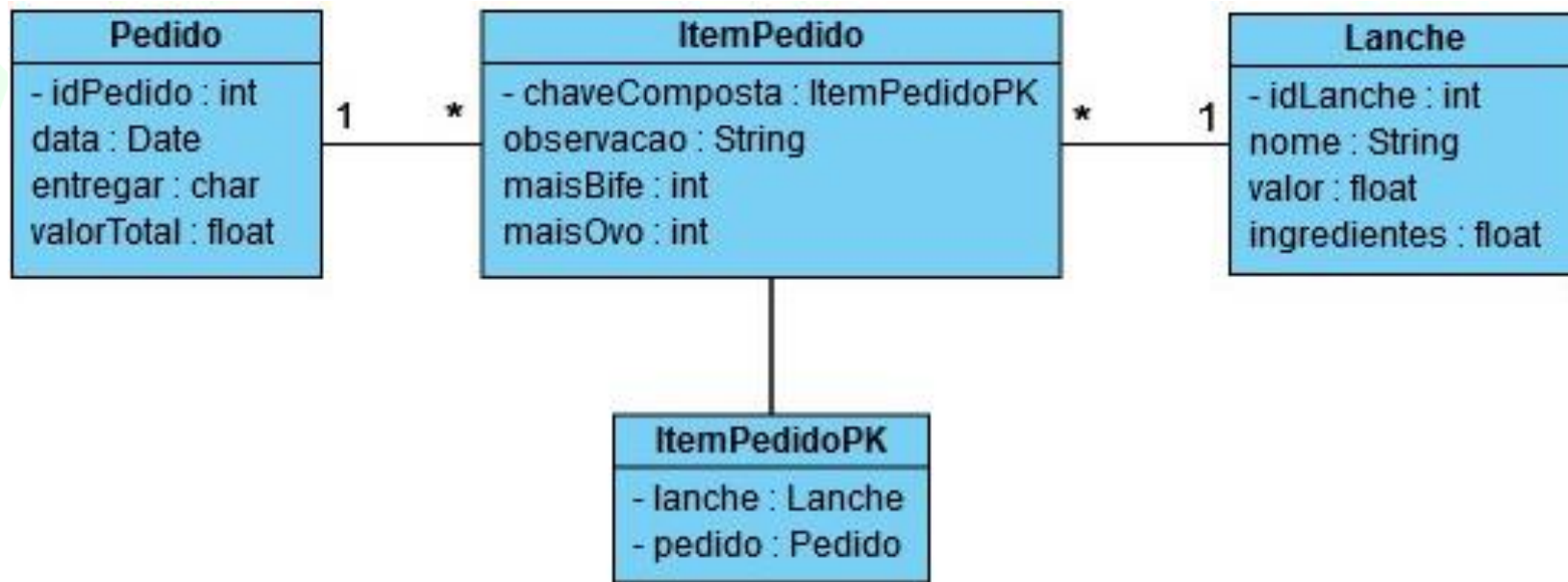
Mapeamento de Relacionamentos 1..N

Chave primária composta



Outra classe:
ItemPedidoPK

Chave primária composta



Chave primária composta

@Embeddeld

- No atributo composto

@Embeddable

- Na classe que representa a chave composta



Chave primária composta

@Embeddable

```
public class ItemPedidoPK implements Serializable{
```

```
...
```

```
@ManyToOne ( fetch = FetchType.EAGER )
```

```
@JoinColumn ( name="idLanche" )
```

```
private Lanche lanche;
```

```
@ManyToOne ( fetch = FetchType.EAGER )
```

```
@JoinColumn ( name="idPedido" )
```

```
private Pedido pedido;
```

```
...
```

```
}
```

@Entity

```
public class ItemPedido implements Serializable {
```

```
@EmbeddedId
```

```
private ItemPedidoPK chComposta;
```

```
...
```

```
}
```



Chave primária composta

```
@Entity
public class Lanche implements Serializable {
    ...
    @OneToMany (mappedBy = "chComposta.lanche",
                fetch = FetchType.LAZY )
    private List<ItemPedido> itensPedido;
    ...
}
```

```
@Entity
public class Pedido implements Serializable {
    ...
    @OneToMany (mappedBy="chComposta.pedido",
                fetch = FetchType.LAZY )
    private List<ItemPedido> itensPedido;
    ...
}
```



Tabela intermediária COM atributos

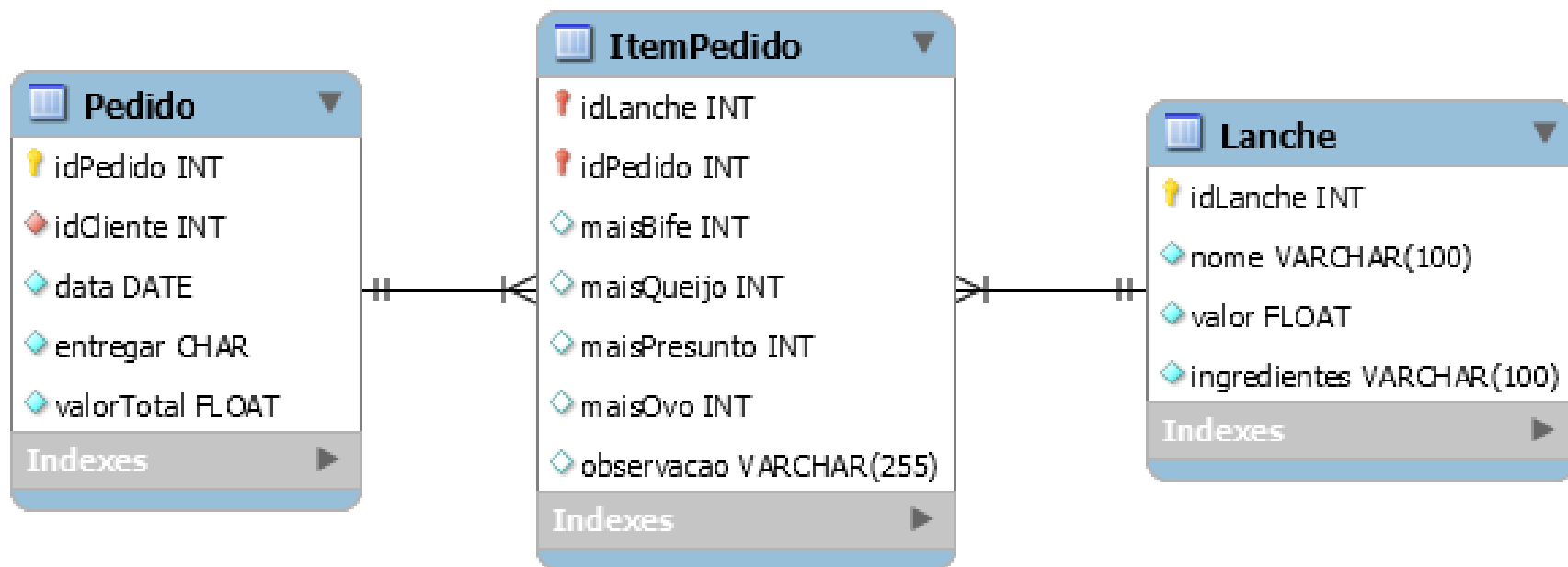
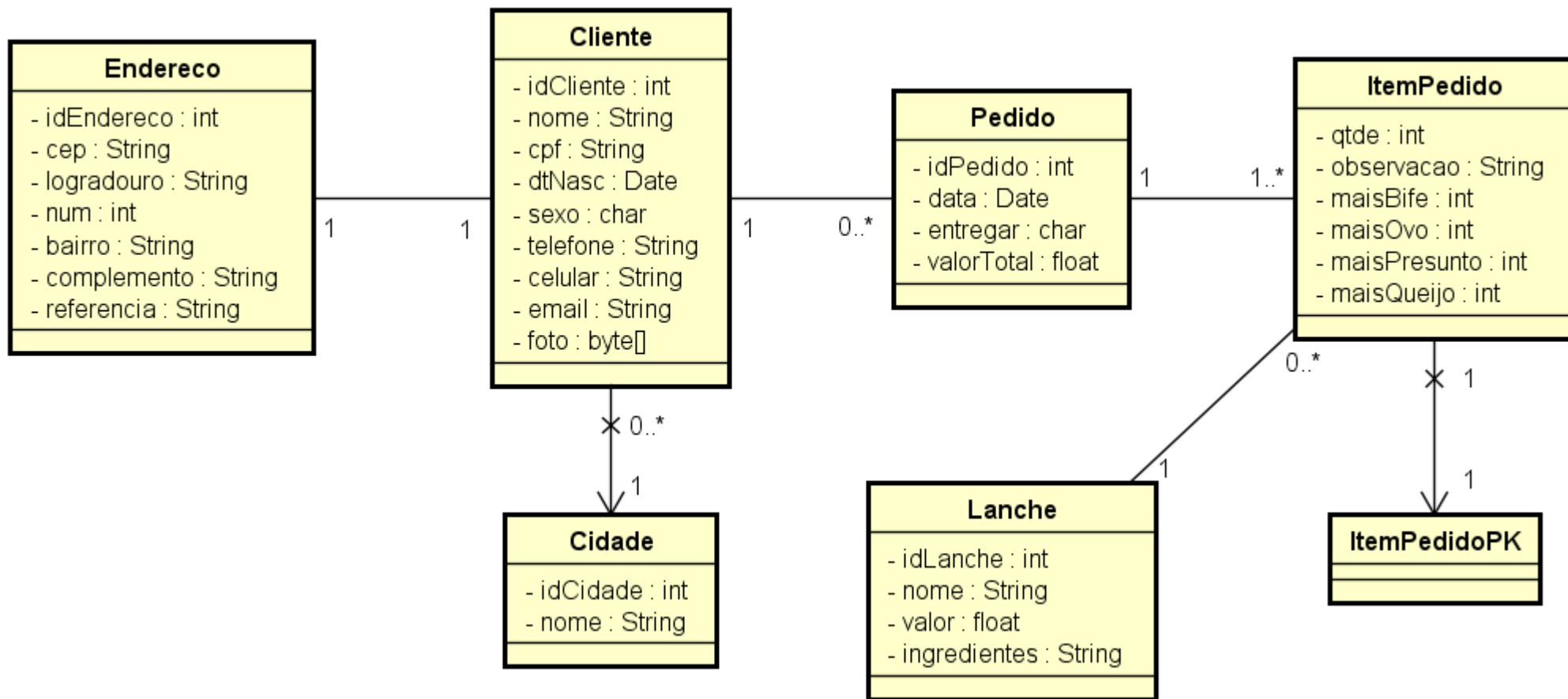


Diagrama de Classes



Banco de Dados Criado

