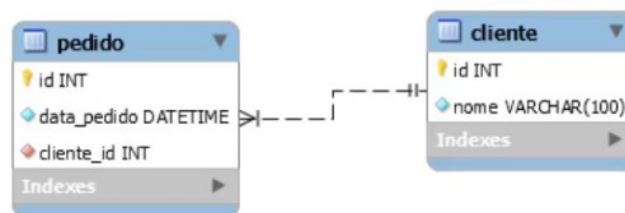


Relacionamentos

- Muitos para um
- Um para muitos
- Um para um
- Muitos para muitos

Muitos para um

- Representada pela anotação @ManyToOne
- É quando temos a possibilidade de ter vários registros, de uma tabela qualquer, fazendo referência para um registro específico que pode ser da mesma tabela ou, como na maioria das vezes, uma tabela diferente.



```
public class Pedido {

    @Id
    private Integer id;

    @ManyToOne
    @JoinColumn(name = "cliente_id")
    private Cliente cliente;

    // Outros atr. omitidos.
}
```

```
public class Cliente {

    @Id
    private Integer id;

    private String nome;

    // Outros atr. omitidos.
}
```

Um para muitos

- Representada pela anotação `@OneToMany`
- É quando temos a possibilidade de um registro específico ser referenciado por vários outros. Basicamente, o contrário do que foi descrito para `@ManyToOne`.
- Só que agora estamos falando de coleções

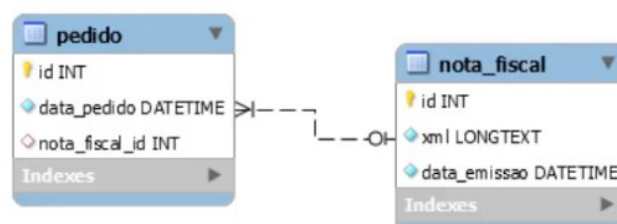


```
public class Pedido {  
  
    @Id  
    private Integer id;  
  
    @ManyToOne  
    @JoinColumn(name = "cliente_id")  
    private Cliente cliente;  
  
    // Outros atr. omitidos.  
}
```

```
public class Cliente {  
  
    @Id  
    private Integer id;  
  
    private String nome;  
  
    @OneToMany(mappedBy = "cliente")  
    private List<Pedido> pedidos;  
  
    // Outros atr. omitidos.  
}
```

Um para um

- Representada pela anotação `@OneToOne`
- É quando uma tabela restringe seus registros a não fazerem referências repetidas. Uma mesma referência não pode ser feita duas vezes.



```
public class Pedido {

    @Id
    private Integer id;

    @OneToOne
    @JoinColumn(name = "nota_fiscal_id")
    private NotaFiscal notaFiscal;

    // Outros atr. omitidos.
}
```

```
public class NotaFiscal {

    @Id
    private Integer id;

    private String xml;

    @OneToOne(mappedBy = "notaFiscal")
    private Pedido pedido;

    // Outros atr. omitidos.
}
```

Muitos para muitos

- Representada pela anotação @ManyToMany
- É quando temos a possibilidade de ter um registro referenciando vários outros ao mesmo tempo que pode ser referenciado várias vezes.
- Como estamos falando de múltiplas referências de e para um mesmo registro, lidamos também com coleções.



```
public class Produto {

    @Id
    private Integer id;

    @ManyToMany
    @JoinTable(name = "produto_categoria",
        joinColumns = @JoinColumn(name = "produto_id"),
        inverseJoinColumns = @JoinColumn(name = "categoria_id"))
    private List<Categoria> categorias;

    // Outros atr. omitidos.
}
```

```
public class Categoria {

    @Id
    private Integer id;

    @ManyToMany(mappedBy = "categorias")
    private List<Produto> produtos;

    // Outros atr. omitidos.
}
```