






02. Consultas avançadas

 Date	
 Categoria	Java e persistência
 Curso	Java e JPA: consultas avançadas performance e modelos complexos

Tópicos

- Projeto da aula anterior
- Consultas com funções de agregação
- Consultas para relatórios
- Consultas com select new
- Gerando relatórios
- Utilizando Named Queries
- Vantagens das named queries
- Faça como eu fiz
- O que aprendemos?

Gerando relatórios

Considere a seguinte consulta JPQL:

```
select new br.com.alura.loja.dto.VendasPorDia(produto.nome, sum(pedido.valorTotal), pedido.data) from Pedido pedido join pedido.itens itens join itens.produto produto group by produto.nome, pedido.data
```

E a seguinte classe DTO:

```
public class VendasPorDia {
    private String nomeDoProduto;
    private BigDecimal valorVendido;
    private LocalDate dataDasVendas;

    //getters e setter de todos os atributos
}
```

Por qual motivo a consulta anterior não vai funcionar corretamente?

- A classe DTO deve conter um construtor compatível com a consulta JPQL.
- A JPA cria instâncias da classe DTO via construtor que recebe parâmetros, conforme a consulta JPQL

Vantagens das named queries

Qual das seguintes alternativas é uma das vantagens de se utilizar named queries?

- Deixar as consultas de uma entidade mais próximas da classe dela
- named queries são declaradas na classe da entidade

O que aprendemos?

Nessa aula, você aprendeu;

- A realizar consultas utilizando funções de agregação, tais como `min`, `max`, `avg` e `sum`;
- Como escrever consultas de *relatórios* utilizando JPQL
- Como utilizar o recurso `select new` em consultas JPQL;
- A realizar consultas utilizando *named queries*.