



Funções de agregação (**MIN, MAX, SUM, AVG**) em consultas também são suportadas pelo **JPA**, basta passar o **atributo** da **Entidade** que você quer utilizar:

```
public BigDecimal valorTotalDePedidos(){
    String jpql = "SELECT SUM(p.valorTotal) FROM Pedido p";
    return entityManager.createQuery(jpql, BigDecimal.class).getSingleResult();
}
```

Na **esmagadora** maioria das vezes fazemos **consultas mais complexas** do que isso, por exemplo, se eu quiser um relatório desse tipo:

Produto	quantidade	ultima venda
Geladeira Gamer	12	17/02/2022
Nokia tijolo	6	15/03/2004

Eu preciso de um **SELECT** que me traga três **informações que estão em entidades diferentes**. O **nome do Produto** está na entidade de **Produto**, a **quantidade** está no **itemProduto** e a **última venda** está no **Pedido**, eu preciso fazer **JOINS** entre essas entidades

Além disso eu estou **AGRUPANDO** e **ORDENANDO** elas pela quantidade de venda **DECRESCENTE**:

```
StringBuilder jpql = new StringBuilder( capacity: 300);
jpql.append("SELECT produto.nome, ")
    .append("SUM(item.quantidade), ")
    .append("MAX(pedido.dataPedido) ")
    .append("FROM Pedido pedido ")
    .append("JOIN pedido.itens item ")
    .append("JOIN item.produto produto ")
    .append("GROUP BY produto.nome ")
    .append("ORDER BY item.quantidade DESC");
```

Eu **SELECIONEI** colunas específicas de entidades diferentes, então como eu deveria representar esse registro de retorno? Aí entra o conceito de **VO(Value Object/Objeto de valor)**, esse tipo de objeto é uma **classe** que representa algum valor, mas **ela não tem nenhum comportamento**, apenas atributos:

```
public class RelatorioDeVendasVo {

    private final String nome;
    private final Long quantidade;
    private final LocalDate ultimaVenda;
}
```

Essa **classe** não é uma Entidade, portanto temos que usar um recurso do JPA para explicitar isso, esse recurso é a palavra chave **“new”** seguido pelo nome da classe VO (*precisa ser o fully qualified name*). Dessa maneira **encapsulamos os atributos buscados dentro** dessa nova **classe**:

```
public List<RelatorioDeVendasVo> relatorioDeVendas() {  
  
    Class<RelatorioDeVendasVo> vo = RelatorioDeVendasVo.class;  
  
    StringBuilder jpql = new StringBuilder("capacity: 300");  
    jpql.append("SELECT new ")  
        .append(vo.getName() + "(")  
        .append("produto.nome,")  
        .append("SUM(item.quantidade), ")  
        .append("MAX(pedido.dataPedido)")  
        .append("FROM Pedido pedido ")  
        .append("JOIN pedido.itens item ")  
        .append("JOIN item.produto produto ")  
        .append("GROUP BY produto.nome, item.quantidade ")  
        .append("ORDER BY item.quantidade DESC");  
  
    return entityManager.createQuery(jpql.toString(), RelatorioDeVendasVo.class).getResultList();  
}
```

```
pedidoDao.relatorioDeVendas().forEach(System.out::println);
```

CadastroPedido X

order by

itens1.quantidade DESC

RelatorioDeVendasVo{nome do produto = 'Nokia', quantidade vendida = 10, ultima Venda = 2022-02-17}

RelatorioDeVendasVo{nome do produto = 'Razor trix', quantidade vendida = 7, ultima Venda = 2022-02-17}

RelatorioDeVendasVo{nome do produto = 'Geladeira gamer', quantidade vendida = 3, ultima Venda = 2022-02-17}

RelatorioDeVendasVo{nome do produto = 'Nokia', quantidade vendida = 1, ultima Venda = 2022-02-17}