

O problema do N + 1

Agora que deixamos o contexto de persistência aberto durante toda a requisição, podemos carregar um relacionamento *lazy* a partir do JSP. Podemos aproveitar para melhorar um pouco a usabilidade da nossa lista de produtos e já mostrar quais as categorias de um determinado produto. Para isso, vamos no arquivo `home.jsp` e acima dos botões `Editar` e `Saiba mais` adicionamos uma lista para exibirmos as categorias:

```
<c:forEach items="${produto.categorias}" var="categoria">
    <li>${categoria.nome}
</c:forEach>
```

O resultado final deve parecer com o seguinte:

The screenshot shows a list of three products on a page titled 'Produto'. Each product card includes the product name, a thumbnail image, a brief description, and a list of categories. Below each description are 'Saiba mais' and 'Editar' buttons.

- Curso de Violão**
musicdot
- Introdução a Arquitetura Java e Design de projetos com Java**
INTRODUÇÃO À ARQUITETURA E DESIGN DE SOFTWARE
- Vire o jogo com Spring Framework**
Vire o jogo com Spring Framework

Vamos verificar no console as "queries" que foram disparadas para popular as categorias dos produtos:

```
Hibernate: select produto0_.id as id1_2_, produto0_.descricao as descricao2_2_, produto0_.linkDa
Hibernate: select categorias0_.Produto_id as Produto_1_2_0_, categorias0_.categorias_id as cate
```

Ao todo, foram executados **6 selects**. Como assim, por quê? Vamos entender o que aconteceu.

O primeiro `select` se refere a lista de produtos cadastrados no banco. Nela, realizamos o primeiro `<c:forEach ...>` para exibir os produtos na tela. Mas como exibir as categorias? Já temos os produtos, para pegar as categorias basta realizar outro `<c:forEach ...>` com as categorias do produto. Ou seja, temos algo assim:

```
<c:forEach items="${produtos}" var="produto">
    ...
    <c:forEach items="${produto.categorias}" var="categoria">
        <li>${categoria.nome}
    </c:forEach>
```

```
...  
</c:forEach>
```

Para cada produto da minha lista, estamos buscando suas categorias. No nosso caso temos **5 produtos**, por causa disso, totalizamos **6 selects: um** para buscar os produtos e 5 para buscar as categorias dos produtos.

Esse é um problema conhecido como **N + 1 queries**. Vale lembrar que o uso do *OpenEntityManagerInView* não causa o problema (apenas torna mais fácil sua ocorrência) e que sua causa não é o comportamento *lazy*, já que ele também ocorrerá caso configuremos explicitamente o relacionamento como *Eager*. Mas, por que?

Quando buscarmos o produto podemos realizar a seguinte query: `select p from Produto p` (Como estamos usando *Hibernate*, podemos abreviar e escrever somente: `from Produto`). No entanto, como ela também precisará dos relacionamentos (categorias) será feito uma nova query. Isso porque em nenhum lugar dizemos à JPA que ela deverá trazer junto as categorias do produto.

Como podemos fazer para trazer as categorias junto com os produtos usando um join?

Reflita e veja a resposta em "Opinião"!