

```
1 package br.com.padon.application.controllers;
2
3 import br.com.padon.application.dtos.ProdutoVendaDto;
4 import br.com.padon.application.dtos.RelatorioDto;
5 import br.com.padon.application.models.Pertence;
6 import br.com.padon.application.models.Venda;
7 import br.com.padon.application.repositories.
    PertenceRepository;
8 import br.com.padon.application.repositories.
    ProdutoRepository;
9 import br.com.padon.application.repositories.
    VendaRepository;
10 import com.fasterxml.jackson.databind.JsonNode;
11 import org.springframework.beans.factory.annotation.
    Autowired;
12 import org.springframework.web.bind.annotation.*;
13
14 import java.text.ParseException;
15 import java.text.SimpleDateFormat;
16 import java.util.List;
17
18 @RestController
19 @CrossOrigin
20 @RequestMapping("/relatorio")
21 public class RelatorioController {
22
23     @Autowired
24     private VendaRepository venda;
25     @Autowired
26     private PertenceRepository pertence;
27     @Autowired
28     private ProdutoRepository produto;
29     private final SimpleDateFormat parser = new
    SimpleDateFormat("dd/MM/yyyy");
30
31     @PostMapping("/bydate")
32     private List<Venda> getVendasByDate(@RequestBody
    JsonNode node) throws ParseException {
33         return venda.getByDate(parser.parse(node.get("
    dtinicial").asText()), parser.parse(node.get("dtfinal").
    asText())));
34     }
35
36     @PostMapping("/produtos")
```

```
37     private List<RelatorioDto> getProdutos(@RequestBody
      JsonNode node) throws ParseException {
38         List<Venda> vendas = venda.getByDate(parser.parse(
            node.get("dtinicial").asText()), parser.parse(node.get("
            dtfinal").asText()));
39         return vendas.stream().map(v -> new RelatorioDto(
40             v.getVendaId(),
41             v.getDataVenda(),
42             v.getStatusVenda(),
43             v.getValorTotal(),
44             v.getComanda(),
45             produto.getProdutosFromVenda(v.getVendaId
            ()).stream().map(p -> {
46                 Pertence pertenceModel = pertence.
            getPertence(p.getProdutoId(), v.getVendaId());
47                 return new ProdutoVendaDto(
48                     p.getProdutoId(),
49                     p.getNome(),
50                     p.getFabricante(),
51                     p.getImage(),
52                     p.getCodigoDeBarras(),
53                     p.isBloqueado(),
54                     p.getPrecoPorQuilo(),
55                     p.getPrecoPorUnidade(),
56                     p.isPorQuilo(),
57                     pertenceModel.getPrecoTotal(),
58                     pertenceModel.getQuantidade(),
59                     pertenceModel.getPrecoAtual()
60                 );
61             }).toList()
62         )).toList();
63     }
64 }
65
```