

---

## Dados do Passageiro

Passageiro: Fred Junior

CPF: 12345678905

## Dados do Bilhete

Codigo: DRYMOD

Assento: 15

Origem: Mordor

Destino: Condado

Hora do Embarque: 17:00

Hora da Partida: 17:30

[Atualizar os dados](#) [Próximo](#)

```
*exibir-dados.html × App.java RepositorioDados.java index.html checking.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Bilhete de Onibus</title>
6 </head>
7 <body>
8   <div align="center">
9     <h1>Dados do Passageiro</h1>
10    <p>Passageiro: $passageiro</p>
11    <p>CPF: $cpf</p>
12  </div>
13  <br>
14  <div align="center">
15    <h1>Dados do Bilhete</h1>
16    <p>Codigo: $codigo</p>
17    #if ($assento)
18      <p>Assento: $assento</p>
19      <p>Data da Compra: $data</p>
20      <p>Origem: $origem</p>
21      <p>Destino: $destino</p>
22      <p>Hora do Embarque: $horaEmbarque</p>
23      <p>Hora da Partida: $horaPartida</p>
24    #else
25      <p>Assento não marcado!</p>
26    #end
27  </div>
28  <a href="/atualizar-dados?bilhete=$codigo">Atualizar os dados</a>
29  <a href="/marcar-assento?bilhete=$codigo">Próximo</a>
30 </body>
31 </html>
```

```

exibir-dados.html  App.java ×  RepositorioDados.java  Bilhete.java  Linha.java
49         } catch (Exception e) {
50             return e.getClass().getName() + ": " + e.getMessage();
51         }
52     }
53 });
54
55 Spark.get("/exibe-dados", (req, res) -> {
56     try {
57         String codigoBilhete = req.queryParams("bilhete");
58         Bilhete b = repositorioDados.buscaBilhetePorCodigo(codigoBilhete);
59         Map<String, Object> ctx = new HashMap<>();
60         ctx.put("codigo", b.getCodigo());
61         ctx.put("assento", b.getAssento());
62         ctx.put("data", b.getDataMarcacao());
63         ctx.put("passageiro", b.getPassageiro().getNome());
64         ctx.put("cpf", b.getPassageiro().getCpf());
65         ctx.put("origem", b.getLinha().getOrigem());
66         ctx.put("destino", b.getLinha().getDestino());
67         ctx.put("horaEmbarque", b.getLinha().getHoraEmbarque());
68         ctx.put("horaPartida", b.getLinha().getHoraPartida());
69
70         return ViewUtil.renderiza("/exibir-dados.html", ctx);
71     } catch (Exception e) {
72         return e.getClass().getName() + ": " + e.getMessage();
73     }
74 }
75 });
76
77 System.out.println(String.format("Escutando na porta %d...", Spark.port()));
78 }
79 }

```

```
exibir-dados.html  App.java  RepositorioDados.java x
12 public RepositorioDados(Connection conexao) {
13     this.conexao = conexao;
14 }
15
16 public Bilhete buscaBilhetePorCodigo(String codigo) throws SQLException {
17     try (PreparedStatement stmt = this.conexao.prepareStatement("SELECT b.cod
18         stmt.setString(1, codigo);
19         Bilhete b = null;
20         try (ResultSet rs = stmt.executeQuery()) {
21             if (rs.next()) {
22                 b = new Bilhete();
23                 b.setCodigo(rs.getString(1));
24                 b.setAssento(rs.getString(2));
25                 b.setDataMarcacao(rs.getDate(3).toLocalDate());
26                 Passageiro p = new Passageiro();
27                 b.setPassageiro(p);
28                 p.setId(rs.getInt(4));
29                 p.setNome(rs.getString(5));
30                 p.setCpf(rs.getString(6));
31                 Linha l = new Linha();
32                 b.setLinha(l);
33                 l.setOrigem(rs.getString(7));
34                 l.setDestino(rs.getString(8));
35                 l.setHoraEmbarque(rs.getDate(9).toLocalDate());
36                 l.setHoraPartida(rs.getDate(10).toLocalDate());
37             }
38         }
39     }
40     return b;
41 }
42 }
```