Linguagem de Programação III Marco Antônio Araújo Relatório de Refatoração — Sistema Ponto Jr.

Bárbara Santana Saab Maria Eduarda Tavares

A) Pacote: Controller – ManterController/PesquisaController

Mau cheiro: código duplicado (pág. 71)

Refatoração: extrair classe, criar método padrão (pág. 293)

Antes da refatoração:

ManterController

```
Histórico
                 Código-Fonte
      package controller;
   + import ...13 linhas
 3
 16
 17
      public class ManterCargoController extends HttpServlet {
 18
          protected void processRequest (HttpServletRequest request, HttpServletResponse response)
 19
   20
                 throws ServletException, IOException, ClassNotFoundException, SQLException {
 21
             String acao = request.getParameter("acao");
 22
             if (acao.equals("confirmarOperacao")) {
 23
                 confirmarOperacao(request, response);
 24
             ) else (
 25
                 if (acao.equals("prepararOperacao")) {
 26
                    prepararOperacao(request, response);
 27
                 }
 28
             3
 29
 30
 31
          @Override
          protected void doGet(HttpServletRequest request, HttpServletResponse response)
 33 +
                 throws ServletException, IOException {...9 linhas }
 42
 43
          protected void doPost(HttpServletRequest request, HttpServletResponse response)
 45 +
                 throws ServletException, IOException [...9 linhas ]
 54
 55
    +
          public String getServletInfo() {...3 linhas }
 59
          public void prepararOperacao(HttpServletRequest request, HttpServletResponse response
```

PesquisaController

```
🕸 PesquisaCargoController.java 🕺 🙆 ManterCargoController.java 🕺
Código-Fonte Histórico 🔯 🖟 🔻 🔻 🗸 🗸 🔁 🕌 🎧 🔁 ڬ 🕒 🗀 🔛 👑 📑
      package controller;
   + import ...8 linhas
11
12
      public class PesquisaCargoController extends HttpServlet {
13
14
   口
          protected void processRequest(HttpServletRequest request, HttpServletResponse response) throws ServletEx
15
              request.setAttribute("cargos", CargoDAO.getInstance().getAllCargos());
              RequestDispatcher view = request.getRequestDispatcher("/pesquisaCargo.jsp");
16
17
              view.forward(request, response);
18
19
20
   +
          protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException
24
25
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
29
```

Pacote Controller: nove classes para ManterController, e nove classes para PesquisaController

```
Serviços
Arquivos
                                 plo_JPA
                                                         Código-Fonte
                                                 Histórico
toFjr
                                             package controller;
Páginas Web
Arquivos Remotos
                                          + import ...13 linhas
                                      16
controller [
                                      17
                                             public class ManterAfastamentoController extends HttpServlet {

    ▲ LoginController.java

                                      18
    ManterAfastamentoController.java
                                       19
                                                 public void processRequest(HttpServletRequest request, HttpServletResponse
  ManterCargoController.iava
                                       20 +
                                                          throws ServletException, IOException, ClassNotFoundException, SQLExc

    ManterCartaoController.java

                                       31
 ManterContatoController.java
                                       32 +
                                                 public void confirmarOperacao(HttpServletReguest reguest, HttpServletRespons
 ManterDepartamentoController.iava
                                       64
 ManterFuncionarioController.java
                                       65
                                                 public void prepararOperacao(HttpServletRequest request, HttpServletResponse
  ManterHorarioController.iava
                                       66 ±
                                                          throws ServletException, ClassNotFoundException {...16 linhas }

    ManterInformeController.java

                                       82
 ManterUsuarioController.java
                                       83
                                                 @Override
 PesquisaAfastamentoController.java
                                                 protected void doGet(HttpServletRequest request, HttpServletResponse response
 PesquisaCargoController.java
                                                         throws ServletException, IOException [...11 linhas ]
                                       85 +
  PesquisaCartaoController.iava
                                       96
 PesquisaContatoController.java
                                       97
                                                 @Override
    PesquisaDepartamentoController.java
                                                 protected void doPost(HttpServletRequest request, HttpServletResponse respon
  PesquisaFuncionarioController.java
                                      99 +
                                                         throws ServletException, IOException {...9 linhas }
 PesquisaHorarioController.iava
                                      108
    PesquisaInformeController.iava
                                     109
                                                 @Override
  PesquisaUsuarioController.java
                                       0
                                                 public String getServletInfo() {
                                     111
                                                     return "Short description";
  dao
                                     112
  model
                                      113
```

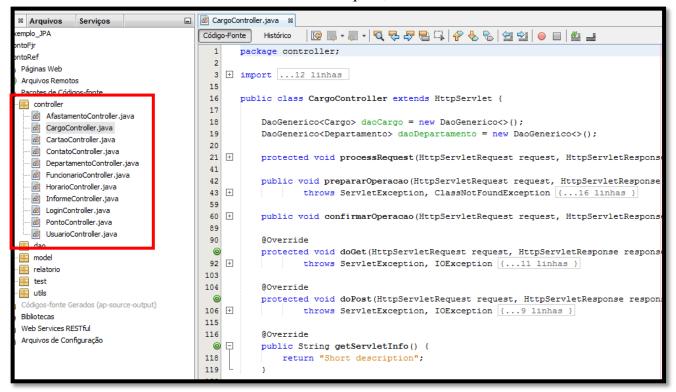
*São nove classes ManterController e nove classes PesquisaController. O procedimento de refatoração foi aplicado em todas, onde Manter e Pesquisa de cada classe foram unificados em uma classe Controller. Ou seja, antes eram 18 classes, após refatoração, ficaram 9 classes.

Após a refatoração:

Manter e Pesquisa unificados

```
4 >
Código-Fonte Histórico 🕼 🖟 🔻 🔻 🗸 🗸 🖓 🖶 🖫 🖓 😓 🖒 🔯 💇 🥚 🔲 🥙 🚅
       package controller;
    + import ...12 linhas
 15
 16
       public class CargoController extends HttpServlet {
 17
 18
           DaoGenerico < Cargo > daoCargo = new DaoGenerico <> ();
 19
          DaoGenerico < Departamento > daoDepartamento = new DaoGenerico <> ();
 20
    口
 21
          protected void processRequest(HttpServletRequest request, HttpServletResponse response) throws ServletException
 22
              String acao = reguest.getParameter("acao");
 23
              if (acao.eguals("confirmarOperacao")) {
 24
                  confirmarOperacao(request, response);
 25
               } else {
 26
                  if (acao.equals("prepararOperacao")) {
 27
                      prepararOperacao(request, response);
 28
                   } else {
 29
                      if (acao.equals("getOnly")) {
 30
                          prepararOperacao(request, response);
                      } else {
 32
 33
                           request.setAttribute("cargos", daoCargo.findAll(Cargo.class));
 34
                           RequestDispatcher view = request.getRequestDispatcher("/pesquisaCargo.jsp");
 35
                           view.forward(request, response);
 36
 37
 38
 39
```

Pacote Controller sem Manter e Pesquisa, e com 11 classes*



* As classes PontoController e LoginController não foram refatoradas, pois não possuem Manter e Pesquisa para as mesmas.

B) Pacote: DAO

Mau cheiro: código duplicado (pág. 71)

Refatoração: extrair classe, criar método padrão (pág. 293)

Antes da refatoração:

DAO

```
package dao;
import java.util.List;
import javax.persistence.EntityManager;
import javax.persistence.EntityTransaction;
import javax.persistence.TypedQuery;
import model.Afastamento;

public class AfastamentoDAO {
    private static AfastamentoDAO instance = new AfastamentoDAO();
    public static AfastamentoDAO getInstance() {...3 linhas }

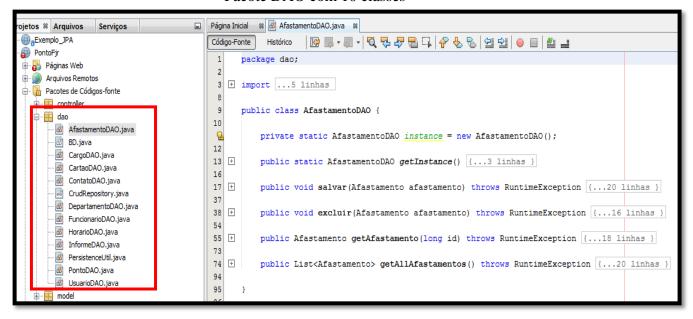
    public void salvar(Afastamento afastamento) throws RuntimeException {...20 linhas }

    public void excluir(Afastamento afastamento) throws RuntimeException {...16 linhas }

    public Afastamento getAfastamento(long id) throws RuntimeException {...18 linhas }

    public List<Afastamento> getAllAfastamentos() throws RuntimeException {...20 linhas }
}
```

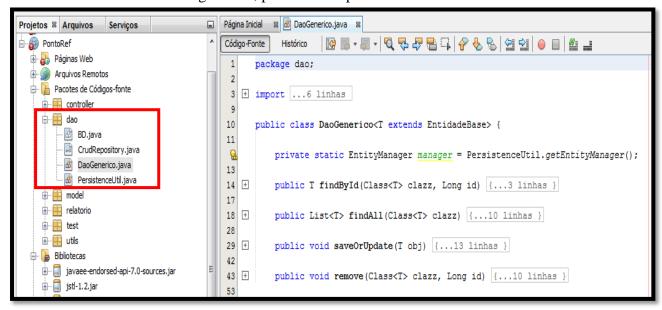
Pacote DAO com 10 classes



^{*}São dez classes DAO. O procedimento de refatoração foi aplicado em todas.

Após a refatoração:

DAO genérico, pacote com apenas uma classe



C) Pacote: Relatório

Mau cheiro: código duplicado (pág. 71)

Refatoração: extrair classe, criar método padrão (pág. 293)

Antes da refatoração: São seis classes que geram relatórios sem parâmetros. Todas foram substituídas por uma classe RelatórioGeneric.

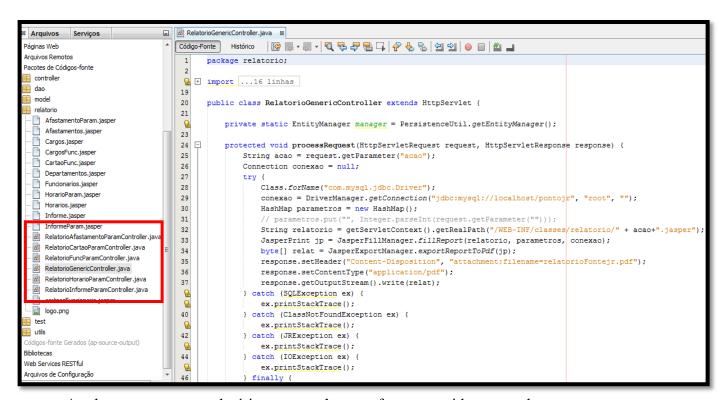
Pacote Relatório

```
■ RelatorioAfastFuncController.java ※
Arquivos
         Serviços
                                     Código-Fonte Histórico | 🚱 👼 🔻 🗸 🗸 🞝 🖶 📮 🖟 😓 🔁 🖆 💇 🔒 📗
  relatorio
    AfastamentoParam.jasper
                                             package relatorio;
    Cargos.jasper
                                       3 + import ...13 linhas
    CargosFunc.jasper
                                      16
    CartaoFunc.jasper
                                             public class RelatorioAfastFuncController extends HttpServlet {
                                      17
    Departamentos, jasper
                                                 protected void processRequest(HttpServletRequest request, HttpServletResponse response) {
                                      18
    Funcionarios, jasper
                                      19
                                              Connection conexao = null;
    HorarioParam, jasper
                                      20
                                                      try {
    Horarios, jasper
                                                          Class.forName("com.mysql.jdbc.Driver");
     intorme, jasper
                                                           conexao = DriverManager.getConnection("jdbc:mysql://localhost/pontojr", "root", "");
                                      22
    InformeParam.jasper
                                      23
                                                          HashMap parametros = new HashMap();
                                      24
                                                           //parametros.put("", Integer.parseInt(request.getParameter("")));
 RelatorioAfastamentoParamController.java
                                      25
                                                          String relatorio = getServletContext().getRealPath("/WEB-INF/classes/relatorio")+"/Afastamentos.jasper
  RelatorioCargDeptController.java
                                      26
                                                          JasperPrint jp = JasperFillManager.fillReport(relatorio, parametros, conexao);
    RelatorioCargFuncController.java
                                                          byte[] relat = JasperExportManager.exportReportToPdf(jp);
                                      27
    RelatorioCartaoController.java
                                      28
                                                          response.setHeader("Content-Disposition", "attachment; filename=relatorioAfastamento.pdf");
 RelatorioCartaoParamController.java
                                      29
                                                          response.setContentType("application/pdf");
    RelatorioCfuncionarioController.java
                                       30
                                                          response.getOutputStream().write(relat);
 RelatorioFuncParamController.iava
                                       31
                                                      } catch (SQLException ex) {
    RelatorioHorarioController.java
                                       32
                                                          ex.printStackTrace();

    RelatorioHorarioParamController.iava

                                       33
                                                      } catch (ClassNotFoundException ex) {
    RelatorioInformeController.java
                                       34
                                                           ex.printStackTrace();
                                       35
                                                      } catch (JRException ex) {
  cartoesFuncionario.jasper
                                      36
                                                          ex.printStackTrace();
 logo.png
                                       37
                                                      } catch (IOException ex) {
E test
                                       38
                                                          ex.printStackTrace();
                                                      } finally {
                                       39
```

Após a refatoração:



As classes que geram relatórios com parâmetros foram mantidas separadas.

REFERÊNCIA BIBLIOGRÁFICA

FOWLER, Martin. Refatoração – Aperfeiçoando o Projeto de Código Existente. **Bookman**, 2004.