Classes DAO

```java
public class DAO {
    protected PreparedStatement ps;
    protected ResultSet rs;
    protected Connection connection;
    protected String sql;
}
```

```java
public class CategoriaDAO extends DAO {

    // método para inserir uma categoria na base de dados
    public void inserir(String categoria) {
        Conexao conexao = new Conexao();
        connection = conexao.conectar();
        sql = "insert into java_categoria values(categoria_sequence.nextval, ?)";

        try {
            ps = connection.prepareStatement(sql);
            ps.setString(1, categoria);
            ps.execute();
            ps.close();
            connection.close();
            conexao.desconectar();
        } catch (SQLException e) {
            System.out.println("Erro ao cadastrar categoria\n" + e);
        }
```

```java
// método para retornar todas as categorias registradas na base de dados
public List<Categoria> listar() {
    List<Categoria> lista = new ArrayList<>();
    Categoria categoria;
    Conexao conexao = new Conexao();
    connection = conexao.conectar();
    sql = "select * from java_categoria order by categoria";

    try {
        ps = connection.prepareStatement(sql);
        rs = ps.executeQuery();
        while(rs.next()) {
            categoria = new Categoria();
            categoria.setCategoriaId(rs.getInt("categoria_id"));
            categoria.setCategoria(rs.getString("categoria"));
            lista.add(categoria);
        }
        ps.close();
        connection.close();
        conexao.desconectar();
    } catch (SQLException e) {
        System.out.println("Erro ao listar categoria\n" + e);
    }
    return lista;
```

```java
/* método para inserir um pedido na base de dados. O método retorna true para saber
 * se  pedido foi inserido ou não. Se o pedido não foi inserido, seus detalhes não serão
 * inseridos na tabela detalhe pedido
 */
public boolean inserir(Pedido pedido) {
    boolean inseriu = false;
    Conexao conexao = new Conexao();
    connection = conexao.conectar();
    sql = "insert into java_pedido values(pedido_sequence.nextval, ?, ?, ?)";

    try {
        ps = connection.prepareStatement(sql);
        ps.setString(1, pedido.getNomeContato());
        ps.setString(2, pedido.getEnderecoContato());
        ps.setDate(3, Date.valueOf(pedido.getData())); // converte a data para Data (sql)
        ps.execute();
        inseriu = true;
        ps.close();
        connection.close();
        conexao.desconectar();
    } catch (SQLException e) {
        System.out.println("Erro ao inserir produto\n" + e);
    }

    return inseriu;
}
```

```java
// mÃ©todo para alterar os dados de um contato
public void alterar(Contato contato) {
    Conexao conexao = new Conexao();
    connection = conexao.conectar();
    sql = "update java_contato set nome = ?, email = ?, "
            + "telefone = ?,"
            + " dataNascimento = ? where id = ?";
    try {
        ps = connection.prepareStatement(sql);
        ps.setString(1, contato.getNome());
        ps.setString(2, contato.getEmail());
        ps.setString(3, contato.getTelefone());
        ps.setDate(4, Date.valueOf(contato.getDataNascimento()));
        ps.setInt(5, contato.getId());
        ps.execute();
        ps.close();
        conexao.desconectar();
    }
    catch(SQLException e) {
        System.out.println("Erro ao alterar dados do contato na
                + "base de dados\n" + e);
    }
}
}
```

```java
/* método para retornar o código do último pedido inserido. Esse
 * código será inserido como chave estrangeira na tabela java_pedidodetalhe
 */
public Integer obterPedidoId() {
    Integer id = 0;
    Conexao conexao = new Conexao();
    connection = conexao.conectar();
    sql = "select max(pedido_id) from java_pedido";

    try {
        ps = connection.prepareStatement(sql);
        rs = ps.executeQuery();
        if(rs.next()) {
            id = rs.getInt("max(pedido_id)");
        }
        rs.close();
        ps.close();
        connection.close();
        conexao.desconectar();
    } catch (SQLException e) {
        System.out.println("erro ao pesquisar o último "
                + "id inserido\n" + e);
    }
    return id;
}
}
```

Classes Servlet

```java
@WebServlet("/listagem")
public class ListagemServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)
     */
    protected void service(HttpServletRequest request,
        HttpServletResponse response) throws ServletException, IOException {
        Util util = new Util();
        ContatoDAO dao = new ContatoDAO();
        Contato contato;
        String senha = util.criptografar(request.getParameter("senha"));
        String email = request.getParameter("email");
        contato = dao.pesquisar(email, senha);
        if(contato != null) {
            RequestDispatcher dispatcher = request.getRequestDispatcher("/form/formListagem.jsp");
            dispatcher.forward(request, response);
        }
    }
}
```

```java
/**
 * Servlet implementation class CadastroServlet
 */
@WebServlet("/cadastro")
public class CadastroServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse respo
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        Util util = new Util();
        ContatoDAO dao = new ContatoDAO();
        Contato contato = new Contato();
        String senha = util.criptografar(request.getParameter("senha"));
        LocalDate data = util.formatarData(request.getParameter("data"));
        contato.setNome(request.getParameter("nome"));
        contato.setEmail(request.getParameter("email"));
        contato.setSenha(senha);
        contato.setTelefone(request.getParameter("fone"));
        contato.setDataNascimento(data);
        dao.inserir(contato);
        // redireciona para index.jsp
        RequestDispatcher dispatcher = request.getRequestDispatcher("index.jsp");
        dispatcher.forward(request, response);
    }
```

```java
@WebServlet("/cadastroPedido")
public class CadastroPedidoServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void service(HttpServletRequest request, HttpServletResponse res
        PedidoDAO pedidoDao = new PedidoDAO();
        PedidoDetalheDAO detalheDao = new PedidoDetalheDAO();

        Produto produto = new Produto();
        Categoria categoria = new Categoria();
        Pedido pedido = new Pedido();
        PedidoDetalhe detalhe = new PedidoDetalhe();

        String nomeContato = request.getParameter("nomecontato");
        String enderecoContato = request.getParameter("enderecocontato");
        LocalDate data = formatarData(request.getParameter("data"));
        Integer produtoId = Integer.parseInt(request.getParameter("produto"));
        int quantidade = Integer.parseInt(request.getParameter("quantidade"));
        double total = Double.parseDouble(request.getParameter("total"));

        // configura o objeto Pedido
        pedido.setNomeContato(nomeContato);
        pedido.setEnderecoContato(enderecoContato);
        pedido.setData(data);

        // insere o objeto pedido no banco de dados
        // método inserir pedido retornar true ou false --> para saber se inseriu ou não

        if(pedidoDao.inserir(pedido) ) {
            produto.setProdutoId(produtoId);
            pedido.setPedidoId(pedidoDao.obterPedidoId());
            detalhe.setPedido(pedido);
            detalhe.setProduto(produto);
            detalhe.setQuantidade(quantidade);
            detalhe.setTotal(total);
            detalheDao.inserir(detalhe);
        }

        response.sendRedirect("index.jsp");

    // método para converter a data de String para LocalDate
    private LocalDate formatarData(String data) {
        String[] dataAux = data.split("-");
        String dataString = "";
        dataString = dataAux[2] + "/" + dataAux[1] + "/" + dataAux[0];

        DateTimeFormatter formato = DateTimeFormatter.ofPattern("dd/MM/yyyy");
        LocalDate localDate = LocalDate.parse(dataString, formato);
        return localDate;
    }
```

```java
@WebServlet("/cadastro")
public class CadastroServelet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    /**
     * @see HttpServlet#HttpServlet()
     */
    public CadastroServelet() {
        super();
        // TODO Auto-generated constructor stub
    }
    /**
     * @see HttpServlet#service(HttpServletRequest request,
     * HttpServletResponse response)
     */
    protected void service(HttpServletRequest request,
    HttpServletResponse response) throws ServletException, IOException {
        String nome = request.getParameter("nome");
        String cpf = request.getParameter("cpf");
        Integer id = Integer.parseInt(request.getParameter("departamento"));
        Departamento departamento = new Departamento();
        departamento.setId(id);
        Empregado empregado = new Empregado();
        empregado.setCpf(cpf);
        empregado.setNome(nome);
        empregado.setDepartamento(departamento);
        new EmpregadoDAO().cadastrar(empregado);
        //redireciona para a pagina index.jsp
        response.sendRedirect("index.jsp");
    }
}
```

```jsp
<%  List<Empregado> lista = new EmpregadoDAO().listar(); %>
<div class="container">
    <table class="striped">
        <thead>
            <tr>
                <th>CPF</th>
                <th>Nome</th>
                <th>Departamento</th>
            </tr>
        </thead>
        <tbody>
            <%  for(Empregado empregado : lista) { %>
            <tr>
                <td><%= empregado.getCpf() %></td>
                <td><%= empregado.getNome() %></td>
                <td><%= empregado.getDepartamento().getNome() %></td>
            </tr>
            <% } %>
        </tbody>
    </table>
</div>
</body>
```

```jsp
<body>
    <%  List<Departamento> lista = new DepartamentoDAO().listar(); %>
    <div class="container">
        <div class="row">
        <form class="col s12" action="cadastro">
            <div class="row">
                <div class="input-field col s6">
                    <input id="nome" name="nome" type="text">
                    <label for="nome">Nome</label>
                </div>
            </div>
            <div class="row">
                <div class="input-field col s6">
                    <input id="cpf" name="cpf" type="text">
                    <label for="cpf">CPF</label>
                </div>
            </div>
            <div class="row">
                <label>Departamento</label>
                <select name="departamento">
                    <option value="" disabled selected>Escolha o departamento</option>
                    <%  for(Departamento dep : lista) { %>
                        <option value="<%= dep.getId()%>"><%= dep.getNome() %></option>
                    <% } %>

                </select>
            </div>
            <div class="row">
                <button class="btn waves-effect waves-teal" type="submit">Cadastrar
                <i class="material-icons right">send</i></button>
            </div>
        </form>
    </div>
    </div>
```

```jsp
<body>

    <%  List<Empregado> lista = new EmpregadoDAO().listar(); %>

    <div class="container">
        <table class="striped">
            <thead>
                <tr>
                    <th>CPF</th>
                    <th>Nome</th>
                    <th>Departamento</th>
                </tr>
            </thead>
            <tbody>
                <%  for(Empregado empregado : lista) { %>
                <tr>
                    <td><%= empregado.getCpf() %></td>
                    <td><%= empregado.getNome() %></td>
                    <td><%= empregado.getDepartamento().getNome() %>
                </tr>
                <% } %>
            </tbody>
        </table>

    </div>

</body>
</html>
```

Imports jsp

```jsp
<%@page import="br.fiap.modelo.entidade.Departamento"%>
<%@page import="br.fiap.dao.conexao.DepartamentoDAO"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
```

Imports html/css

```jsp
<%@page import="br.fiap.modelo.entidade.Departamento"%>
<%@page import="br.fiap.dao.conexao.DepartamentoDAO"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
```

```jsp
<div class="container">
    <div class="content">

        <!--FORMULÁRIO DE ALTERAÇÃO-->
        <div id="cadastro">
            <form method="post" action="../alterar">
                <h1>Alteração de Dados</h1>
                <%  Integer id = Integer.parseInt(request.getParameter("id"));
                    Contato contato = new ContatoDAO().pesquisarId(id);
                %>
                <p>
                    <input id="id" name="id" type="hidden" value=<%= contato.getId() %>>
                </p>
                <p>
                    <label for="nome">Nome</label>
                    <input id="nome" name="nome" type="text" value="<%= contato.getNome() %>" />
                </p>
                <p>
                    <label for="email">Email</label>
                    <input id="email" name="email" type="email" value="<%= contato.getEmail() %>" />
                </p>
                <p>
                    <label for="senha">Senha</label>
                    <input id="senha" name="senha" type="password" value="" />
                </p>
                <p>
                    <label for="fone">Telefone</label>
                    <input id="fone" name="fone" type="text" value="<%= contato.getTelefone() %>" />
                </p>
                <p>
                    <label for="data">Data Nascimento</label>
                    <input id="data" name="data" type="date" value=<%= contato.getDataNascimento() %>
                </p>
                <p>
                    <input type="submit" value="Alterar" />
```