**Desenvolvendo cadastro, consulta,**

**edição e exclusão de contas:**

Primeiro, precisamos desenvolver a classe

de repositório de conta no banco de dados:



/repositories/ContaRepository.java

**package** br.com.cotiinformatica.repositories;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.text.SimpleDateFormat;

**import** java.util.Date;

**import** java.util.List;

**import** javax.sql.DataSource;

**import** org.springframework.jdbc.core.JdbcTemplate;

**import** org.springframework.jdbc.core.RowMapper;

**import** br.com.cotiinformatica.entities.Conta;

**import** br.com.cotiinformatica.enums.TipoConta;

**public** **class** ContaRepository {

// atributo

**private** JdbcTemplate jdbcTemplate;

// construtor

**public** ContaRepository(DataSource dataSource) {

jdbcTemplate = **new** JdbcTemplate(dataSource);

}

**public** **void** create(Conta conta) **throws** Exception {

String query = "insert into conta(nome, valor, data, tipo,

observacoes, idusuario) values(?,?,?,?,?,?)";

Object[] params = {

conta.getNome(),

conta.getValor(),

java.sql.Date.*valueOf*(**new** SimpleDateFormat

("yyyy-MM-dd").format(conta.getData())),

conta.getTipo() == TipoConta.***RECEBER***

? 1 : conta.getTipo() == TipoConta.***PAGAR*** ? 2 : 0,

conta.getObservacoes(),

conta.getUsuario().getIdUsuario()

};

jdbcTemplate.update(query, params);

}

**public** **void** update(Conta conta) **throws** Exception {

String query = "update conta set nome=?, valor=?,

data=?, tipo=?, observacoes=? where idconta=?";

Object[] params = {

conta.getNome(),

conta.getValor(),

java.sql.Date.*valueOf*(**new** SimpleDateFormat

("yyyy-MM-dd").format(conta.getData())),

conta.getTipo() == TipoConta.***RECEBER***

? 1 : conta.getTipo() == TipoConta.***PAGAR*** ? 2 : 0,

conta.getObservacoes(),

conta.getIdConta()

};

jdbcTemplate.update(query, params);

}

**public** **void** delete(Conta conta) **throws** Exception {

String query = "delete from conta where idconta=?";

Object[] params = {

conta.getIdConta()

};

jdbcTemplate.update(query, params);

}

**public** List<Conta> findByUsuarioAndData(Integer idUsuario,

Date dataIni, Date dataFim) **throws** Exception {

String query = "select \* from conta where idusuario=?

and data between ? and ? order by data";

Object[] params = {

idUsuario,

java.sql.Date.*valueOf*(**new** SimpleDateFormat

("yyyy-MM-dd").format(dataIni)),

java.sql.Date.*valueOf*(**new** SimpleDateFormat

("yyyy-MM-dd").format(dataFim))

};

List<Conta> lista = jdbcTemplate.query

(query, params, **new** RowMapper<Conta>() {

@Override

**public** Conta mapRow(ResultSet rs, **int** rowNum)

**throws** SQLException {

Conta conta = **new** Conta();

**try** {

conta.setIdConta(rs.getInt("idconta"));

conta.setNome(rs.getString("nome"));

conta.setValor(rs.getDouble("valor"));

conta.setData(**new** SimpleDateFormat()

.parse(rs.getString("data")));

conta.setTipo(rs.getInt("tipo") == 1 ?

TipoConta.***RECEBER***

: rs.getInt("tipo") == 2

? TipoConta.***PAGAR*** : **null**);

conta.setObservacoes

(rs.getString("observacoes"));

}

**catch**(Exception e) {

e.printStackTrace();

}

**return** conta;

}

});

**return** lista;

}

**public** Conta findById(Integer idConta, Integer idUsuario)

**throws** Exception {

String query = "select \* from conta where idconta=?

and idusuario=?";

Object[] params = {

idConta, idUsuario

};

List<Conta> lista = jdbcTemplate.query

(query, params, **new** RowMapper<Conta>() {

@Override

**public** Conta mapRow(ResultSet rs, **int** rowNum)

**throws** SQLException {

Conta conta = **new** Conta();

**try** {

conta.setIdConta(rs.getInt("idconta"));

conta.setNome(rs.getString("nome"));

conta.setValor(rs.getDouble("valor"));

conta.setData(**new** SimpleDateFormat()

.parse(rs.getString("data")));

conta.setTipo(rs.getInt("tipo") == 1

? TipoConta.***RECEBER***

: rs.getInt("tipo") == 2

? TipoConta.***PAGAR*** : **null**);

conta.setObservacoes

(rs.getString("observacoes"));

}

**catch**(Exception e) {

e.printStackTrace();

}

**return** conta;

}

});

**if**(lista.size() == 1)

**return** lista.get(0);

**else**

**return** **null**;

}

}

**Criando as páginas de cadastro, consulta e edição de contas:**

JSP – JAVA SERVER PAGES

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Para cada página JSP, vamos criar uma classe de controle capaz de mapear a navegação de cada página do sistema:

**package** br.com.cotiinformatica.controller;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.servlet.ModelAndView;

@Controller

**public** **class** CadastroContasController {

@RequestMapping(value = "/admin/cadastro-contas")

**public** ModelAndView cadastroContas() {

//WEB-INF/views/admin/cadastro-contas.jsp

ModelAndView modelAndView

= **new** ModelAndView("admin/cadastro-contas");

**return** modelAndView;

}

}

**package** br.com.cotiinformatica.controller;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.servlet.ModelAndView;

@Controller

**public** **class** ConsultaContasController {

@RequestMapping(value = "/admin/consulta-contas")

**public** ModelAndView consultaContas() {

//WEB-INF/views/admin/consulta-contas.jsp

ModelAndView modelAndView

= **new** ModelAndView("admin/consulta-contas");

**return** modelAndView;

}

}

**package** br.com.cotiinformatica.controller;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.servlet.ModelAndView;

@Controller

**public** **class** EdicaoContasController {

@RequestMapping(value = "/admin/edicao-contas")

**public** ModelAndView edicaoContas() {

//WEB-INF/views/admin/edicao-contas.jsp

ModelAndView modelAndView

= **new** ModelAndView("admin/edicao-contas");

**return** modelAndView;

}

}

Desenhando a página de cadastro de contas:

/WEB-INF/views/admin/cadastro-contas.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

<!-- CDN da folha de estilos CSS do bootstrap -->

<link href=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/css/bootstrap.min.css"* rel=*"stylesheet"* />

<style>

**label***.error* { color: *#df4759*; }

**input***.error***,** **textarea***.error* { border: *2px solid #df4759*; }

</style>

</head>

<body>

<!-- menu superior da aplicação -->

<jsp:include page=*"/WEB-INF/views/admin/components/navbar.jsp"*>

</jsp:include>

<div class=*"container mt-3"*>

<div class=*"card"*>

<div class=*"card-body"*>

<h5>Cadastro de Contas</h5>

<p>Preencha os campos para incluir uma conta.</p>

<hr/>

<form id=*"form\_cadastro"*>

<div class=*"row mb-2"*>

<div class=*"col-md-6"*>

<label>Nome da conta:</label>

<input type=*"text"* name=*"nome"*

id=*"nome"* class=*"form-control"*/>

</div>

<div class=*"col-md-3"*>

<label>Valor da conta:</label>

<input type=*"text"* name=*"valor"*

id=*"valor"* class=*"form-control"*/>

</div>

<div class=*"col-md-3"*>

<label>Data da conta:</label>

<input type=*"date"* name=*"data"*

id=*"data"* class=*"form-control"*/>

</div>

</div>

<div class=*"row mb-2"*>

<div class=*"col-md-3"*>

<label>Selecione o tipo da

conta:</label>

<div>

<input type=*"radio"*

name=*"tipo"* value=*"1"* checked=*"checked"*/> Conta a receber

</div>

<div>

<input type=*"radio"*

name=*"tipo"* value=*"2"*/>

Conta a pagar

</div>

</div>

<div class=*"col-md-9"*>

<label>Observações da

conta:</label>

<textarea name=*"observacoes"*

id=*"observacoes"* class=*"form-*

*control"*></textarea>

</div>

</div>

<div class=*"row mb-2"*>

<div class=*"col-md-12"*>

<input type=*"submit"* class=*"btn*

*btn-success"* value=*"Realizar Cadastro"*/>

</div>

</div>

</form>

</div>

</div>

</div>

<!-- CDN do arquivo javascript do bootstrap -->

<script src=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"*></script>

<!-- CDN do arquivo javascript do JQuery -->

<script src=*"https://code.jquery.com/jquery-3.6.3.min.js"*></script>

<!-- CDN dos arquivos da biblioteca JQuery Validation -->

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/jquery.validate.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/additional-methods.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/localization/messages\_pt\_BR.min.js"*></script>

<script>

$(document).ready(**function**() {

$("#form\_cadastro").validate({

rules: {

'nome' : { required: **true**, minlength : 6,

maxlength: 150 },

'valor' : { required: **true** },

'data' : { required: **true** },

'tipo' : { required: **true** },

'observacoes' : { required: **true**,

minlength : 6, maxlength: 500 },

}

});

})

</script>

</body>

</html>

Criando a classe de modelo para capturar os campos da página de cadastro de contas:

/models/CadastroContasModel.java

**package** br.com.cotiinformatica.models;

**import** lombok.AllArgsConstructor;

**import** lombok.Getter;

**import** lombok.NoArgsConstructor;

**import** lombok.Setter;

**import** lombok.ToString;

@Setter

@Getter

@NoArgsConstructor

@AllArgsConstructor

@ToString

**public** **class** CadastroContasModel {

**private** String nome;

**private** String valor;

**private** String data;

**private** String tipo;

**private** String observacoes;

}

**Utilizando a classe model para fazer a captura do formulário:**

package br.com.cotiinformatica.controller;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.servlet.ModelAndView;

import br.com.cotiinformatica.models.CadastroContasModel;

@Controller

public class CadastroContasController {

@RequestMapping(value = "/admin/cadastro-contas")

public ModelAndView cadastroContas() {

//WEB-INF/views/admin/cadastro-contas.jsp

ModelAndView modelAndView = new ModelAndView

("admin/cadastro-contas");

**modelAndView.addObject**

**("model", new CadastroContasModel());**

return modelAndView;

}

}

Utilizando o objeto **model** para fazermos a captura do formulário:

/WEB-INF/views/admin/cadastro-contas.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ taglib uri=*"http://www.springframework.org/tags/form"* prefix=*"form"* %>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

<!-- CDN da folha de estilos CSS do bootstrap -->

<link href=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/css/bootstrap.min.css"* rel=*"stylesheet"* />

<style>

**label***.error* { color: *#df4759*; }

**input***.error***,** **textarea***.error* { border: *2px solid #df4759*; }

</style>

</head>

<body>

<!-- menu superior da aplicação -->

<jsp:include page=*"/WEB-INF/views/admin/components/navbar.jsp"*></jsp:include>

<div class=*"container mt-3"*>

<div class=*"card"*>

<div class=*"card-body"*>

<h5>Cadastro de Contas</h5>

<p>Preencha os campos para incluir uma conta.</p>

<hr/>

<div class=*"mb-2"*>

<strong>${mensagem}</strong>

</div>

<form id=*"form\_cadastro"*

action=*"cadastrar-conta"* method=*"post"*>

<div class=*"row mb-2"*>

<div class=*"col-md-6"*>

<label>Nome da conta:</label>

<form:input path=*"model.nome"*

type=*"text"* name=*"nome"* id=*"nome"* class=*"form-control"*/>

</div>

<div class=*"col-md-3"*>

<label>Valor da conta:</label>

<form:input path=*"model.valor"*

type=*"text"* name=*"valor"* id=*"valor"* class=*"form-control"*/>

</div>

<div class=*"col-md-3"*>

<label>Data da conta:</label>

<form:input path=*"model.data"*

type=*"date"* name=*"data"* id=*"data"* class=*"form-control"*/>

</div>

</div>

<div class=*"row mb-2"*>

<div class=*"col-md-3"*>

<label>Selecione o tipo da

conta:</label>

<div>

<form:input

path=*"model.tipo"* type=*"radio"* name=*"tipo"* value=*"1"* checked=*"checked"*/> Conta a receber

</div>

<div>

<form:input

path=*"model.tipo"* type=*"radio"* name=*"tipo"* value=*"2"*/> Conta a pagar

</div>

</div>

<div class=*"col-md-9"*>

<label>Observações da

conta:</label>

<form:textarea

path=*"model.observacoes"* name=*"observacoes"* id=*"observacoes"* class=*"form-control"*></form:textarea>

</div>

</div>

<div class=*"row mb-2"*>

<div class=*"col-md-12"*>

<input type=*"submit"* class=*"btn*

*btn-success"*

value=*"Realizar Cadastro"*/>

</div>

</div>

</form>

</div>

</div>

</div>

<!-- CDN do arquivo javascript do bootstrap -->

<script src=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"*></script>

<!-- CDN do arquivo javascript do JQuery -->

<script src=*"https://code.jquery.com/jquery-3.6.3.min.js"*></script>

<!-- CDN dos arquivos da biblioteca JQuery Validation -->

<script src=*"https://cdnjs.cloudflare.com/ajax/libs*

*/jquery-validate/1.19.5/jquery.validate.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs*

*/jquery-validate/1.19.5/additional-methods.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs*

*/jquery-validate/1.19.5/localization/messages\_pt\_BR.min.js"*></script>

<script>

$(document).ready(**function**() {

$("#form\_cadastro").validate({

rules: {

'nome' : { required: **true**, minlength : 6,

maxlength: 150 },

'valor' : { required: **true** },

'data' : { required: **true** },

'tipo' : { required: **true** },

'observacoes' : { required: **true**, minlength : 6,

maxlength: 500 },

}

});

})

</script>

</body>

</html>

**Voltando no controlador para criar o método POST que irá receber o SUBMIT do formulário:**

**package** br.com.cotiinformatica.controller;

**import** java.text.SimpleDateFormat;

**import** javax.servlet.http.HttpServletRequest;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.servlet.ModelAndView;

**import** br.com.cotiinformatica.dtos.UsuarioDTO;

**import** br.com.cotiinformatica.entities.Conta;

**import** br.com.cotiinformatica.entities.Usuario;

**import** br.com.cotiinformatica.enums.TipoConta;

**import** br.com.cotiinformatica.models.CadastroContasModel;

**import** br.com.cotiinformatica.repositories.ContaRepository;

@Controller

**public** **class** CadastroContasController {

@Autowired

**private** ContaRepository contaRepository;

@RequestMapping(value = "/admin/cadastro-contas")

**public** ModelAndView cadastroContas() {

// WEB-INF/views/admin/cadastro-contas.jsp

ModelAndView modelAndView

= **new** ModelAndView("admin/cadastro-contas");

modelAndView.addObject

("model", **new** CadastroContasModel());

**return** modelAndView;

}

@RequestMapping(value = "/admin/cadastrar-conta",

method = RequestMethod.***POST***)

**public** ModelAndView cadastrarConta(CadastroContasModel model,

HttpServletRequest request) {

ModelAndView modelAndView

= **new** ModelAndView("admin/cadastro-contas");

**try** {

//capturando o usuário autenticado na sessão

UsuarioDTO usuarioDTO = (UsuarioDTO)

request.getSession().getAttribute("usuario");

Conta conta = **new** Conta();

conta.setUsuario(**new** Usuario());

conta.setNome(model.getNome());

conta.setValor(Double.*parseDouble*

(model.getValor()));

conta.setData(**new** SimpleDateFormat

("yyyy-MM-dd").parse(model.getData()));

conta.setTipo(model.getTipo().equals("1")

? TipoConta.***RECEBER***

: model.getTipo().equals("2")

? TipoConta.***PAGAR*** : **null**);

conta.setObservacoes(model.getObservacoes());

conta.getUsuario().setIdUsuario

(usuarioDTO.getIdUsuario());

contaRepository.create(conta);

model = **new** CadastroContasModel();

modelAndView.addObject("mensagem",

"Conta cadastrada com sucesso.");

}

**catch**(Exception e) {

modelAndView.addObject("mensagem",

"Falha ao cadastrar conta: " + e.getMessage());

}

modelAndView.addObject("model", model);

**return** modelAndView;

}

}

**Executando e testando o cadastro da conta:**

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**No PostGreSQL:**

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Desenhando a página de consulta de contas:

/WEB-INF/views/admin/consulta-contas.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ taglib uri=*"http://www.springframework.org/tags/form"* prefix=*"form"* %>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

<!-- CDN da folha de estilos CSS do bootstrap -->

<link href=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/css/bootstrap.min.css"* rel=*"stylesheet"* />

<style>

**label***.error* { color: *#df4759*; }

**input***.error***,** **textarea***.error* { border: *2px solid #df4759*; }

</style>

</head>

<body>

<!-- menu superior da aplicação -->

<jsp:include page=*"/WEB-INF/views/admin/components/navbar.jsp"*>

</jsp:include>

<div class=*"container mt-3"*>

<div class=*"card"*>

<div class=*"card-body"*>

<h5>Consultas de Contas</h5>

<p>Utilize o filtro abaixo para

consultar contas.</p>

<hr/>

<div class=*"mb-2"*>

<strong>${mensagem}</strong>

</div>

<form id=*"form\_consulta"*>

<div class=*"row mb-2"*>

<div class=*"col-md-3"*>

<input type=*"date"* id=*"dataIni"*

name=*"dataIni"*

class=*"form-control"*/>

</div>

<div class=*"col-md-3"*>

<input type=*"date"* id=*"dataFim"*

name=*"dataFim"*

class=*"form-control"*/>

</div>

<div class=*"col-md-6"*>

<input type=*"submit"* class=*"btn*

*btn-success"* value=*"Pesquisar Contas"*/>

</div>

</div>

</form>

<div class=*"table-responsive mt-2"*>

<table class=*"table table-sm table-hover"*>

<thead>

<tr>

<th>Nome da conta</th>

<th>Valor</th>

<th>Data</th>

<th>Tipo</th>

<th>Operações</th>

</tr>

</thead>

<tbody>

<tr>

<td></td>

<td></td>

<td></td>

<td></td>

<td>

<a

href=*"/sistemacontas/admin/edicao-contas"* class=*"btn btn-primary btn-sm"*>

Editar

</a>

<a

href=*"/sistemacontas/admin/exclusao-contas"* class=*"btn btn-danger btn-sm"*>

Excluir

</a>

</td>

</tr>

</tbody>

</table>

</div>

</div>

</div>

</div>

<!-- CDN do arquivo javascript do bootstrap -->

<script src=*"https://cdn.jsdelivr.net/npm/bootstrap*

*@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"*></script>

<!-- CDN do arquivo javascript do JQuery -->

<script src=*"https://code.jquery.com/jquery-3.6.3.min.js"*></script>

<!-- CDN dos arquivos da biblioteca JQuery Validation -->

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/jquery.validate.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/additional-methods.min.js"*></script>

<script src=*"https://cdnjs.cloudflare.com/ajax/libs/jquery-*

*validate/1.19.5/localization/messages\_pt\_BR.min.js"*></script>

<script>

$(document).ready(**function**() {

$("#form\_consulta").validate({

rules: {

'dataIni' : { required: **true** },

'dataFim' : { required: **true** }

}

});

})

</script>

</body>

</html>

Interface gráfica do usuário, Texto, Email

Descrição gerada automaticamente

Criando a model para a página de consulta:

/models/ConsultaContasModel.java

**package** br.com.cotiinformatica.models;

**import** lombok.AllArgsConstructor;

**import** lombok.Getter;

**import** lombok.NoArgsConstructor;

**import** lombok.Setter;

**import** lombok.ToString;

@Setter

@Getter

@NoArgsConstructor

@AllArgsConstructor

@ToString

**public** **class** ConsultaContasModel {

**private** String dataIni;

**private** String dataFim;

}

**Subindo o trabalho para o GITHUB:**

Samsung@DESKTOP-P9F6D9F MINGW64 ~/Desktop/COTI - Aulas/2023 - Java WebDeveloper SQS 18h as 22h (Início em 09.01)/workspace/sistemacontas (main)

**$ git add .**

Samsung@DESKTOP-P9F6D9F MINGW64 ~/Desktop/COTI - Aulas/2023 - Java WebDeveloper SQS 18h as 22h (Início em 09.01)/workspace/sistemacontas (main)

**$ git commit -m 'Cadastro de contas'**

Samsung@DESKTOP-P9F6D9F MINGW64 ~/Desktop/COTI - Aulas/2023 - Java WebDeveloper SQS 18h as 22h (Início em 09.01)/workspace/sistemacontas (main)

**$ git push -u origin main**

Texto

Descrição gerada automaticamente