<%--

Document : Budget

Created on : Apr 13, 2015, 3:30:50 PM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<link href="../bootstrap.min.css" rel="stylesheet"/>

<div style="line-height: 2em; padding: 10px;" >

<form action="../beanHandlers/budgetHandler.jsp" style="lead">

<a>Quarter</a></br> <select name="quarter">

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

</select></br>

<a>Year:</a></br> <select name="quarter">

<option>2014</option>

<option>2015</option>

<option>2016</option>

<option>2017</option>

<option>2018</option>

<option>2019</option>

<option>2020</option>

<option>2021</option>

<option>2022</option>

<option>2023</option>

<option>2024</option>

<option>2025</option>

<option>2026</option>

<option>2027</option>

</select></br>

<a>Set the quarter Budget :</a></br><input name="qbudget" type="number" min="1" required=""/></br></br>

<input type="submit" value="Submit Budget" />

</form>

</div>

<%--

Document : DisplayProject

Created on : May 1, 2015, 8:26:43 PM

Author : Athinodoros

--%>

<%@page import="java.io.File"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<%@page import="layer2.domain.interfaces.NamingConv" %>

<%@page import="layer2.domain.bean.UserInfo" %>

<%@page import="layer2.domain.bean.Comment" %>

<%@page import="layer2.domain.bean.Partner" %>

<%@page import="layer2.domain.bean.Project" %>

<%@page import="java.util.ArrayList" %>

<%@page import="java.io.File" %>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<% Project pr = (Project) session.getAttribute("thisProject"); %>

<% UserInfo ui = (UserInfo) session.getAttribute(NamingConv.USER); %>

<% ArrayList<Comment> cmLi = (ArrayList<Comment>) session.getAttribute("commentList");%>

</head>

<body>

<br/>

<br/>

<span style="font-size: 1.2em;">Project Title : </span> <span style="border-radius: 2px; background-color: rgba(255,255,255,0.5 );font-size: 1.1em;margin: 2px;padding: 2px;"><%= pr.getTitle()%></span><br/>

<span style="font-size: 1.2em;">Created By : </span> <span style="border-radius: 2px; background-color: rgba(255,255,255,0.5 );font-size: 1.1em;margin: 2px;padding: 2px;"><%= pr.getPartner().getCompanyName()%></span><br/><br/>

<div style="margin:3%;">

<span style="font-size: 1.1em; border-style: dashed;border-color: rgba(255,255,255,0.5);padding: 4px;">Starting Date :<%= pr.getSdate().toString()%></span> <span style="font-weight: 900;font-size: 1.5em;">-</span>

<span style="font-size: 1.1em; border-style: dashed;border-color: rgba(255,255,255,0.5);padding: 4px;">Finishing Date : <%= pr.getSdate().toString()%></span> <br/>

</div>

<% if (ui.getUrole().equals(NamingConv.ADMIN)) {%>

<span style="font-size: 1.2em; padding: 10px; margin-bottom: 20px;">Description : </span> <br/> <span style="background-color: rgba(255,255,255,0.5 );font-size: 1.1em; margin-top: 25px ;margin-left: 3%;padding: 3px; border-radius: 5px;border-style: solid;border-color: #31b0d5;"><%= pr.getDescription()%></span><br/>

<%} else if (ui.getUrole().equals(NamingConv.PARTNER) && pr.getStage().equals(NamingConv.PENDING)) {%>

<form action="UIServlet">

<input type="hidden" name="<%= NamingConv.COMMAND%>" value="<%= NamingConv.RELOAD\_MAIN%>" />

<input type="hidden" name="<%= NamingConv.MAINAREA%>" value="<%= NamingConv.SEE%>" />

<input type="hidden" name="<%= NamingConv.UPDATE\_DESCR%>" value="<%= NamingConv.UPDATE\_DESCR%>" />

<textarea name="newDescription" rows="4" cols="20"><%= pr.getDescription()%>

</textarea><br/>

<input type="hidden" name="thisProjectID" value="<%= pr.getProjectID()%>" />

<input type="submit" value="Update Description" />

</form>

<% } %>

<% if (ui.getUrole().equals(NamingConv.PARTNER) && pr.getStage().equals("approved") ) {%> <br/><br/>

<span style="font-size: 1.2em; margin: 30px;color: #101010; border-bottom: #000 dashed; padding: 1px; ">Upload a single file </span>

<form class="lead" style="margin: 2%;" action="UIServlet" method ="POST" enctype="multipart/form-data">

<input type="file" name="file" />

<input type="hidden" name="ProjectPOE" value="<%= pr.getProjectID() %>" />

<input type="hidden" name="command" value="<%= NamingConv.UPLOAD%>" />

<span style="background-color: lightcoral">Please make sure that the uploaded file does not have <br/>

a name that matches the name of an already existing file <br/>

</span>

<input type="submit" value="submit" />

<input type="email" name="" hidden=""/>

</form>

<% }%>

<% File[] fli = (File[])session.getAttribute("fli"); %>

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

<% if ( fli != null ) { %>

<Span style="font-size: 2em; text-decoration: #000 solid blink;">Proof of execution :</Span><br/><br/>

<%

for(File sfile : fli ){

String filepathBuild = pr.getPartner().getCompanyName()+"\\"+pr.getProjectID()+"\\"+sfile.getName(); %>

<div style="float: left; width: 220px; height: 220px;">

<a href="<%= filepathBuild %>"><img src="<%= filepathBuild %>" style=" width: 150px; height: 150px; border-radius: 5px;" /></a><br/>

<div style="text-align: center; width: 150px ; border-radius: 5px; height: 1.3em; background-color: rgba(120,80,200,0.5);color: blue;"> <%= sfile.getName() %> </div><br/>

</div>

<%}%>

<%}%>

</div>

<br/>

<div style="padding: 2%;">

<span style="font-size: 1.2em;">The Budget Asked is : </span> <span style="background-color: rgba(150,50,150,0.5 );font-size: 1.1em;margin: 2px;padding: 2px;"><%= String.valueOf(pr.getProjectBudget())%>DKK</span><br/>

</div> <br/>

<% if (ui.getUrole().equals("admin") && !pr.getStage().equals(NamingConv.APPROVED)) {%>

<form>

<input type="hidden" name="<%= NamingConv.COMMAND%>" value="<%= NamingConv.RELOAD\_MAIN%>" />

<input type="hidden" name="<%= NamingConv.MAINAREA%>" value="<%= NamingConv.SEE%>" />

<input type="hidden" name="<%= NamingConv.APPROVED%>" value="<%= NamingConv.APPROVED%>" />

<input type="hidden" name="thisProjectID" value="<%= pr.getProjectID()%>" />

<input type="submit" value="Approve" />

</form>

<% } else {%>

<span style="font-size: 2em;color: greenyellow; text-underline-position: auto">This Project has been approved </span>

<%}%>

<br/>

<br/>

<jsp:include page="/Forms/newComment.jsp" />

<% if (cmLi != null) { %>

<span style="font-size: 2em; font: sans-serif; color: #000 ">Comments</span>

<% for (Comment cm : cmLi) {%>

<% if (cm.getUser().getUrole().equals(NamingConv.ADMIN)) {%>

<div style=" border-top: #000; margin: 5px;"> <span style="background-color: lightgreen ; padding: 1px; border-radius: 15px;"><span style="font-weight: 800; font-size: 1.2em;">user:</span> <%= cm.getUser().getFirstname() + " " + cm.getUser().getLastname()%></span></div> <div style="margin: 10px;"> <span style="border-color: whitesmoke; background-color: rgba(255,255,255,0.6); border-radius: 40px; margin: 35px; padding: 10px;">Said : <%= cm.getComment()%></span> </div>

<% } %>

<% if (cm.getUser().getUrole().equals(NamingConv.PARTNER)) {%>

<div style=" border-top: #000; margin: 5px;"> <span style="background-color: lightskyblue ; padding: 1px; border-radius: 15px;"><span style="font-weight: 800; font-size: 1.2em;">user:</span> <%= cm.getUser().getFirstname() + " " + cm.getUser().getLastname()%></span></div> <div style="margin: 10px;"> <span style="border-color: whitesmoke; background-color: rgba(255,255,255,0.6); border-radius: 40px; margin: 35px; padding: 10px;">Said : <%= cm.getComment()%></span> </div>

<% } %>

<div style="background-color: rgba(100,100,100,0.4); border-top: #000;"> ${Scomment.getUser.getFirstname}</div> : ${Scomment.getComment}

<% } %>

<% }%>

<div class="col-md-12" style="height: 250px"></div>

</body>

</html>

<%--

Document : addUser

Created on : Apr 10, 2015, 12:15:40 PM

Author : Athinodoros

--%>

<%@page import="layer2.domain.interfaces.NamingConv"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<link href="bootstrap.min.css" rel="stylesheet">

<%@page import="layer2.domain.bean.UserInfo" %>

<%@page import="layer2.domain.bean.UserAuthentication" %>

<%@page import="layer2.domain.bean.Partner" %>

<%@page import="java.util.ArrayList" %>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<% ArrayList<Partner> list = (ArrayList<Partner>)request.getAttribute(NamingConv.PARTNER\_LIST); %>

<% session.setAttribute("list", list); %>

<% UserInfo userInfo = (UserInfo) session.getAttribute("userInfo"); %>

<% UserAuthentication userath = (UserAuthentication) session.getAttribute("userAth"); %>

<div class="form-user" style="line-height: 2em;" >

<% if (userInfo == null) { %>

<form class="lead" action="beanHandlers/userHandler.jsp" method="POST"> </br></br>

<a>First Name : </a></br><input type="text" name="firstname" value="" required=""/></br>

<a>Last Name : </a></br><input type="text" name="lastname" value="" required=""/></br>

<a>User Name : </a></br><input type="text" name="uname" value="" required=""/></br>

<a>Password : </a></br><input type="email" name="password" value="" required=""/></br>

<a>e-mail : </a></br><input type="text" name="email" value="" required=""/></br>

<a>User Type : </a></br> <select name="urole" required="">

<option>Partner</option>

<option value="ADMIN">DELL User</option>

<option>admin</option>

</select></br>

<a>Country : </a></br> <select name="country" required="" >

<option>DK</option>

<option>SE</option>

<option>FI</option>

<option>NO</option>

<option>IS</option>

</select></br>

<a>company : </a></br> <select name="companyName" required="" >

<option selected="" value="NULL"> NOTHING </option>

<% for (Partner singlePartner : list) { %>

<option> <%= singlePartner.getCompanyName() %></option>

<% } %>

</select>

</br></br>

<input type="submit" value="Create new User" />

</form>

<% }else { %>

<form class="lead" action="beanHandlers/userHandler.jsp" method="post" > </br></br>

<a>First Name : </a></br><input type="text" name="firstname" value="<%= userInfo.getFirstname() %>" required=""/></br>

<a>Last Name : </a></br><input type="text" name="lastname" value="<%= userInfo.getLastname()%>" required=""/></br>

<a>User Name : </a></br><input type="text" name="uname" value="<%= userath.getUname()%>" required=""/></br>

<a>Password : </a></br><input type="text" name="password" value="" required=""/></br>

<a>e-mail : </a></br><input type="text" name="email" value="<%= userath.getPassword() %>" required=""/></br>

<a>User Type : </a></br> <select name="urole" required="">

<option>Partner</option>

<option>DELL User</option>

<option>\*region manager</option>

</select></br>

<a>Country : </a></br> <select name="country" required="" >

<option selected=""></option>

<option>DK</option>

<option>SE</option>

<option>FI</option>

<option>NO</option>

<option>IS</option>

</select></br>

<a>company : </a></br> <select name="companyName" required="" >

<% for (Partner singlePartner : list) { %>

<option><%= singlePartner.getCompanyName() %></option>

<% } %>

</select>

</br></br>

<input type="submit" value="Create new User" />

</form>

<% } %>

</div>

<%--

Document : createCompanyForm

Created on : Apr 13, 2015, 10:32:39 AM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<link href="bootstrap.min.css" rel="stylesheet">

<div>

<h1>Register a company</h1>

<br>

<form class="lead" action="beanHandlers/createCompanyHandler.jsp" method="POST">

Company Name:

<input type="text" name="companyName" value="" required="">

<br>

Company ID:

<input type="number" name="companyID" value="" required="">

<br>

<input type="submit" value="Add" />

</form>

</div>

<%--

Document : fileUploadForm

Created on : Apr 23, 2015, 12:45:25 AM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<div>

<% session.setAttribute("command", NamingConv.UPLOAD); %>

<% request.setAttribute(NamingConv.MAINAREA, NamingConv.FAIL); %>

<form action="UIServlet" method ="POST" enctype="multipart/form-data">

<input type="file" name="file" />

<input type="hidden" name="command" value="<%= NamingConv.UPLOAD %>" />

<input type="submit" value="submit" />

<input type="email" name=""/>

</form>

</div>

<%--

Document : newComment

Created on : Apr 22, 2015, 11:35:47 PM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<link href="../bootstrap.min.css" rel="stylesheet" />

<div style="line-height: 2em; padding: 10px;">

<%@page import="layer2.domain.bean.Project" %>

<form style="lead" action="beanHandlers/commentHandler.jsp" >

<input type="text" name="commentID" value="" hidden/>

<input type="text" name="thisProjectID" value="${ thisProject.projectID }" hidden/>

<input type="text" name="projectID" value="${thisProject.projectID}" hidden/>

<input type="text" name="usedID" value="${UserInfo.userID}" hidden />

<a> Add a comment :</a></br> <textarea name="comment" rows="4" cols="30"></textarea>

</br> <input type="submit" value="Submit comment" /></br>

</form>

</div>

<%--

Document : newProject

Created on : Apr 22, 2015, 9:54:26 PM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="layer2.domain.interfaces.NamingConv"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html>

<link href="../bootstrap.min.css" rel="stylesheet" />

<div style="line-height: 2em; padding: 10px;">

<%@page import="layer2.domain.bean.Project" %>

<% Project newone = (Project) session.getAttribute("newProject"); %>

<% if (newone != null) {%>

<form style="" action="beanHandlers/newProjectsHandler.jsp" method="POST">

<a>Title: </a></br><input type="text" name="title" value="<%= newone.getTitle()%>" required=""/></br>

<a>Description : </a><br><textarea name="description" rows="4" cols="20" required=""><%= newone.getDescription()%>

</textarea></br>

<a>Starting date : </a></br> <input type="date" name="startdate" value="" required=""/></br>

<a>Finishing date : </a></br> <input type="date" name="findate" value="" required=""/></br>

<a>Budget request : </a></br><input type="number" name="projectBudget" value="" required=""/></br>

<input type="submit" value="Create Project" />

</form>

<% } else {%>

<form style="" action="beanHandlers/newProjectsHandler.jsp" method="POST">

<a>Title: </a></br><input type="text" name="title" value="" required=""/></br>

<a>Description : </a><br><textarea name="description" rows="4" cols="20" required="">

</textarea></br>

<a>Starting date : </a></br> <input type="date" name="startdate" value="" required=""/></br>

<a>Finishing date : </a></br> <input type="date" name="findate" value="" required=""/></br>

<a>Budget request : </a></br><input type="number" name="projectBudget" value="" required=""/></br>

<input type="submit" value="Create Project" />

</form>

<% } %>

</div>

<%--

Document : setBudget

Created on : Apr 29, 2015, 1:38:38 PM

Author : Alex

--%>

<%@page import="layer2.domain.interfaces.NamingConv"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="layer2.domain.bean.Budget" %>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<h1>Set the Budget</h1>

<form class="lead" action="beanHandlers/budgetHandler.jsp" method="POST" >

Year:

<input type="number" name="year" required="">

<br>

Quarter:

<input type="number" name="quarter" required="">

<br>

Budget:

<input type="number" name="budget" required="">

<br>

<input name="command" value=<%=NamingConv.SET\_BUDGET%> hidden/>

<input type="submit" value="Set" />

</form>

</body>

</html>

<%--

Document : header

Created on : Apr 15, 2015, 2:00:18 PM

Author : Athinodoros

--%>

<nav class="navbar navbar-inverse navbar-fixed-top" style="min-height: 5vh;">

<link href="../bootstrap.min.css" />

<div class="col-md-4" style="float: left; font-size: 2em; color: #ccc">Powered by DELL</div>

<div class="col-md-4" style="float: left; margin: auto;"></div>

<div class="col-md-4" style="float: left; padding-left: 15%;">

<form class="lead" action="UIServlet" >

<input type="hidden" name="command" value="log-out" />

<input class="MenuButtons" style="font-size: 1.2em; color: cyan; padding-top: 2px;" type="submit" value="Log out" />

</form>

</div>

</nav>

<%@page import="java.io.PrintWriter"%>

<%@page import="layer2.domain.interfaces.NamingConv"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html>

<html>

<head>

<style>

.MenuButtons{

padding: 1;

border-radius: 8px;

border: none;

background: rgba(255,255,255,0.2);

}

</style>

<style>li {background-color: 255,255,255,1 ; padding: 1px ; margin: 3%;}</style>

</head>

<body>

<%@page import="layer2.domain.\*" %>

<%@page import="layer2.domain.bean.UserInfo" %>

<% UserInfo user = (UserInfo) session.getAttribute("user"); %>

<% if (user.getUrole().equals(NamingConv.ADMIN) && true) { %>

<ul class="nav nav-sidebar">

<li><a href><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.PROJECT\_OVERVIEW %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Overview of all projects</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.PENDING\_PROJECTS %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">View pending projects</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.APPROVED\_PROJECTS %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">View approved projects</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.VIEW\_BUDGET %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">View budget</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.SET\_BUDGET %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Set Budget</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.CREATE\_USER %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Add User</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.CREATE\_COMPANY %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Add partner</button></form></a></li>

</ul>

<% } else if (user.getUrole().equals(NamingConv.PARTNER)) { %>

<ul class="nav nav-sidebar ">

<li><a href><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.PROJECT\_OVERVIEW %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Overview of all projects</button></form></a></li>

<li><a href="#"><form action="UIServlet" method="POST"><input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.CREATE\_PROJECT %>" hidden/><input name="command" value="reloadMain" hidden/><button class="MenuButtons" type="submit">Launch new project</button></form></a></li>

</ul>

<% } else if (user.getUrole().equals(NamingConv.PARTNER)) { %>

<ul class="nav nav-sidebar ">

<li class="active"><a href="#">Overview of projects <span class="sr-only">(current)</span></a></li>

<li><a href="#">POE Inspect</a></li>

</ul>

<%}%>

</body>

</html>

<%--

Document : Fail

Created on : Apr 29, 2015, 1:20:55 PM

Author : Bancho

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<h1>An error occurred while saving the new <%=request.getAttribute("type")%>. </h1>

</body>

</html>

<%--

Document : Success

Created on : Apr 29, 2015, 1:20:46 PM

Author : Bancho

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<h1>New <%=request.getAttribute("type")%> saved successfully. </h1>

<% String file = (String)request.getAttribute("type") ;

if (file.equals("POE")) {%>

<br/> Now you can go back and see your file. <br/>

If in any case you can't see the file refresh the page :)

<%} %>

</body>

</html>

<%--

Document : POEHandler

Created on : Apr 23, 2015, 1:00:50 AM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<link href="../bootstrap.min.css" />

<html>

<head>

</head>

<body background="http://cdn.wonderfulengineering.com/wp-content/uploads/2014/06/gear-wallpaper-14.jpg">

<div class="col-md-5"></div>

<div class="col-md-3" style="border-radius: 30px ; padding: 20px;background-color:rgba(255,255,255,0.5); margin-top: 20%; margin-bottom: 20%; margin-left: 30%; margin-right: 30%; ">

<a style="font-size: 1.5em; font-weight: 600 ;">

<!-- info strings here -->

<span style="color: #fff; background: rgba(255,255,255,0.2); border-radius: 20px; padding: 2px;"> Do you want to save the user?</span> <br/>

<jsp:useBean id="newFile" class="layer2.domain.bean.POE" scope="session" />

<jsp:setProperty name="newFile" property="\*" />

<%= newFile.getFile().toString()%>

<jsp:include page="../Forms/fileUploadForm.jsp" />

</a>

<div style="float: left">

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Save" />

</form>

</div>

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Go Back" />

</form>

</div>

</body>

</html>

<%--

Document : budgetHandler

Created on : Apr 23, 2015, 12:21:00 AM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<!DOCTYPE html>

<jsp:useBean id="newBudget" class="layer2.domain.bean.Budget" scope="session" />

<jsp:setProperty name="newBudget" property="\*" />

Year:

${newBudget.getQbudget()}

${newBudget.getQuarter()}

${newBudget.getQyear()}

<br>

<a>The budget has been set </a><br/>

<a>Do you want to save it?</a>

<form style="lead" action="../UIServlet" method="POST">

<input name="<%=NamingConv.COMMAND%>" value=<%=NamingConv.SET\_BUDGET%> hidden/>

<input type="submit" value="Set"/>

</form>

<form style="lead" action="../UIServlet" method="POST">

<input name="<%=NamingConv.COMMAND%>" value="<%=NamingConv.RELOAD\_MAIN%>" hidden/>

<input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.SET\_BUDGET %>" hidden/>

<input type="submit" value="Go Back" />

</form>

<%--

Document : commentHandler

Created on : Apr 22, 2015, 11:57:45 PM

Author : Athinodoros

--%>

<%@page import="layer2.domain.bean.UserInfo"%>

<%@page import="layer2.domain.bean.Project"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<link href="../bootstrap.min.css" />

<html>

<head>

</head>

<body background="http://cdn.wonderfulengineering.com/wp-content/uploads/2014/06/gear-wallpaper-14.jpg">

<div class="col-md-5"></div>

<div class="col-md-3" style="border-radius: 30px ; padding: 20px;background-color:rgba(255,255,255,0.5); margin-top: 20%; margin-bottom: 20%; margin-left: 30%; margin-right: 30%; ">

<a style="font-size: 1.5em; font-weight: 600 ;">

<%@page import="layer2.domain.interfaces.NamingConv" %>

<!-- info strings here -->

<span style="color: #fff; background: rgba(255,255,255,0.2); border-radius: 20px; padding: 2px;"> Do you want to save the user?</span> <br/>

<jsp:useBean id="inComment" class="layer2.domain.bean.Comment" scope="session" />

<jsp:setProperty name="inComment" property="\*" />

<%= inComment.getComment() %><br/>

</a>

<div style="float: left">

<!-- First buttons here -->

<% inComment.setProject((Project)session.getAttribute("thisProject")); %>

<% inComment.setUser((UserInfo)session.getAttribute(NamingConv.USER)); %>

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input type="hidden" name="<%= NamingConv.COMMAND %>" value="<%= NamingConv.RELOAD\_MAIN %>" />

<input type="hidden" name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.SEE %>" />

<input type="hidden" name="<%= NamingConv.NEWCOMMENT %>" value="<%= NamingConv.INCOMMENT %>" />

<input type="hidden" name="thisProjectID" value="${param.projectID}" />

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Save" />

</form>

</div>

<!-- buttons under here-->

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input type="hidden" name="thisProjectID" value="${param.projectID}" />

<input type="hidden" name="<%= NamingConv.COMMAND %>" value="<%= NamingConv.RELOAD\_MAIN %>" />

<input type="hidden" name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.SEE %>" />

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Go Back" />

</form>

</div>

</body>

</html>

<%--

Document : createCompanyHandler

Created on : Apr 28, 2015, 10:13:58 PM

Author : Алеx

--%>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html><html>

<head>

<jsp:useBean id="newPartner" class="layer2.domain.bean.Partner" scope="session" />

<jsp:setProperty name="newPartner" property="\*" />

<link href="../bootstrap.min.css" />

</head>

<body background="http://cdn.wonderfulengineering.com/wp-content/uploads/2014/06/gear-wallpaper-14.jpg">

<div class="col-md-5"></div>

<div class="col-md-3" style="border-radius: 30px ; padding: 20px;background-color:rgba(255,255,255,0.5); margin-top: 20%; margin-bottom: 20%; margin-left: 30%; margin-right: 30%; ">

<a style="font-size: 1.5em; font-weight: 600 ;">

<!-- info strings here -->

Company Name: <%=newPartner.getCompanyName()%><br/>

<span style="color: #fff; background: rgba(255,255,255,0.2); border-radius: 20px; padding: 2px;"> Do you want to save the Partner ?</span> <br/>

</a>

<div style="float: left">

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input name="<%=NamingConv.COMMAND%>" value=<%=NamingConv.CREATE\_COMPANY%> hidden/>

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Save" />

</form>

</div>

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input name="<%=NamingConv.COMMAND%>" value=<%=NamingConv.CREATE\_COMPANY %> hidden/>

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Go Back" />

</form>

</div>

</body>

</html>

<%--

Document : fileHandling

Created on : May 1, 2015, 3:47:04 PM

Author : Athinodoros

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<%= request.getPart("file").toString() %>

</body>

</html>

<%--

Document : newProjectsHandler

Created on : Apr 22, 2015, 9:55:46 PM

Author : Athinodoros

--%>

<!DOCTYPE html>

<link href="../bootstrap.min.css" />

<html>

<head>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<link href="../bootstrap.min.css" type="text/css" />

<jsp:useBean id="newProject" class="layer2.domain.bean.Project" scope="session" />

<jsp:setProperty name="newProject" property="\*" />

<%

session.setAttribute(NamingConv.SDATE, request.getParameter("startdate"));

session.setAttribute(NamingConv.FDATE, request.getParameter("findate"));

%>

</head>

<body background="http://cdn.wonderfulengineering.com/wp-content/uploads/2014/06/gear-wallpaper-14.jpg">

<div class="col-md-5"></div>

<div class="col-md-3" style="border-radius: 30px ; padding: 20px;background-color:rgba(255,255,255,0.5); margin-top: 20%; margin-bottom: 20%; margin-left: 30%; margin-right: 30%; ">

<a style="font-size: 1.5em; font-weight: 600 ;">

<!-- info strings here -->

Project Name: ${newProject.getTitle()} <br/>

Description: ${newProject.description} <br/>

Start date: <%=request.getParameter("startdate")%><br/>

Final date: <%=request.getParameter("findate")%><br/>

<span style="color: #fff; background: rgba(255,255,255,0.2); border-radius: 20px; padding: 2px;"> Do you want to save the Project?</span> <br/>

</a>

<div style="float: left">

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input name="command" value=<%=NamingConv.CREATE\_PROJECT%> hidden/>

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Save" />

</form>

</div>

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input name="<%=NamingConv.COMMAND%>" value="<%=NamingConv.RELOAD\_MAIN%>" hidden/>

<input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.CREATE\_PROJECT %>" hidden/>

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Go Back" />

</form>

</div>

</body>

</html>

<%--

Document : userHandler

Created on : Apr 15, 2015, 8:22:25 PM

Author : Athinodoros

--%>

<!DOCTYPE html>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<link href="../bootstrap.min.css" />

<html>

<head>

</head>

<body background="http://cdn.wonderfulengineering.com/wp-content/uploads/2014/06/gear-wallpaper-14.jpg">

<div class="col-md-5"></div>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<%@page import="layer2.domain.bean.Partner" %>

<%@page import="layer2.domain.bean.UserInfo" %>

<%@page import="java.util.ArrayList" %>

<jsp:useBean id="newUserInfo" class="layer2.domain.bean.UserInfo" scope="session" />

<% String NAME = (String) request.getParameter("companyName"); %>

<% ArrayList<Partner> li = (ArrayList<Partner>) session.getAttribute("list"); %>

<div class="col-md-3" style="border-radius: 30px ; padding: 20px;background-color:rgba(255,255,255,0.5); margin-top: 20%; margin-bottom: 20%; margin-left: 30%; margin-right: 30%; ">

<a style="font-size: 1.5em; font-weight: 600 ;">

<% Partner newPartnerInTheBean = new Partner(); %>

<% for (Partner one : li) {

if (one.getCompanyName().equals(NAME)) {

newPartnerInTheBean = one;

break;

}

}%>

<% newUserInfo.setCompany(newPartnerInTheBean); %>

<!-- creating the userinfo bean-->

<jsp:setProperty name="newUserInfo" property="\*" />

<jsp:useBean id="newUserAth" class="layer2.domain.bean.UserAuthentication" scope="session" />

<jsp:setProperty name="newUserAth" property="\*" />

<% session.setAttribute("companyName", request.getParameter("company"));%>

First Name : <%= newUserInfo.getFirstname()%><br/>

Last Name :<%= newUserInfo.getLastname()%><br/>

Country : <%= newUserInfo.getCountry()%><br/>

Company : <%= NAME%> <br/>

User Role : <%= newUserInfo.getUrole()%><br/>

UserName : <%= newUserAth.getUname()%><br/>

Password : <%= newUserAth.getPassword()%><br/>

e-mail : <%= newUserAth.getEmail()%><br/>

<br/>

<br/>

<span style="color: #fff; background: rgba(255,255,255,0.2); border-radius: 20px; padding: 2px;"> Do you want to save the user?</span> <br/>

</a>

<div style="float: left">

<form class="lead" style="margin: 4px;" action="../UIServlet">

<input type="hidden" name="<%=NamingConv.MAINAREA%>" value="<%= NamingConv.CREATE\_USER %>" />

<% session.setAttribute(NamingConv.COMMAND, NamingConv.CREATE\_USER); %>

<input style="padding: 18px; border-radius: 15px; border-color: rgba(200,200,200,0.8)" type="submit" value="Save" />

</form>

</div>

<form class="lead" style="margin: 4px; border-color: rgba(200,200,200,0.8)" action="../UIServlet">

<% session.setAttribute("userInfo", newUserInfo); %>

<% session.setAttribute("userAth", newUserAth);%>

<input type="hidden" name="<%= NamingConv.COMMAND %>" value="<%= NamingConv.RELOAD\_MAIN %>" />

<input type="hidden" name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.CREATE\_USER %>" />

<input style="padding: 18px; border-radius: 15px;" type="submit" value="GoBack" />

</form>

</div>

</body>

</html>

<%--

Document : DashBoardPicker

Created on : Apr 11, 2015, 8:16:06 PM

Author : Athinodoros

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html>

--%>

<%@page import="layer2.domain.interfaces.NamingConv"%>

<link href="bootstrap.min.css" rel="stylesheet">

<div class="col-md-10">

<% String pagepoint = (String) request.getAttribute(NamingConv.MAINAREA);

if (pagepoint == NamingConv.VIEW\_BUDGET) { %>

<jsp:include page="Forms/Budget.jsp"/>

<%} else if (pagepoint == NamingConv.CREATE\_PROJECT) { %>

<jsp:include page="Forms/newProject.jsp"/>

<% } else if (pagepoint == NamingConv.CREATE\_COMPANY) { %>

<jsp:include page="Forms/createCompanyForm.jsp" />

<% } else if (pagepoint == NamingConv.CREATE\_USER) { %>

<jsp:include page="Forms/addUser.jsp" />

<% } else if (pagepoint == NamingConv.PROJECT\_OVERVIEW) { %>

<jsp:include page="ProjectOverview.jsp" />

<% } else if (pagepoint == NamingConv.SUCCESS) { %>

<jsp:include page="UserFeedback/Success.jsp" />

<% } else if (pagepoint == NamingConv.FAIL) { %>

<jsp:include page="UserFeedback/Fail.jsp" />

<% } else if (pagepoint == NamingConv.SEE) { %>

<jsp:include page="Forms/DisplayProject.jsp" />

<% } else { %>

<p>no jsp was loaded </p>

<% }%>

</div>

<%--

Document : Dashboard

Created on : Apr 11, 2015, 8:16:06 PM

Author : Athinodoros

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Dell rules</title>

<!-- Bootstrap core CSS -->

<link href="bootstrap.min.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="dashboard.css" rel="stylesheet">

<link href="bootstrap.min.css" rel="stylesheet">

</head>

<body style="height: 100%">

<div style="background-image: url('http://www.hdwallpapersimages.com/wp-content/uploads/2013/11/Dell-Computer-Wallpaper.jpg'); background-size: contain;" >

<%-- Top area header etc. --%>

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

<jsp:include page="PageComponents/header.jsp" />

</div>

<%-- left menu item --%>

<div style="background-color: rgba(255,255,255,0.5);min-height: 178.5vh; max-height: 220vh" class="col-md-2" >

<a style="font-size: 2em; font-style: italic; ">DashBoard</a> <jsp:include page="PageComponents/menus.jsp"/>

</div>

<%-- dashboard main area --%>

<div style="height: 90vh; max-height: 100vh; margin-bottom: 10px;" class="col-md-10">

<jsp:include page="DashBoardPicker.jsp" />

</div>

<%-- footer area --%>

<div class="navbar-fixed-bottom col-md-12" style="background-color: #31b0d5;" >

<a style="font-size: 2em;color: black;">CopyRight Group\_05\_2\_semester </a>

</div>

</div>

</body>

</html>

<%--

Document : ProjectOverview

Created on : Apr 27, 2015, 11:10:29 PM

Author : Alex

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<!DOCTYPE html>

<html>

<head>

<% request.removeAttribute("comment"); %>

<% session.removeAttribute("comment"); %>

<% session.removeValue("comment"); %>

</head>

<body>

<h1>Project overview</h1><br/>

<div class="" style="padding: auto; width: 100% ;">

<table class="table-responsive table table-condensed table-striped table-bordered" style="min-width: 900px ; background-color: rgba(255,255,255,0.5);">

<tr style=" width: auto;">

<th>Title</th>

<th>Company Name</th>

<th style="padding-left: 10%;"><form action="UIServlet" method="POST"> <input type="hidden" name="command" value="reloadMain" /><input type="hidden" name="order" value="BUDGET" /> <input type="hidden" name="<%= NamingConv.MAINAREA %>" value="<%= request.getAttribute(NamingConv.MAINAREA) %>" /> <input class="MenuButtons" type="submit" value="Project Budget" /> </form></th>

<th>Start Date</th>

<th>End Date</th>

<th><form action="UIServlet" method="POST"> <input type="hidden" name="command" value="reloadMain" /><input type="hidden" name="order" value="BUDGET" /> <input type="hidden" name="<%= NamingConv.MAINAREA %>" value="<%= request.getAttribute(NamingConv.MAINAREA) %>" /> <input class="MenuButtons" type="submit" value="Stage" /> </form></th>

<th> View More</th>

</tr>

<c:if test="${projects != null}" >

<c:forEach var="Project" items="${projects}">

<tr>

<td>${Project.getTitle()}</td>

<td>${Project.getPartner().getCompanyName()}</td>

<td style="text-align: right;"><div style="padding-right: 40%;">${Project.getProjectBudget()} DKK</div></td>

<td>${Project.getSdate()}</td>

<td>${Project.getFdate()}</td>

<c:if test="${ Project.stage == 'pending' }" >

<td style="color: red;">${Project.getStage()}</td>

</c:if>

<c:if test="${ Project.stage == 'approved' }" >

<td style="color: blue;">${Project.getStage()}</td>

</c:if>

<c:if test="${ Project.stage == 'finished' }" >

<td style="color: chartreuse; font-weight: 900;">${Project.getStage()}</td>

</c:if>

<td><a href="#"><form action="UIServlet" method="POST">

<input name="<%= NamingConv.MAINAREA %>" value="<%= NamingConv.SEE %>" hidden/>

<input name="thisProjectID" value="${ Project.projectID }" hidden/>

<input name="<%= NamingConv.COMMAND %>" value="<%= NamingConv.RELOAD\_MAIN %>" hidden/><button class="MenuButtons" type="submit">

View More</button></form></a></td>

</tr>

</c:forEach>

</c:if>

</table>

</div>

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<%@page import="layer2.domain.interfaces.NamingConv" %>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<link rel="icon" href="../../favicon.ico">

<title>Cover Template for Bootstrap</title>

<!-- Bootstrap core CSS -->

<link href="bootstrap.min.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="cover.css" rel="stylesheet">

<!-- Just for debugging purposes. Don't actually copy these 2 lines! -->

<!--[if lt IE 9]><script src="../../assets/js/ie8-responsive-file-warning.js"></script><![endif]-->

<script src="../../assets/js/ie-emulation-modes-warning.js"></script>

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>

<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body>

<div class="site-wrapper">

<div class="site-wrapper-inner">

<div class="cover-container">

<div class="masthead clearfix">

<div class="inner">

<h3 class="masthead-brand">Powered by DELL</h3>

<nav>

<ul class="nav masthead-nav">

<li class="active"><a href="#">Home</a></li>

<li><a href="#">Features</a></li>

<li><a href="#">Contact</a></li>

</ul>

</nav>

</div>

</div>

<div class="inner cover">

<h1 class="cover-heading">Welcome to DELL's new web app</h1>

<form class="lead" action="UIServlet" method="POST">

Username:<br/>

<input class ="input" type="text" name = "<%=NamingConv.USERNAME%>" required/><br/><br/>

Password:<br/>

<input class="input" type='password' name = "<%=NamingConv.PASSWORD%>" required/><br/><br/>

<input name="<%=NamingConv.COMMAND%>" value="<%=NamingConv.LOG\_IN%>" hidden/>

<input name="<%=NamingConv.MAINAREA%>" value="<%=NamingConv.PROJECT\_OVERVIEW%>" hidden/>

<button class = "input" type = "submit" name = "LogIn" value="LogIn">Log in</button>

</form>

</div>

<div class="mastfoot">

<div class="inner">

<p>Cover template for <a href="http://getbootstrap.com">Bootstrap</a>, by <a href="https://twitter.com/mdo">@mdo</a>.</p>

</div>

</div>

</div>

</div>

</div>

<!-- Bootstrap core JavaScript

================================================== -->

<!-- Placed at the end of the document so the pages load faster -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>

<script src="../../dist/js/bootstrap.min.js"></script>

<script src="../../assets/js/docs.min.js"></script>

<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->

<script src="../../assets/js/ie10-viewport-bug-workaround.js"></script>

</body>

</html>

Paokpaok

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer1.presentation;

import java.io.File;

import java.io.FileOutputStream;

import java.io.FileWriter;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import static java.sql.Types.INTEGER;

import java.util.ArrayList;

import java.util.Calendar;

import java.util.Collection;

import java.util.Collections;

import java.util.Comparator;

import java.util.Iterator;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletContext;

import javax.servlet.ServletException;

import javax.servlet.annotation.MultipartConfig;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import javax.servlet.http.Part;

import layer2.domain.Controller;

import layer2.domain.bean.Comment;

import layer2.domain.bean.POE;

import layer2.domain.bean.Partner;

import layer2.domain.bean.Project;

import layer2.domain.bean.UserAuthentication;

import layer2.domain.bean.UserInfo;

import layer2.domain.interfaces.NamingConv;

import sun.rmi.server.Dispatcher;

//comment

/\*\*

\*

\* @author Bancho

\*/

@WebServlet(name = "UIServlet", urlPatterns = {"/UIServlet"})

@MultipartConfig

public class UIServlet extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

\* methods.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

if (ctrl == null) {

ctrl = new Controller();

session.setAttribute(NamingConv.CONTROLLER, ctrl);

}

String command = (String) request.getParameter(NamingConv.COMMAND);

RequestDispatcher dispatcher;

if (command == null) {

command = (String) session.getAttribute(NamingConv.COMMAND);

}

switch (command) {

case NamingConv.LOG\_IN:

boolean status = validateCredentials(request, response);

if (status) {

viewProjects(request, response);

} else {

dispatcher = request.getRequestDispatcher("index.jsp");

dispatcher.forward(request, response);

}

break;

case NamingConv.LOG\_OUT:

logOut(request, response);

break;

case NamingConv.UPLOAD:

upload(request, response);

break;

case NamingConv.CREATE\_USER:

createUser(request, response);

break;

case NamingConv.CREATE\_COMPANY:

createPartner(request, response);

break;

case NamingConv.CREATE\_PROJECT:

createProject(request, response);

break;

case NamingConv.SET\_BUDGET:

//not done

break;

case NamingConv.RELOAD\_MAIN:

String mainArea = (String) request.getParameter(NamingConv.MAINAREA);

if (mainArea == null) {

mainArea = (String) session.getAttribute(NamingConv.MAINAREA);

}

switch (mainArea) {

case NamingConv.SET\_BUDGET:

request.setAttribute(NamingConv.MAINAREA, NamingConv.SET\_BUDGET);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

case NamingConv.VIEW\_BUDGET:

request.setAttribute(NamingConv.MAINAREA, NamingConv.VIEW\_BUDGET);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

case NamingConv.CREATE\_COMPANY:

request.setAttribute(NamingConv.MAINAREA, NamingConv.CREATE\_COMPANY);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

case NamingConv.CREATE\_USER:

request.setAttribute(NamingConv.MAINAREA, NamingConv.CREATE\_USER);

request.setAttribute(NamingConv.PARTNER\_LIST, ctrl.getAllPartners());

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

case NamingConv.CREATE\_PROJECT:

request.setAttribute(NamingConv.MAINAREA, NamingConv.CREATE\_PROJECT);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

case NamingConv.SEE:

String incom = request.getParameter(NamingConv.NEWCOMMENT);

String app = request.getParameter(NamingConv.APPROVED);

String updes = request.getParameter(NamingConv.UPDATE\_DESCR);

String path = request.getServletContext().getRealPath("/");

File f = new File(path + ctrl.getProject(Integer.parseInt(request.getParameter("thisProjectID"))).getPartner().getCompanyName() + "\\" + request.getParameter("thisProjectID"));

File[] fileli = f.listFiles();

session.setAttribute("fli", fileli);

session.setAttribute("id", request.getParameter("thisProjectID"));

if (incom != null) {

if (saveComment(request, response, (Comment) session.getAttribute("inComment"))) {

session.removeAttribute("inComment");

openOneProject(request, response);

} else {

request.setAttribute(NamingConv.TYPE, "Comment");

request.setAttribute(NamingConv.RELOAD\_MAIN, NamingConv.FAIL);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

}

}

if (app != null && app.equals(NamingConv.APPROVED)) {

Project tempPr = ctrl.getProject(Integer.valueOf(request.getParameter("thisProjectID")));

tempPr.setStage(app);

if (ctrl.editProject(tempPr)) {

openOneProject(request, response);

} else {

request.setAttribute(NamingConv.TYPE, "Project approval");

request.setAttribute(NamingConv.RELOAD\_MAIN, NamingConv.FAIL);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

}

}

if (updes != null && updes.equals(NamingConv.UPDATE\_DESCR)) {

Project tempPr = ctrl.getProject(Integer.valueOf(request.getParameter("thisProjectID")));

ctrl.createComment(new Comment(1, tempPr, (UserInfo) session.getAttribute(NamingConv.USER), tempPr.getDescription() + " <span style=\"font-size: 1.1em; font-weight: 600;\"> Was changed to : </span>" + request.getParameter("newDescription")));

tempPr.setDescription(request.getParameter("newDescription"));

if (ctrl.editProject(tempPr)) {

openOneProject(request, response);

} else {

request.setAttribute(NamingConv.TYPE, "Project new Description");

request.setAttribute(NamingConv.RELOAD\_MAIN, NamingConv.FAIL);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

}

}

openOneProject(request, response);

break;

case NamingConv.PROJECT\_OVERVIEW:

case NamingConv.PENDING\_PROJECTS:

case NamingConv.APPROVED\_PROJECTS:

viewProjects(request, response);

break;

default:

request.setAttribute(NamingConv.MAINAREA, NamingConv.SEE);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

}

break;

default:

request.setAttribute(NamingConv.MAINAREA, NamingConv.SEE);

dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

break;

}

}

private boolean validateCredentials(HttpServletRequest request, HttpServletResponse response) {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

String username = request.getParameter(NamingConv.USERNAME);

String password = request.getParameter(NamingConv.PASSWORD);

boolean logInSuccessful = ctrl.validateCredentials(username, password);

if (logInSuccessful) {

UserAuthentication ua = ctrl.getUserAuthentication(username);

UserInfo user = ua.getUserInfo();

session.setAttribute(NamingConv.USER, user);

}

return logInSuccessful;

}

private boolean saveComment(HttpServletRequest request, HttpServletResponse response, Comment cm) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

return ctrl.createComment(cm);

}

private void openOneProject(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

UserInfo currentUser = (UserInfo) session.getAttribute(NamingConv.USER);

ArrayList<Project> allProjects = (ArrayList<Project>) ctrl.getAllProjects();

String thisProject = request.getParameter("thisProjectID");

ArrayList<Comment> allComments = (ArrayList<Comment>) ctrl.getAllComments(Integer.valueOf(thisProject));

session.setAttribute("commentList", allComments);

session.setAttribute("thisProject", ctrl.getProject(Integer.valueOf(thisProject)));

request.setAttribute(NamingConv.MAINAREA, NamingConv.SEE);

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

private void logOut(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

ctrl.closeConnection();

session.invalidate();

RequestDispatcher dispatcher = request.getRequestDispatcher("index.jsp");

dispatcher.forward(request, response);

}

private void createProject(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

UserInfo currentUser = (UserInfo) session.getAttribute(NamingConv.USER);

Project newProject = (Project) session.getAttribute(NamingConv.NEW\_PROJECT);

newProject.setPartner(currentUser.getCompany());

newProject.setStage(NamingConv.PENDING);

String sdate = (String) session.getAttribute(NamingConv.SDATE);

String fdate = (String) session.getAttribute(NamingConv.FDATE);

newProject.setSdate(string2date(sdate));

newProject.setFdate(string2date(fdate));

boolean status = ctrl.createProject(newProject);

if (status) {

request.setAttribute(NamingConv.MAINAREA, NamingConv.SUCCESS);

} else {

request.setAttribute(NamingConv.MAINAREA, NamingConv.FAIL);

}

request.setAttribute(NamingConv.TYPE, "project");

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

private void viewProjects(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

request.setAttribute(NamingConv.MAINAREA, NamingConv.PROJECT\_OVERVIEW);

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

UserInfo currentUser = (UserInfo) session.getAttribute(NamingConv.USER);

ArrayList<Project> allProjects = (ArrayList<Project>) ctrl.getAllProjects();

String order = request.getParameter("order");

if (currentUser.getUrole().equals(NamingConv.ADMIN)) {

String requsted = request.getParameter(NamingConv.MAINAREA);

switch (requsted) {

case NamingConv.PROJECT\_OVERVIEW:

request.setAttribute(NamingConv.PROJECTS, allProjects);

break;

case NamingConv.PENDING\_PROJECTS:

ArrayList<Project> pendingProjects = new ArrayList<Project>();

for (Project project : allProjects) {

if (project.getStage().equals(NamingConv.PENDING)) {

pendingProjects.add(project);

}

}

if (order != null) {

if (order.equals("BUGET")) {

Collections.sort(pendingProjects, new Comparator<Project>() {

@Override

public int compare(Project o1, Project o2) {

return o1.getProjectBudget() - o2.getProjectBudget();

}

});

}

}

request.setAttribute(NamingConv.PROJECTS, pendingProjects);

break;

case NamingConv.APPROVED\_PROJECTS:

ArrayList<Project> approvedProjects = new ArrayList<Project>();

for (Project project : allProjects) {

if (project.getStage().equals(NamingConv.PENDING) == false) {

approvedProjects.add(project);

}

}

if (order != null) {

if (order.equals("BUGET")) {

Collections.sort(approvedProjects, new Comparator<Project>() {

@Override

public int compare(Project o1, Project o2) {

return o2.getProjectBudget() - o1.getProjectBudget();

}

});

}

}

request.setAttribute(NamingConv.PROJECTS, approvedProjects);

break;

}

} else if (currentUser.getUrole().equals(NamingConv.PARTNER)) {

ArrayList<Project> onlyPartnerProjects = new ArrayList<Project>();

for (Project project : allProjects) {

if (project.getPartner().getCompanyName().equals(currentUser.getCompany().getCompanyName())) {

onlyPartnerProjects.add(project);

}

}

request.setAttribute(NamingConv.PROJECTS, onlyPartnerProjects);

}

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

private void createPartner(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

Partner newPartner = (Partner) session.getAttribute(NamingConv.NEW\_PARTNER);

boolean status = ctrl.createPartner(newPartner);

if (status) {

request.setAttribute(NamingConv.MAINAREA, NamingConv.SUCCESS);

} else {

request.setAttribute(NamingConv.MAINAREA, NamingConv.FAIL);

}

request.setAttribute(NamingConv.TYPE, "partner");

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

private void createUser(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

UserAuthentication newUserAth = (UserAuthentication) session.getAttribute(NamingConv.NEW\_USER\_ATH);

UserInfo newUserInfo = (UserInfo) session.getAttribute(NamingConv.NEW\_USER\_INFO);

boolean status = ctrl.createUserInfo(newUserInfo);

newUserAth.setUserInfo(newUserInfo);

boolean status1 = ctrl.createUserAth(newUserAth);

if (status && status1) {

request.setAttribute(NamingConv.MAINAREA, NamingConv.SUCCESS);

} else {

request.setAttribute(NamingConv.MAINAREA, NamingConv.FAIL);

}

request.setAttribute(NamingConv.TYPE, "user");

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

private java.util.Date string2date(String date) {

Calendar cal = Calendar.getInstance();

String[] splitted = date.split("-");

int year = Integer.parseInt(splitted[0]);

int month = Integer.parseInt(splitted[1]) - 1;

int day = Integer.parseInt(splitted[2]);

cal.set(year, month, day);

return cal.getTime();

}

private void upload(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession();

Controller ctrl = (Controller) session.getAttribute(NamingConv.CONTROLLER);

String path = request.getServletContext().getRealPath("/");

Project pr = (Project) session.getAttribute("thisProject");

File f = new File(path + pr.getPartner().getCompanyName());

f.mkdir();

f = new File(f + "\\" + session.getAttribute("id"));

f.mkdir();

POE poe = new POE();

Part part = request.getPart("file");

poe.setPrefix(part.getHeader("content-type"));

poe.setInStream(part.getInputStream());

poe.setFileName(part.getSubmittedFileName());

poe.setProject(ctrl.getProject(1));

//response.setContentType(contentType);

//////FilenameUtils.getExtension("file or file path+file");/////////////////

File file = new File(f + "\\" + poe.getFileName());

OutputStream out = new FileOutputStream(file);

byte[] buffer = new byte[1024];

int count = 0;

do {

count = poe.getInStream().read(buffer);

out.write(buffer, 0, count);

} while (count == 1024);

request.setAttribute(NamingConv.MAINAREA, NamingConv.SUCCESS);

request.setAttribute(NamingConv.COMMAND, NamingConv.RELOAD\_MAIN);

request.setAttribute(NamingConv.TYPE, "POE");

RequestDispatcher dispatcher = request.getRequestDispatcher("Dashboard.jsp");

dispatcher.forward(request, response);

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain;

import java.io.FileNotFoundException;

import java.util.Collection;

import layer2.domain.bean.Comment;

import layer2.domain.bean.POE;

import layer2.domain.bean.Partner;

import layer2.domain.bean.Project;

import layer2.domain.bean.UserAuthentication;

import layer2.domain.bean.UserInfo;

import layer3.dataSource.DBFacade;

/\*\*

\*

\* @author Bancho

\*/

public class Controller {

private DBFacade dbf;

public Controller() {

dbf = DBFacade.getInstance();

}

public boolean createProject(Project newProject){

return dbf.createProject(newProject);

}

public boolean createPartner(Partner partner){

return dbf.createPartner(partner);

}

public boolean editProject(Project project) {

return dbf.editProject(project);

}

public Project getProject(int projectID){

return dbf.getProject(projectID);

}

public Collection<Project> getAllProjects(){

return dbf.getAllProjects();

}

public String getFileDirec(int projectid){

Project project = dbf.getProject(projectid);

String companyName = project.getPartner().getCompanyName();

return companyName + projectid;

}

public Collection<Partner> getAllPartners(){

return dbf.getAllPartners();

}

public boolean createComment(Comment cm){

return dbf.createComment(cm);

}

public Collection<Comment> getAllComments(int ProjectID){

return dbf.getAllCommentForproject(ProjectID);

}

public boolean createPOE(POE poe) throws FileNotFoundException{

return dbf.createPOE(poe);

}

public boolean createUserInfo(UserInfo ui){

return dbf.createUserInfo(ui);

}

public UserInfo getUserInfo(int userid){

return dbf.getUserInfo(userid);

}

public UserAuthentication getUserAuthentication(String username){

return dbf.getUserAuthentication(username);

}

public boolean createUserAth(UserAuthentication ua){

return dbf.createUserAth(ua);

}

public boolean validateCredentials(String username, String password){

UserAuthentication uAth = dbf.getUserAuthentication(username);

if (uAth == null) {

return false;

}

return uAth.getPassword().equals(password);

}

public void closeConnection(){

dbf.closeConnection();

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

/\*\*

\*

\* @author bo

\*/

public class Budget {

private int quarter; // maybe this shoud be a special type fx. num.?

private int qyear; // maybe this should be a data and not integer?

private int qbudget;

public Budget(int quarter, int qyear, int budget) {

this.quarter = quarter;

this.qyear = qyear;

this.qbudget = budget;

}

public Budget() {

}

public int getQuarter() {

return quarter;

}

public void setQuarter(int quarter) {

this.quarter = quarter;

}

public int getQyear() {

return qyear;

}

public void setQyear(int qyear) {

this.qyear = qyear;

}

public int getQbudget() {

return qbudget;

}

public void setQbudget(int budget) {

this.qbudget = budget;

}

@Override

public String toString() {

return "Budget{" + "quarter=" + quarter + ", qyear=" + qyear + ", qbudget=" + qbudget + '}';

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

/\*\*

\*

\* @author Athinodoros

\*/

public class Comment {

private int commentID;

private Project project;

private UserInfo user;

private String comment;

public Comment() {

}

public Comment(int commentID, Project project, UserInfo user, String comment) {

this.commentID = commentID;

this.project = project;

this.user = user;

this.comment = comment;

}

public Comment(Comment c) {

this.commentID = c.commentID;

this.project = c.project;

this.user = c.user;

this.comment = c.comment;

}

public int getCommentID() {

return commentID;

}

public void setCommentID(int commentID) {

this.commentID = commentID;

}

public Project getProject() {

return project;

}

public void setProject(Project projectID) {

this.project = projectID;

}

public UserInfo getUser() {

return user;

}

public void setUser(UserInfo userID) {

this.user = userID;

}

public String getComment() {

return comment;

}

public void setComment(String comment) {

this.comment = comment;

}

@Override

public String toString() {

return "Comment{" + "commentID=" + commentID + ", project=" + project + ", user=" + user + ", comment=" + comment + '}';

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

import java.io.File;

import java.io.FileOutputStream;

import java.io.InputStream;

import java.io.OutputStream;

/\*\*

\*

\* @author Athinodoros

\*/

public class POE {

private int POEID;

private long size;

private Project project;

private String fileName;

private String prefix;

private InputStream inStream;

private OutputStream fileOut;

//new costructors ->

public POE() {

}

public POE(int POEID, Project project, String fileName, String prefix, File file, InputStream inStream, InputStream fileIn, FileOutputStream fileOut) {

this.POEID = POEID;

this.project = project;

this.fileName = fileName;

this.prefix = prefix;

this.inStream = inStream;

this.fileOut = fileOut;

}

public POE(POE p){

this.POEID = p.POEID;

this.project = p.project;

this.fileName = p.fileName;

this.prefix = p.prefix;

this.fileOut = p.fileOut;

this.inStream = p.inStream;

}

public long getSize() {

return size;

}

public void setSize(long size) {

this.size = size;

}

public OutputStream getFileOut() {

return fileOut;

}

public void setFileOut(OutputStream fileOut) {

this.fileOut = fileOut;

}

// Getters and setters ->

public InputStream getInStream() {

return inStream;

}

public void setInStream(InputStream inStream) {

this.inStream = inStream;

}

public Project getProject() {

return project;

}

public void setProject(Project project) {

this.project = project;

}

public String getFilePath() {

return fileName;

}

public void setFilePath(String filePath) {

this.fileName = filePath;

}

// ->

@Override

public String toString() {

return super.toString(); //To change body of generated methods, choose Tools | Templates.

}

public int getPOEID() {

return POEID;

}

public void setPOEID(int POEID) {

this.POEID = POEID;

}

public String getFileName() {

return fileName;

}

public void setFileName(String fileName) {

this.fileName = fileName;

}

public String getPrefix() {

return prefix;

}

public void setPrefix(String filename) {

this.prefix = filename;

}

} // End of Class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

/\*\*

\*

\* @author bo

\*/

public class Partner {

private String companyName; //primary key

private int companyID; // -> project

public Partner() {}

public Partner(String companyName, int companyID) {

this.companyName = companyName;

this.companyID = companyID;

}

public Partner(Partner p) {

this.companyName = p.companyName;

this.companyID = companyID;

}

public String getCompanyName() {

return companyName;

}

public void setCompanyName(String companyName) {

this.companyName = companyName;

}

public int getCompanyID() {

return companyID;

}

public void setCompanyID(int companyID) {

this.companyID = companyID;

}

@Override

public String toString() {

return "Partner{" + "companyName=" + companyName + ", companyID=" + companyID + '}';

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

import java.util.Date;

/\*\*

\*

\* @author Bancho , athinodwros

\*/

public class Project {

private int projectID; // primary key

private Partner partner; // foreign key (companyName)

private String title;

private String description;

private String stage;

private Date sdate;

private Date fdate;

private int projectBudget;

// private boolean hasPOE;

// private String comments;

// DateFormat format = new SimpleDateFormat("yyyy,mm,dd", Locale.ENGLISH);

// DateFormat df = new SimpleDateFormat(mydate);

// this.sdate = format.parse(sdate.toString());

// DateFormat format = new SimpleDateFormat("yyyy,mm,dd", Locale.ENGLISH);

// DateFormat df = new SimpleDateFormat(mydate);

// this.fdate = format.parse(fdate.toString());

public Project(){}

public Project(Project p) {

this.projectID = p.projectID;

this.partner = p.partner;

this.title = p.title;

this.description = p.description;

this.stage = p.stage;

this.sdate = p.sdate;

this.fdate = p.fdate;

this.projectBudget = p.projectBudget;

}

public Project(int projectID, Partner partner, String title, String description, String stage, Date sdate, Date fdate, int projectBudget) {

this.projectID = projectID;

this.partner = partner;

this.title = title;

this.description = description;

this.stage = stage;

this.sdate = sdate;

this.fdate = fdate;

this.projectBudget = projectBudget;

}

public int getProjectID() {

return projectID;

}

public void setProjectID(int projectID) {

this.projectID = projectID;

}

public Partner getPartner() {

return partner;

}

public void setPartner(Partner partner) {

this.partner = partner;

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public String getStage() {

return stage;

}

public void setStage(String stage) {

this.stage = stage;

}

public Date getSdate() {

return sdate;

}

public void setSdate(Date sdate) {

this.sdate = sdate;

}

public Date getFdate() {

return fdate;

}

public void setFdate(Date fdate) {

this.fdate = fdate;

}

public int getProjectBudget() {

return projectBudget;

}

public void setProjectBudget(int projectBudget) {

this.projectBudget = projectBudget;

}

@Override

public String toString() {

return "Project{" + "projectID=" + projectID + ", partner=" + partner + ", title=" + title + ", description=" + description + ", stage=" + stage + ", sdate=" + sdate + ", fdate=" + fdate + ", projectBudget=" + projectBudget + '}';

}

} // End of class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

/\*\*

\*

\* @author bo

\*/

public class UserAuthentication {

private UserInfo userInfo; // foreign key

private String uname; // primary key (userID)

private String password;

private String email;

public UserAuthentication() {}

public UserAuthentication(UserInfo userInfo, String uname, String password, String email) {

this.userInfo = userInfo;

this.uname = uname;

this.password = password;

this.email = email;

}

public UserAuthentication(UserAuthentication ua) {

this.userInfo = ua.userInfo;

this.uname = ua.uname;

this.password = ua.password;

this.email = ua.email;

}

public UserInfo getUserInfo() {

return userInfo;

}

public void setUserInfo(UserInfo userInfo) {

this.userInfo = userInfo;

}

public String getUname() {

return uname;

}

public void setUname(String uname) {

this.uname = uname;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

@Override

public String toString() {

return "UserAutentication{" + "userInfo=" + userInfo + ", uname=" + uname + ", password=" + password + ", email=" + email + '}';

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.bean;

/\*\*

\*

\* @author bo

\*/

public class UserInfo {

private int userID; // primary key

private String firstname;

private String lastname;

private String country;

private Partner partner;

private String urole; //

public UserInfo() {

}

public UserInfo(int userID, String firstname, String lastname, String country, Partner partner, String urole) {

this.userID = userID;

this.firstname = firstname;

this.lastname = lastname;

this.country = country;

this.partner = partner;

this.urole = urole;

}

public UserInfo(UserInfo u) {

this.userID = u.userID;

this.firstname = u.firstname;

this.lastname = u.lastname;

this.country = u.country;

this.partner = u.partner;

this.urole = u.urole;

}

public int getUserID() {

return userID;

}

public void setUserID(int userID) {

this.userID = userID;

}

public String getFirstname() {

return firstname;

}

public void setFirstname(String firstname) {

this.firstname = firstname;

}

public String getLastname() {

return lastname;

}

public void setLastname(String lastname) {

this.lastname = lastname;

}

public String getCountry() {

return country;

}

public void setCountry(String country) {

this.country = country;

}

public Partner getCompany() {

return partner;

}

public void setCompany(Partner partner) {

this.partner = partner;

}

public String getUrole() {

return urole;

}

public void setUrole(String urole) {

this.urole = urole;

}

@Override

public String toString() {

return "UserInfo{" + "userID=" + userID + ", firstname=" + firstname + ", lastname=" + lastname + ", country=" + country + ", companyName=" + partner + ", urole=" + urole + '}';

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer2.domain.interfaces;

/\*\*

\*

\* @author Bancho

\*/

public class NamingConv {

//roles of users

public static final String ROLE = "role";

public static final String PARTNER = "partner";

public static final String ADMIN = "admin";

//log-in

public static final String USERNAME = "username";

public static final String PASSWORD = "password";

//stages of projects

public static final String STAGE = "stage";

public static final String PENDING = "pending";

public static final String APPROVED = "approved";

public static final String FINISHED = "finished";

//commands for Servlet switch case

public static final String NEWCOMMENT = "newcomment";

public static final String INCOMMENT = "newcomment";

public static final String UPDATE\_DESCR = "updatedescription";

public static final String COMMAND = "command";

public static final String RELOAD\_MAIN = "reloadMain";

public static final String LOG\_IN = "log-in";

public static final String LOG\_OUT = "log-out";

public static final String UPLOAD = "upload";

public static final String CREATE\_PROJECT = "createProject";

public static final String CREATE\_COMPANY = "createPartner";

public static final String CREATE\_USER = "createUser";

//beans, created by beanHandlers

public static final String NEW\_PROJECT = "newProject";

public static final String NEW\_PARTNER = "newPartner";

public static final String NEW\_USER\_ATH = "newUserAth";

public static final String NEW\_USER\_INFO = "newUserInfo";

//requset and session attributes

public static final String PROJECT = "project";

public static final String CONTROLLER = "controller";

public static final String PARTNER\_LIST = "partnerList";

public static final String MAINAREA = "mainArea";

public static final String USER = "user";

public static final String PROJECTS = "projects";

public static final String TYPE = "type";

public static final String SDATE = "sdate";

public static final String FDATE = "fdate";

//----------------------------------------------------Nos--------------------------------------------------------

//pages

public static final String DASHBOARD = "dashboard";

public static final String SET\_BUDGET = "setBudget";

public static final String VIEW\_BUDGET = "viewBudget";

public static final String PROJECT\_OVERVIEW = "projectOverview";

public static final String VIEW\_PROJECT\_DETAILS = "viewProjectDetails";

public static final String SEE = "seeProjects";

public static final String PENDING\_PROJECTS = "viewPending";

public static final String APPROVED\_PROJECTS = "viewApproved";

public static final String SUCCESS = "success";

public static final String FAIL = "fail";

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

import java.io.FileNotFoundException;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBConnector {

private DBType dbType = DBType.ORACLE\_THIN;

private Connection conn = null;

// Singleton Start -->

private static DBConnector instance = null;

private DBConnector() {

}

public static DBConnector getInstance() {

if (instance == null) {

instance = new DBConnector();

}

return instance;

}

public Connection getConnection() {

if (conn == null) {

if( openConnection() ) {

System.out.println("Connection opened");

return conn;

} else {

return null;

}

}

return conn;

}

// <-- End of Singleton

public void setDBType(DBType dbType) {

this.dbType = dbType;

}

private boolean openConnection() {

try {

switch (dbType) {

case ORACLE\_THIN:

Class.forName(Oracle.DRIVER);

conn = DriverManager.getConnection(Oracle.URL, Oracle.USERNAME, Oracle.PASSWORD);

return true;

case ORACLE\_THIN\_TEST\_DATABASE:

Class.forName(Oracle.DRIVER);

conn = DriverManager.getConnection(OracleTest.URL, OracleTest.USERNAME, OracleTest.PASSWORD);

return true;

case SOME\_OTHER\_DATA\_BASE:

// Class.forName(SOME\_OTHER\_DATA\_BASE\_DRIVER);

// return DriverManager.getConnection(SOME\_OTHER\_DATA\_BASE, USERNAME, PASSWORD);

return true;

default:

return false;

}

} catch (ClassNotFoundException | SQLException e ) {

System.err.println(e);

return false;

}

}

public void close() {

System.out.println("Closing connection");

try {

conn.close();

conn = null;

} catch (Exception e) {

} // End of try-catch

} // End of close()

public static void processException(SQLException e) {

StringBuilder sb = new StringBuilder();

sb.append("One or more Error(s) or Exception(s) are thrown form the Data Base :: ");

sb.append("\n\n");

sb.append("Error message :: "); sb.append( e.getMessage() ); sb.append("\n");

sb.append("Error code :: "); sb.append( e.getErrorCode() ); sb.append("\n");

sb.append("SQL state :: "); sb.append( e.getSQLState() ); sb.append("\n");

System.err.println(sb);

} // End of processException()

} // End of Class:: ConnectionManager

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

import java.io.FileNotFoundException;

import java.sql.Connection;

import java.sql.SQLException;

import java.util.Collection;

import java.util.logging.Level;

import java.util.logging.Logger;

import layer2.domain.bean.Comment;

import layer2.domain.bean.POE;

import layer2.domain.bean.Partner;

import layer2.domain.bean.Project;

import layer2.domain.bean.UserAuthentication;

import layer2.domain.bean.UserInfo;

import layer3.dataSource.mapper.CommentManager;

import layer3.dataSource.mapper.POEManager;

import layer3.dataSource.mapper.PartnerManager;

import layer3.dataSource.mapper.ProjectManager;

import layer3.dataSource.mapper.UserAutenticationManager;

import layer3.dataSource.mapper.UserInfoManager;

/\*\*

\*

\* @author bo

\*/

public class DBFacade {

private static Connection conn;

private ProjectManager projectManager;

private UserInfoManager userInfoManager;

private UserAutenticationManager userAthMan;

private PartnerManager partnerManager;

private POEManager poeManager;

private CommentManager commentManager;

// Singleton Start -->

private DBFacade() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN);

conn = DBConnector.getInstance().getConnection();

partnerManager = new PartnerManager();

projectManager = new ProjectManager();

userInfoManager = new UserInfoManager();

userAthMan = new UserAutenticationManager();

poeManager = new POEManager();

commentManager= new CommentManager();

}

public static DBFacade getInstance() {

conn = DBConnector.getInstance().getConnection();

return DBFacadeHolder.INSTANCE;

}

private static class DBFacadeHolder {

private static final DBFacade INSTANCE = new DBFacade();

}

// <-- End of Singleton

// private Connection getConnection(){

// try {

// if (conn.isClosed()) {

// conn = DBConnector.getInstance().getConnection();

// }

// } catch (SQLException ex) {

// DBConnector.processException(ex);

// }

// return conn;

// }

public void closeConnection() {

DBConnector.getInstance().close();

}

public boolean createProject(Project newProject) {

// return projectManager.insert(getConnection(), newProject);

return projectManager.insert(conn, newProject);

}

public Project getProject(int projectid) {

// return projectManager.getRow(getConnection(), projectid);

return projectManager.getRow(conn, projectid);

}

public boolean createPartner(Partner partner) {

// return partnerManager.insert(getConnection(), partner);

return partnerManager.insert(conn, partner);

}

public Collection<Project> getAllProjects() {

// return projectManager.getAllRows(getConnection());

return projectManager.getAllRows(conn);

}

public boolean editProject(Project project) {

// return projectManager.update(getConnection(), project);

return projectManager.update(conn, project);

}

public Collection<Partner> getAllPartners() {

// return partnerManager.getAllRows(getConnection());

return partnerManager.getAllRows(conn);

}

public Collection<Comment> getAllCommentForproject(int proNo) {

// return partnerManager.getAllRows(getConnection());

return commentManager.getAllRows(conn, proNo);

}

public boolean createPOE(POE poe) throws FileNotFoundException {

// return poeManager.insert(getConnection(), poe);

return poeManager.insert(conn, poe);

}

public boolean createUserInfo(UserInfo ui) {

// return userInfoManager.insert(getConnection(), ui);

return userInfoManager.insert(conn, ui);

}

public boolean updateProject(Project pr) {

// return userInfoManager.insert(getConnection(), ui);

return projectManager.update(conn, pr);

}

public UserInfo getUserInfo(int userid) {

// return userInfoManager.getRow(getConnection(), userid);

return userInfoManager.getRow(conn, userid);

}

public UserAuthentication getUserAuthentication(String username) {

// return userAthMan.getRow(getConnection(), username);

return userAthMan.getRow(conn, username);

}

public boolean createUserAth(UserAuthentication ua) {

// return userAthMan.insert(getConnection(), ua);

return userAthMan.insert(conn, ua);

}

public boolean createComment(Comment cm) {

// return userAthMan.insert(getConnection(), ua);

return commentManager.insert(conn, cm);

}

} // End of Class :: DBFacade

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

/\*\*

\*

\* @author bo

\*/

public enum DBType {

ORACLE\_THIN, ORACLE\_THIN\_TEST\_DATABASE, SOME\_OTHER\_DATA\_BASE

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

/\*\*

\*

\* @author bo

\*/

public final class Oracle {

public static final String USERNAME = "cphbm75";

public static final String PASSWORD = "cphbm75";

// This is the path to the Oracle database:

public static final String URL = "jdbc:oracle:thin:@datdb.cphbusiness.dk:1521:dat";

// The driver is not needed anymore since jave 7.

// ... But since the programm doesn't work without the driver! I included it.

public static final String DRIVER = "oracle.jdbc.driver.OracleDriver";

// PRIVATE //

/\*\*

\* The caller references the constants using Oracle.USERNAME,

\* and so on. Thus, the caller should be prevented from constructing objects

\* of this class, by declaring this private constructor.

\*/

private Oracle() {

//this prevents even the native class from

//calling this ctor as well :

throw new AssertionError();

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

/\*\*

\*

\* @author bo

\*/

public class OracleTest {

public static final String USERNAME = "cphas293";

public static final String PASSWORD = "cphas293";

// This is the path to the Oracle database:

public static final String URL = "jdbc:oracle:thin:@datdb.cphbusiness.dk:1521:dat";

// The driver is not needed anymore since jave 7.

// ... But since the programm doesn't work without the driver! I included it.

public static final String DRIVER = "oracle.jdbc.driver.OracleDriver";

// PRIVATE //

/\*\*

\* The caller references the constants using OracleTest.USERNAME,

\* and so on. Thus, the caller should be prevented from constructing objects

\* of this class, by declaring this private constructor.

\*/

private OracleTest() {

//this prevents even the native class from

//calling this ctor as well :

throw new AssertionError();

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.\*;

import java.util.ArrayList;

import layer2.domain.bean.Budget;

import layer3.dataSource.DBConnector;

/\*\*

\*

\* @author Bancho

\*/

public class BudgetManager {

public boolean insert(Connection conn, Budget bean){

String sql = "INSERT INTO budget (quarter, qyear, qbudget) VALUES (? , ? , ?)";

int rowsInserted = 0;

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, bean.getQuarter());

stmt.setInt(2, bean.getQyear());

stmt.setInt(3, bean.getQbudget());

rowsInserted = stmt.executeUpdate();

return rowsInserted == 1;

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

}

public boolean update(Connection conn, Budget bean){

String sql = "UPDATE budget SET qbudget = ? WHERE quarter = ? AND qyear = ?";

int rowsAffected = 0;

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, bean.getQbudget());

stmt.setInt(2, bean.getQuarter());

stmt.setInt(3, bean.getQyear());

rowsAffected = stmt.executeUpdate();

return rowsAffected == 1;

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

}

public Budget getRow(Connection conn, int quarter, int qyear){

String sql = "SELECT qbudget FROM budget WHERE quarter = ? AND qyear = ?";

ResultSet rs = null;

Budget budget = new Budget();

budget.setQuarter(quarter);

budget.setQyear(qyear);

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, quarter);

stmt.setInt(2, qyear);

rs = stmt.executeQuery();

if (rs.next()) {

budget.setQbudget(rs.getInt("qbudget"));

}

return budget;

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException ex) {

DBConnector.processException(ex);

}

}

}

}

public boolean delete(Connection conn, int quarter, int qyear){

String sql = "DELETE FROM budget WHERE quarter = ? AND qyear = ?";

int rowsAffected = 0;

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, quarter);

stmt.setInt(2, qyear);

rowsAffected = stmt.executeUpdate();

return rowsAffected == 1;

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

}

public ArrayList<Budget> getAllRows(Connection conn){

String sql = "SELECT \* FROM budget";

ResultSet rs = null;

ArrayList<Budget> rows = new ArrayList<Budget>();

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

rs = stmt.executeQuery();

while(rs.next()){

Budget budget = new Budget();

budget.setQuarter(rs.getInt("quarter"));

budget.setQyear(rs.getInt("qyear"));

budget.setQbudget(rs.getInt("qbudget"));

rows.add(budget);

}

return rows;

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException ex) {

DBConnector.processException(ex);

}

}

}

}

public int deleteAllRows(Connection conn, String confirm){

if (confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM budget";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.Comment;

import layer3.dataSource.DBConnector;

/\*\*

\*

\* @author Athinodoros

\*/

public class CommentManager {

public boolean insert(Connection conn, Comment bean) {

int rowsInserted = 0;

String sql1 = "INSERT into comments (commentID, projectID, userID, comments) VALUES (?,?,?,?)";

// This code is taken from Henrik's DataSourceLayerDemo :: Class OrderMapper.jave

//-------------------------------------------------------------------------------

String sql2

= "select commentSequence.nextval " //= "select orderseq.nextval "

+ "from dual"; // <- this (dual) is a dummy table, and it is needed

// because of the select statemant (oracle data-base only)

//--------------------------------------------------------------------------------

ResultSet keys = null;

try (

PreparedStatement prep1 = conn.prepareStatement(sql1);

PreparedStatement prep2 = conn.prepareStatement(sql2);) {

//== (Henrik's code) get unique key

//--------------------------------

keys = prep2.executeQuery();

if (keys.next()) {

bean.setCommentID(keys.getInt(1)); // <- get the key from dummy table (dual)

} else {

// ToDo -

}

//--------------------------------

prep1.setInt(1, bean.getCommentID());

prep1.setInt(2, bean.getProject().getProjectID());

prep1.setInt(3, bean.getUser().getUserID());

prep1.setString(4, bean.getComment());

rowsInserted = prep1.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

} finally {

if (keys != null) {

try {

keys.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

return rowsInserted == 1;

} //end of method ->Insert

public Comment getRow(Connection conn, int commentID) {

String sql1 = "SELECT \* FROM comments WHERE commentID = ? ";

ResultSet rs = null;

try (PreparedStatement prep = conn.prepareStatement(sql1);) {

prep.setInt(1, commentID);

rs = prep.executeQuery();

if (rs.next()) {

Comment bean = new Comment();

UserInfoManager um = new UserInfoManager();

ProjectManager pm = new ProjectManager();

bean.setCommentID(commentID);

bean.setProject(pm.getRow(conn, rs.getInt("projectID")));

bean.setUser(um.getRow(conn, rs.getInt("userID")));

bean.setComment("comments");

return bean;

} else {

return null;

}

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

}//end of method -> getrow

public Collection<Comment> getAllRows(Connection conn, int projectID) {

Collection<Comment> rows = new ArrayList();

String sql = "SELECT \* FROM comments WHERE projectID = ? ";

ResultSet rs = null;

try (PreparedStatement prep = conn.prepareStatement(sql);) {

prep.setInt(1, projectID);

rs = prep.executeQuery();

while (rs.next()) {

Comment bean = new Comment();

ProjectManager pm = new ProjectManager();

UserInfoManager um = new UserInfoManager();

bean.setCommentID(rs.getInt("commentID"));

bean.setProject(pm.getRow(conn, rs.getInt("projectID")));

bean.setUser(um.getRow(conn, rs.getInt("userID")));

bean.setComment(rs.getString("comments"));

rows.add(bean);

}

return rows;

} catch (SQLException e) {

DBConnector.processException(e);

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

return rows;

}

public int deleteAllRows(Connection conn,String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM comments";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

//public boolean update(Connection conn , Comment bean){}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import com.sun.xml.ws.transport.tcp.io.OutputWriter;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.InputStream;

import java.io.PrintWriter;

import java.sql.\*;

import layer2.domain.bean.POE;

import layer3.dataSource.DBConnector;

import oracle.sql.BLOB;

/\*\*

\*

\* @author bo

\*/

public class POEManager {

public boolean insert(Connection conn, POE bean) {

int rowsInserted = 0;

String sql = "INSERT into POE (projectID, filePath) VALUES (?, ?, ?, ?)";

// This code is taken from Henrik's DataSourceLayerDemo :: Class OrderMapper.jave

//-------------------------------------------------------------------------------

String sql2

= "select POESequence.nextval " //= "select orderseq.nextval "

+ "from dual"; // <- this (dual) is a dummy table, and it is needed

// because of the select statemant (oracle data-base only)

//--------------------------------------------------------------------------------

try ( PreparedStatement stmt = conn.prepareStatement( sql ); ) {

PreparedStatement stmt1 = conn.prepareStatement(sql2);

ResultSet keys;

keys = stmt1.executeQuery();

if (keys.next()) {

bean.setPOEID(keys.getInt(1)); // <- get the key from dummy table (dual)

}

// ProjectManager pm = new ProjectManager();

//== insert

conn.setAutoCommit(false);

stmt.setInt(1, bean.getPOEID() );

stmt.setInt(2, bean.getProject().getProjectID() );

stmt.setString(3, bean.getPrefix());

stmt.setString(4, bean.getFileName());

// stmt.setBinaryStream(5, null );

rowsInserted = stmt.executeUpdate();

conn.commit();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

return rowsInserted == 1;

} // End of method :: insert()

public POE getRow(Connection conn, int projectID) {

String sql = "SELECT \* FROM POE WHERE projectID = ?";

ResultSet rs = null;

try (PreparedStatement stmt = conn.prepareStatement(sql);) {

stmt.setInt(1, projectID);

rs = stmt.executeQuery();

if (rs.next()) {

ProjectManager pm = new ProjectManager();

POE bean = new POE();

InputStream inStr = rs.getBinaryStream("fileBin");

bean.setProject( pm.getRow(conn, rs.getInt("projectID")));

File file = new File("C:\\"+rs.getString("fileName"));

return bean;

} else {

return null;

}

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

} // End of method :: getRow()

public boolean update(Connection conn, POE bean) {

String sql = "UPDATE POE SET fileName = ?, fileBin = ? WHERE projectID = ?, fileName=?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setString (1, bean.getFileName());

stmt.setInt (2, bean.getProject().getProjectID());

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: update()

public boolean delete(Connection conn, int projectID) {

String sql = "DELETE FROM POE WHERE projectID = ?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setInt(1, projectID);

int effected = stmt.executeUpdate();

if (effected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: Delete()

public int deleteAllRows(Connection conn, String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM POE";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

} // End of method :: deleteAllRows

} // End of Class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.\*;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.Partner;

import layer3.dataSource.DBConnector;

/\*\*

\*

\* @author bo

\*/

public class PartnerManager {

public Collection getAllRows(Connection conn) {

Collection<Partner> rows = new ArrayList();

String sql = "SELECT companyName, companyID FROM partner";

try ( Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery( sql );

) {

while (rs.next()) {

Partner partner = new Partner();

partner.setCompanyName(rs.getString("companyName"));

partner.setCompanyID(rs.getInt("companyID"));

rows.add(partner);

}

} catch (SQLException e) {

DBConnector.processException(e);

}

return rows;

} // End of method :: gitAllRows()

public boolean insert(Connection conn, Partner bean) {

int rowsInserted = 0;

String sql1 = "INSERT into partner (companyName, companyID) " + "VALUES (?, ?)";

try ( PreparedStatement stmt = conn.prepareStatement(sql1); ) {

//== insert tuple

stmt.setString (1, bean.getCompanyName());

stmt.setInt (2, bean.getCompanyID());

rowsInserted = stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

return rowsInserted == 1;

} // End of method :: insert()

public Partner getRow(Connection conn, String companyName) {

String sql = "SELECT \* FROM partner WHERE companyName = ?";

ResultSet rs = null;

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setString(1, companyName);

rs = stmt.executeQuery();

if (rs.next()) {

Partner bean = new Partner();

bean.setCompanyName(companyName);

bean.setCompanyID(rs.getInt("companyID"));

return bean;

} else {

return null;

}

} catch ( SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

} // End of method :: getRow()

public boolean update(Connection conn, Partner bean) {

String sql

= "UPDATE partner SET "

+ "companyID = ? "

+ "WHERE companyName = ?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setInt (1, bean.getCompanyID());

stmt.setString (2, bean.getCompanyName());

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: update()

public boolean delete(Connection conn, String companyName) {

String sql = "DELETE FROM partner WHERE companyName = ?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setString(1, companyName);

int effected = stmt.executeUpdate();

if(effected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: Delete()

public int deleteAllRows(Connection conn, String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM partner";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.\*;

import java.text.ParseException;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.Partner;

import layer2.domain.bean.Project;

import layer3.dataSource.DBConnector;

import layer3.dataSource.utility.Convert;

import oracle.sql.DATE;

/\*\*

\*

\* @author bo

\*/

public class ProjectManager {

public boolean insert(Connection conn, Project bean) {

int rowsInserted = 0;

String sql1 = "INSERT into project (projectID, companyName, title, description, stage, sdate, fdate, projectBudget) VALUES (?, ?, ?, ?, ?, ?, ?, ?)";

// This code is taken from Henrik's DataSourceLayerDemo :: Class OrderMapper.jave

//-------------------------------------------------------------------------------

String sql2

= "select projectSequence.nextval " //= "select orderseq.nextval "

+ "from dual"; // <- this (dual) is a dummy table, and it is needed

// because of the select statemant (oracle data-base only)

//--------------------------------------------------------------------------------

ResultSet keys = null;

try (

PreparedStatement stmt = conn.prepareStatement(sql1, Statement.RETURN\_GENERATED\_KEYS);

PreparedStatement stmt1 = conn.prepareStatement(sql2);) {

//== (Henrik's code) get unique key

//--------------------------------

keys = stmt1.executeQuery();

if (keys.next()) {

bean.setProjectID(keys.getInt(1)); // <- get the key from dummy table (dual)

} else {

// ToDo -

}

//--------------------------------

//== insert tuple

stmt.setInt(1, bean.getProjectID());

stmt.setString(2, bean.getPartner().getCompanyName());

stmt.setString(3, bean.getTitle());

stmt.setString(4, bean.getDescription());

stmt.setString(5, bean.getStage());

stmt.setDate(6, Convert.date2SqlDate(bean.getSdate()));

stmt.setDate(7, Convert.date2SqlDate(bean.getFdate()));

stmt.setInt(8, bean.getProjectBudget());

rowsInserted = stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

} finally {

if (keys != null) {

try {

keys.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

return rowsInserted == 1;

} // End of method :: insert()

public Project getRow(Connection conn, int projectid) {

String sql = "SELECT \* FROM project WHERE projectID = ?";

ResultSet rs = null;

Project bean = new Project();

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, projectid);

rs = stmt.executeQuery();

if (rs.next()) {

Partner parter = new Partner();

PartnerManager pm = new PartnerManager();

bean.setProjectID(projectid);

bean.setPartner(pm.getRow(conn, rs.getString("companyName")));

bean.setTitle(rs.getString("title"));

bean.setDescription(rs.getString("description"));

bean.setStage(rs.getString("stage"));

bean.setSdate(rs.getDate("sdate"));

bean.setFdate(rs.getDate("fdate"));

bean.setProjectBudget(rs.getInt("projectBudget"));

}

return bean;

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException ex) {

DBConnector.processException(ex);

}

}

}

} // End of method :: getRow()

public boolean delete(Connection conn, int projectid){

String sql = "DELETE FROM project WHERE projectID = ?";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setInt(1, projectid);

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

}

else{

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: delete()

public boolean update(Connection conn, Project bean){

String sql = "UPDATE project SET companyName = ? , title = ? , description = ? , stage = ? , sdate = ?, fdate = ?, projectBudget = ? WHERE projectID = ?";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

stmt.setString(1, bean.getPartner().getCompanyName());

stmt.setString(2, bean.getTitle());

stmt.setString(3, bean.getDescription());

stmt.setString(4, bean.getStage());

stmt.setDate(5, Convert.date2SqlDate(bean.getSdate()));

stmt.setDate(6, Convert.date2SqlDate(bean.getFdate()));

stmt.setInt(7, bean.getProjectBudget());

stmt.setInt(8, bean.getProjectID());

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

}

else{

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

}

public Collection getAllRows(Connection conn) {

Collection<Project> rows = new ArrayList();

String sql = "SELECT \* FROM project";

try ( Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery( sql );

) {

while (rs.next()) {

// CompanyManager cm = new CompanyManager();

// User bean = new User();

PartnerManager pm = new PartnerManager();

Project bean = new Project();

bean.setProjectID(rs.getInt("projectID"));

bean.setPartner(pm.getRow(conn, rs.getString("companyName")));

bean.setTitle(rs.getString("title"));

bean.setDescription(rs.getString("description"));

bean.setStage(rs.getString("stage"));

bean.setSdate(rs.getDate("sdate"));

bean.setFdate(rs.getDate("fdate"));

bean.setProjectBudget(rs.getInt("projectBudget"));

rows.add(bean);

}

} catch (SQLException e) {

DBConnector.processException(e);

}

return rows;

} // End of method :: getAllRows()

public int deleteAllRows(Connection conn, String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM project";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

} // End of Class :: ProjectManager+

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.\*;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.UserAuthentication;

import layer3.dataSource.DBConnector;

/\*\*

\*

\* @author bo

\*/

public class UserAutenticationManager {

public Collection getAllRows(Connection conn) {

Collection<UserAuthentication> rows = new ArrayList();

String sql = "SELECT userID, uname, password, email FROM userAthentication";

// String sql = "SELECT \* FROM userAthentication";

try ( Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery( sql );

) {

UserInfoManager uim = new UserInfoManager();

while (rs.next()) {

UserAuthentication bean = new UserAuthentication();

bean.setUserInfo(uim.getRow(conn, rs.getInt("userID")));

bean.setUname(rs.getString("uname"));

bean.setPassword(rs.getString("password"));

bean.setEmail(rs.getString("email"));

rows.add(bean);

}

} catch (SQLException e) {

DBConnector.processException(e);

}

return rows;

} // End of method :: gitAllRows()

public boolean insert(Connection conn, UserAuthentication bean) {

int rowsInserted = 0;

String sql1 = "INSERT into userAthentication (userID, uname, password, email) "

+ "VALUES (?, ?, ?, ?)";

ResultSet keys = null;

try ( PreparedStatement stmt = conn.prepareStatement(sql1 );) {

// //== insert tuple

stmt.setInt(1, bean.getUserInfo().getUserID());

stmt.setString(2, bean.getUname());

stmt.setString(3, bean.getPassword());

stmt.setString(4, bean.getEmail());

rowsInserted = stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

} finally {

if (keys != null) {

try {

keys.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

return rowsInserted == 1;

} // End of method :: insert()

public UserAuthentication getRow(Connection conn, String uname) {

String sql = "SELECT \* FROM userAthentication WHERE uname = ?";

ResultSet rs = null;

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setString(1, uname);

rs = stmt.executeQuery();

if (rs.next()) {

UserInfoManager uim = new UserInfoManager();

UserAuthentication bean = new UserAuthentication();

bean.setUname(uname);

bean.setUserInfo(uim.getRow(conn, rs.getInt("userID")));

bean.setPassword(rs.getString("password"));

bean.setEmail(rs.getString("email"));

return bean;

} else {

return null;

}

} catch ( SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

} // End of method :: getRow()

public boolean update(Connection conn, UserAuthentication bean) {

String sql

= "UPDATE userAthentication SET "

+ "userID = ?, "

+ "password = ?, "

+ "email = ? "

+ "WHERE uname = ?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

UserInfoManager uim = new UserInfoManager();

stmt.setInt(1, bean.getUserInfo().getUserID());

stmt.setString(2, bean.getPassword());

stmt.setString(3, bean.getEmail());

stmt.setString(4, bean.getUname());

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: update()

public boolean delete(Connection conn, String uname) {

String sql = "DELETE FROM userAthentication WHERE uname = ?";

try ( PreparedStatement stmt = conn.prepareStatement(sql); ) {

stmt.setString(1, uname);

int effected = stmt.executeUpdate();

if(effected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: Delete()

public int deleteAllRows(Connection conn, String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM userAthentication";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

} // End of class UserAuthentication

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.\*;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.UserInfo;

import layer3.dataSource.DBConnector;

/\*\*

\*

\* @author bo

\*/

public class UserInfoManager {

public Collection getAllRows(Connection conn) {

Collection<UserInfo> rows = new ArrayList();

String sql = "SELECT userID, firstname, lastname, country, companyName, urole FROM userInfo";

try (Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery(sql);) {

PartnerManager pm = new PartnerManager();

while (rs.next()) {

UserInfo userInfo = new UserInfo();

userInfo.setUserID(rs.getInt("userID"));

userInfo.setFirstname(rs.getString("firstname"));

userInfo.setLastname(rs.getString("lastname"));

userInfo.setCountry(rs.getString("country"));

userInfo.setCompany((pm.getRow(conn, rs.getString("companyName"))));

userInfo.setUrole(rs.getString("urole"));

rows.add(userInfo);

}

} catch (SQLException e) {

DBConnector.processException(e);

}

return rows;

} // End of method :: gitAllRows()

public boolean insert(Connection conn, UserInfo bean) {

int rowsInserted = 0;

String sql1 = "INSERT into userInfo (userID, firstname, lastname, country, companyName, urole) "

+ "VALUES (?, ?, ?, ?, ?, ?)";

// This code is taken from Henrik's DataSourceLayerDemo :: Class OrderMapper.jave

//-------------------------------------------------------------------------------

String sql2

= "select UserInfoSequence.nextval " //= "select orderseq.nextval "

+ "from dual"; // <- this (dual) is a dummy table, and it is needed

// because of the select statemant (oracle data-base only)

//--------------------------------------------------------------------------------

ResultSet keys = null;

try (

//PreparedStatement stmt = conn.prepareStatement(sql1, Statement.RETURN\_GENERATED\_KEYS);

PreparedStatement stmt = conn.prepareStatement(sql1 );

// Henrik's code (needed to update primary key (oracle data-base only)->

PreparedStatement stmt1 = conn.prepareStatement( sql2 );

) {

//== (Henrik's code) get unique key

//--------------------------------

keys = stmt1.executeQuery();

if (keys.next()) {

bean.setUserID(keys.getInt(1)); // <- get the key from dummy table (dual)

}

else {

// ToDo -

}

//--------------------------------

//== insert tuplet

stmt.setInt (1, bean.getUserID());

stmt.setString (2, bean.getFirstname());

stmt.setString (3, bean.getLastname());

stmt.setString (4, bean.getCountry());

stmt.setString (5, bean.getCompany().getCompanyName());

stmt.setString (6, bean.getUrole());

rowsInserted = stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return false;

} finally {

if (keys != null) {

try {

keys.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

return rowsInserted == 1;

} // End of method :: insert()

public UserInfo getRow(Connection conn, int userID) {

String sql = "SELECT \* FROM userInfo WHERE userID = ?";

ResultSet rs = null;

try (PreparedStatement stmt = conn.prepareStatement(sql);) {

stmt.setInt(1, userID);

rs = stmt.executeQuery();

if (rs.next()) {

PartnerManager pm = new PartnerManager();

UserInfo bean = new UserInfo();

bean.setUserID(userID);

bean.setFirstname(rs.getString("firstname"));

bean.setLastname(rs.getString("lastname"));

bean.setCountry(rs.getString("country"));

bean.setCompany((pm.getRow(conn, rs.getString("companyName"))));

bean.setUrole(rs.getString("urole"));

return bean;

} else {

return null;

}

} catch (SQLException e) {

DBConnector.processException(e);

return null;

} finally {

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

DBConnector.processException(e);

}

}

}

} // End of method :: getRow()

public boolean update(Connection conn, UserInfo bean) {

String sql

= "UPDATE userInfo SET "

+ "firstname = ?, "

+ "lastname = ?, "

+ "country = ?, "

//+ "companyName = ?, "

+ "urole = ? "

+ "WHERE userID = ?";

try (PreparedStatement stmt = conn.prepareStatement(sql);) {

stmt.setString(1, bean.getFirstname());

stmt.setString(2, bean.getLastname());

stmt.setString(3, bean.getCountry());

//stmt.setString(4, bean.getCompany().getCompanyName());

stmt.setString(4, bean.getUrole());

stmt.setInt(5, bean.getUserID());

int affected = stmt.executeUpdate();

if (affected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: update()

public boolean delete(Connection conn, int userID) {

String sql = "DELETE FROM userInfo WHERE userID = ?";

try (PreparedStatement stmt = conn.prepareStatement(sql);) {

stmt.setInt(1, userID);

int effected = stmt.executeUpdate();

if (effected == 1) {

return true;

} else {

return false;

}

} catch (SQLException e) {

DBConnector.processException(e);

return false;

}

} // End of method :: Delete()

public int deleteAllRows(Connection conn, String confirm){

if( confirm.equalsIgnoreCase("yes")) {

String sql = "DELETE FROM userInfo";

try (PreparedStatement stmt = conn.prepareStatement(sql)) {

return stmt.executeUpdate();

} catch (SQLException e) {

DBConnector.processException(e);

return -1;

}

}else{

return -1;

}

}

} // End of Class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.utility;

import java.util.Calendar;

/\*\*

\*

\* @author bo

\*/

public class Convert {

public static java.sql.Date date2SqlDate(java.util.Date date) {

return new java.sql.Date(date.getTime());

}

public static java.util.Date sqlDate2Date(java.sql.Date date) {

return new java.util.Date();

}

public static String boolean2String(boolean bool) {

if(bool == true)

return "y";

else

return "n";

} // End of method()

public static boolean string2Boolean(String string) {

if( string.equalsIgnoreCase("y") )

return true;

else

return false;

}

} // End of Class :: Convert

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import layer2.domain.bean.POE;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class POEManagerTest {

private static Connection conn;

public POEManagerTest() {}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

@Test

public void testInsert() {

System.out.println("Testing :: POEManager.insert()");

//making sure I insert all the necessary rows in the other tables first

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean result = false;

if (status1 & status2) {

result = poeManager.insert(conn, poe);

}

assertTrue(result);

} // End of method :: testInsert()

@Test

public void testGetRow() {

System.out.println("Testing :: POEManager.getRow()");

//making sure I first insert all the necessary rows, before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

POE result = null;

if (status1 & status2 & status3) {

result = poeManager.getRow(conn, poe.getProject().getProjectID());

}

assertNotNull(result);

}

@Test

public void testUpdate() {

System.out.println("Testing :: POEManager.update()");

//making sure I first insert all the necessary rows, before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

POE poeUpdated = new POE(poe);

//--------------------- ReDo Later---------------

// poeUpdated.setFilePath(FILEPATH + "\_/This is added to the file path");

boolean status4 = poeManager.update(conn, poeUpdated);

POE result = poe;

if (status1 & status2 & status3 & status4) {

result = poeManager.getRow(conn, poeUpdated.getProject().getProjectID());

}

assertFalse(poe.getFilePath().equals(result.getFilePath()));

} // End of method :: testUpdate()

@Test

public void testDelete() {

System.out.println("Testing :: POEManager.delete()");

//making sure I first insert all the necessary rows, before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

boolean result = false;

if (status1 & status2 & status3) {

result = poeManager.delete(conn, poe.getProject().getProjectID());

}

assertTrue(result);

} // End of method :: testDelete()

@Test

public void testDeleteAllRows() {

System.out.println("Testing :: poeManager.DeleteAllRows");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

int rowsDeleted = 0;

if(status1 & status2 & status3) {

rowsDeleted = poeManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource;

import java.sql.Connection;

import java.sql.SQLException;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

/\*\*

\*

\* @author bo

\*/

public class DBConnectorTest {

public DBConnectorTest() {}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {}

@After

public void tearDown() {}

/\*\*

\* Test of getInstance method, of class DBConnector.

\*/

@Test

public void test\_A\_DBConnector\_GetInstance() {

System.out.println("Testing :: DBConnector.getInstance()");

// Get a Instance of DBConnector

DBConnector result1 = DBConnector.getInstance();

assertNotNull(result1);

// Do singleton object DBConnector work?

DBConnector result2 = DBConnector.getInstance();

assertSame(result1, result2);

} // End of test testGetInstance()

/\*\*

\* Test of getConnection method, of class DBConnector.

\*/

@Test

public void test\_B\_DBConnector\_GetConnection() {

System.out.println("Testing :: DBConnector.getConnection()");

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN);

// Get a connection to the data base

Connection result1 = DBConnector.getInstance().getConnection();

assertNotNull(result1);

// Get the samme connection to the data base

Connection result2 = DBConnector.getInstance().getConnection();

assertSame(result1, result2);

} // End of method :: testGetConnection()

/\*\*

\* Test of close method, of class DBConnector.

\*/

@Test

public void test\_C\_DBConnector\_Close() {

System.out.println("Testing :: DBConnector.close()");

// Get a connection to the data base

Connection oldConnection = DBConnector.getInstance().getConnection();

// Closing the connection

DBConnector.getInstance().close();

// Initiates a new connection

Connection newConnection = DBConnector.getInstance().getConnection();

// See if the old and the new connections are the same

assertNotSame(newConnection, oldConnection);

} // End of method :: testClose()

} // End of Class :: DBConnectorTest

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import layer2.domain.bean.Budget;

import layer3.dataSource.DBConnector;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.Test;

import java.sql.\*;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import static org.junit.Assert.\*;

import org.junit.BeforeClass;

/\*\*

\*

\* @author Bancho

\*/

public class BudgetManagerTest {

private static Connection conn;

private BudgetManager budgetManager;

public BudgetManagerTest() {

budgetManager = new BudgetManager();

}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

//Test methods below

@Test

public void testInsert(){

Budget budget = new Budget(1, 2015, 300000);

boolean result = budgetManager.insert(conn, budget);

assertTrue(result);

}

@Test

public void testGetRow(){

Budget budget = new Budget(1, 2015, 300000);

budgetManager.insert(conn, budget);

Budget result = budgetManager.getRow(conn, budget.getQuarter(), budget.getQyear());

assertEquals(budget.toString(), result.toString());

}

@Test

public void testUpdate(){

Budget budget = new Budget(1, 2015, 300000);

budgetManager.insert(conn, budget);

budget.setQbudget(400000);

budgetManager.update(conn, budget);

Budget result = budgetManager.getRow(conn, budget.getQuarter(), budget.getQyear());

assertEquals(budget.toString(), result.toString());

}

@Test

public void testDelete(){

Budget budget = new Budget(1, 2015, 300000);

budgetManager.insert(conn, budget);

boolean result = budgetManager.delete(conn, budget.getQuarter(), budget.getQyear());

assertTrue(result);

}

@Test

public void testDeleteAllRows(){

Budget budget = new Budget(1, 2015, 300000);

budgetManager.insert(conn, budget);

Budget budget2 = new Budget(2, 2015, 400000);

budgetManager.insert(conn, budget2);

Budget budget3 = new Budget(3, 2015, 500000);

budgetManager.insert(conn, budget3);

int rowsDeleted = budgetManager.deleteAllRows(conn, "yes");

assertTrue(rowsDeleted == 3);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import layer2.domain.bean.POE;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class POEManagerTest {

private static Connection conn;

public POEManagerTest() {}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

@Test

public void testInsert() {

System.out.println("Testing :: POEManager.insert()");

//making sure I insert all the necessary rows in the other tables first

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean result = false;

if (status1 & status2) {

result = poeManager.insert(conn, poe);

}

assertTrue(result);

} // End of method :: testInsert()

@Test

public void testGetRow() {

System.out.println("Testing :: POEManager.getRow()");

//making sure I first insert all the necessary rows, before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

POE result = null;

if (status1 & status2 & status3) {

result = poeManager.getRow(conn, poe.getProject().getProjectID());

}

assertNotNull(result);

}

@Test

public void testUpdate() {

System.out.println("Testing :: POEManager.update()");

//making sure I first insert all the necessary rows, before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

POE poeUpdated = new POE(poe);

//--------------------- ReDo Later---------------

// poeUpdated.setFilePath(FILEPATH + "\_/This is added to the file path");

boolean status4 = poeManager.update(conn, poeUpdated);

POE result = poe;

if (status1 & status2 & status3 & status4) {

result = poeManager.getRow(conn, poeUpdated.getProject().getProjectID());

}

assertFalse(poe.getFilePath().equals(result.getFilePath()));

} // End of method :: testUpdate()

@Test

public void testDelete() {

System.out.println("Testing :: POEManager.delete()");

//making sure I first insert all the necessary rows, before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

boolean result = false;

if (status1 & status2 & status3) {

result = poeManager.delete(conn, poe.getProject().getProjectID());

}

assertTrue(result);

} // End of method :: testDelete()

@Test

public void testDeleteAllRows() {

System.out.println("Testing :: poeManager.DeleteAllRows");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = poeManager.insert(conn, poe);

int rowsDeleted = 0;

if(status1 & status2 & status3) {

rowsDeleted = poeManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.util.ArrayList;

import java.util.Collection;

import java.util.Random;

import layer2.domain.bean.Partner;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class PartnerManagerTest {

private static Connection conn;

public PartnerManagerTest() {}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

/\*\*

\* Test of insert method, of class PartnerManager.

\*/

@Test

public void test\_A\_Partner\_Insert() {

System.out.println("Testing :: PartnerManager.insert()");

boolean result = partnerManager.insert(conn, partner);

assertTrue(result);

}

/\*\*

\* Test of delete method, of class PartnerManager.

\*/

@Test

public void test\_B\_PartnerManager\_Delete() {

System.out.println("Testing :: PartnerManager.delete()");

// Making sure I first insert all the necessary rows,

// before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean result = false;

if (status1) {

result = partnerManager.delete(conn, partner.getCompanyName());

}

assertTrue(result);

}

/\*\*

\* Test of getRow method, of class PartnerManager.

\*/

@Test

public void test\_C\_PartnerManager\_GetRow() {

System.out.println("Testing :: PartnerManager.getRow()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

Partner result = null;

if (status1) {

result = partnerManager.getRow(conn, partner.getCompanyName());

}

assertNotNull(result);

} // End of method :: testgetRow

/\*\*

\* Test of update method, of class PartnerManager.

\*/

@Test

public void test\_D\_ParnerManager\_Update() {

System.out.println("Testing :: PartnerManager.Update()");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

// Create a new company with the same name as 'company',

// but with a new budget

Partner partnerUpdated = new Partner(partner);

partnerUpdated.setCompanyID(COMPANY\_ID + 100);

// Update company data in the database

boolean status2 = partnerManager.update(conn, partnerUpdated);

Partner result = partner;

if( status1 & status2 ) {

// retrieve updated data from database

result = partnerManager.getRow(conn, partnerUpdated.getCompanyName());

}

assertFalse(partner.getCompanyID() == result.getCompanyID());

} // End of Method :: testUpdate()

/\*\*

\* Test of getAllRows method, of class PartnerManager.

\*/

@Test

public void test\_E\_PartnerManager\_getAllRows() {

System.out.println("Testing :: PartnerManager.getAllRows()");

// setting up test by creating two companies and inserting them in the database

Partner partner1 = new Partner(COMPANY\_NAME + "\_one", COMPANY\_ID);

Partner partner2 = new Partner(COMPANY\_NAME + "\_Two", COMPANY\_ID);

boolean status1 = partnerManager.insert(conn, partner1);

boolean status2 = partnerManager.insert(conn, partner2);

ArrayList<Partner> rows = new ArrayList<>();

if( status1 & status2 ) {

// retrieve the two inserted companies from the database

rows = new ArrayList<>(partnerManager.getAllRows(conn));

}

int expResult = 2;

int result = rows.size();

assertTrue(expResult == result);

}

@Test

public void test\_F\_UserManager\_DeleteAllRows() {

System.out.println("Testing :: UserManager.DeleteAllRows");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

int rowsDeleted = 0;

if(status1) {

rowsDeleted = partnerManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.util.\*;

import layer2.domain.bean.Project;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class ProjectManagerTest {

private static Connection conn;

public ProjectManagerTest() {}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

/\*\*

\* Test of insert method, of class ProjectManager.

\*/

@Test

public void testInsert() {

System.out.println("Testing :: ProjectManager.insert()");

//making sure I insert all the necessary rows in the other tables first

boolean status1 = partnerManager.insert(conn, partner);

boolean result = false;

if (status1) {

result = projectManager.insert(conn, project);

}

assertTrue(result);

} // End of method :: testInsert()

@Test

public void testGetRow() {

System.out.println("Testing :: ProjectManager.getRow()");

//making sure I first insert all the necessary rows, before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

Project result = null;

if (status1 & status2) {

result = projectManager.getRow(conn, project.getProjectID());

}

assertNotNull(result);

} // End of method :: testGetRow()

@Test

public void testDelete(){

System.out.println("Testing :: ProjectManager.delete()");

//making sure I first insert all the necessary rows, before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean result = false;

if (status1 & status2) {

result = projectManager.delete(conn, project.getProjectID());

}

assertTrue(result);

} // End of method :: testDelete()

@Test

public void testUpdate() {

System.out.println("Testing :: ProjectManager.update()");

//making sure I first insert all the necessary rows, before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

Project projectUpdated = new Project(project);

projectUpdated.setDescription(DESCRIPTION + "\_Updated");

boolean status3 = projectManager.update(conn, projectUpdated);

Project result = project;

if (status1 & status2 & status3) {

result = projectManager.getRow(conn, projectUpdated.getProjectID());

}

assertFalse(project.getDescription().equals(result.getDescription()));

} // End of method :: testUpdate()

/\*\*

\* Test of getAllRows method, of class ProjectManager.

\*/

@Test

public void test\_E\_ProjectManager\_getAllRows() {

System.out.println("Testing :: ProjectManager.getAllRows()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

Project project1 = new Project(project);

Project project2 = new Project(project);

boolean status2 = projectManager.insert(conn, project1);

boolean status3 = projectManager.insert(conn, project2);

ArrayList<Project> rows = new ArrayList();

if (status1 & status2 & status3) {

// Retrieve the two inserted companies from the database

rows = new ArrayList<>(projectManager.getAllRows(conn));

}

int expResult = 2;

int result = rows.size();

assertTrue(expResult == result);

}

@Test

public void test\_F\_UserInfoManager\_DeleteAllRows() {

System.out.println("Testing :: ProjectManager.DeleteAllRows");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

int rowsDeleted = 0;

if(status1 & status2) {

rowsDeleted = projectManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

} // End of class ::

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.UserAuthentication;

import layer2.domain.bean.UserInfo;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class UserAuthenticationManagerTest {

private static Connection conn;

public UserAuthenticationManagerTest() {}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

/\*\*

\* Test of insert method, of class UserAutenticationManager.

\*/

@Test

public void test\_A\_UserAutenticationManager\_Insert() {

System.out.println("Testing :: UserAutenticationManager.insert()");

// Making sure I insert all the necessary rows in the other tables first

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean result = false;

if(status1 & status2) {

result = userAutenticationManager.insert(conn, userAutenticantion);

}

assertTrue(result);

} // End of method :: testInert()

/\*\*

\* Test of delete method, of class UserAutenticationManager.

\*/

@Test

public void test\_B\_UserAutenticationManager\_Delete() {

System.out.println("Testing :: UserAutenticationManager.delete()");

// Making sure I first insert all the necessary rows,

// before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean status3 = userAutenticationManager.insert(conn, userAutenticantion);

boolean result = false;

if(status1 & status2 & status3) {

result = userAutenticationManager.delete(conn, userAutenticantion.getUname());

}

assertTrue(result);

} // End of method :: testDelete()

/\*\*

\* Test of getRow method, of class UserAutenticationManager.

\*/

@Test

public void test\_C\_UserAutenticationManager\_GetRow() {

System.out.println("Testing :: UserAutenticationManager.getRow()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean status3 = userAutenticationManager.insert(conn, userAutenticantion);

UserAuthentication result = null;

if( status1 & status2 & status3 ) {

result = userAutenticationManager.getRow(conn, userAutenticantion.getUname());

}

assertNotNull(result);

} // End of method :: testgetRow

/\*\*

\* Test of update method, of class UserAutenticationManager.

\*/

@Test

public void test\_D\_UserAutenticationManager\_Update() {

System.out.println("Testing :: UserAutenticationManager.Update()");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean status3 = userAutenticationManager.insert(conn, userAutenticantion);

// Change password

UserAuthentication userAuthenciationUpdate = new UserAuthentication(userAutenticantion);

userAuthenciationUpdate.setPassword(PASSWORD + "\_Update");

// Update the user's data in the database

boolean status4 = userAutenticationManager.update(conn, userAuthenciationUpdate);

UserAuthentication result = userAutenticantion;

if( status1 & status2 & status3 & status4) {

result = userAutenticationManager.getRow(conn, userAuthenciationUpdate.getUname());

}

assertFalse(userAutenticantion.getPassword().equals(result.getPassword()));

} // End of Method :: testUpdate()

/\*\*

\* Test of getAllRows method, of class UserAutenticationManager.

\*/

@Test

public void test\_E\_UserAuthenticationManager\_getAllRows() {

System.out.println("Testing :: CompanyManager.getAllRows()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

UserAuthentication user1 = new UserAuthentication(userAutenticantion);

UserAuthentication user2 = new UserAuthentication(userAutenticantion);

// create one user with TWO different usernames (unames)

user1.setUname(UNAME + "\_ONE");

user2.setUname(UNAME + "\_TWO");

boolean status3 = userAutenticationManager.insert(conn, user1);

boolean status4 = userAutenticationManager.insert(conn, user2);

ArrayList<UserAuthentication> rows = new ArrayList();

if( status1 & status2 & status3 & status4) {

// Retrieve the two inserted companies from the database

rows = new ArrayList<>( userAutenticationManager.getAllRows(conn) );

}

int expResult = 2;

int result = rows.size();

assertTrue(expResult == result);

}

@Test

public void test\_F\_UserAuthenticationManager\_testDeleteAllRows(){

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean status3 = userAutenticationManager.insert(conn, userAutenticantion);

int rowsDeleted = 0;

if(status1 & status2 & status3) {

rowsDeleted = userAutenticationManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

} // End of Class

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.util.ArrayList;

import java.util.Collection;

import layer2.domain.bean.UserInfo;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author bo

\*/

public class UserInfoManagerTest {

private static Connection conn;

public UserInfoManagerTest() {

}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

/\*\*

\* Test of insert method, of class UserInfoManager.

\*/

@Test

public void test\_A\_UserInfoManager\_Insert() {

System.out.println("Testing :: UserInfoManager.insert()");

// Making sure I first insert all the necessary rows,

// before I try to Insert something

boolean status = partnerManager.insert(conn, partner);

boolean result = false;

if(status) {

result = userInfoManager.insert(conn, userInfo);

}

assertTrue(result);

}

/\*\*

\* Test of delete method, of class UserInfoManager.

\*/

@Test

public void test\_B\_UserInfoManager\_Delete() {

System.out.println("Testing :: UserInfoManager.delete()");

// Making sure I first insert all the necessary rows,

// before I try to delete something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

boolean result = false;

if (status1 & status2) {

result = userInfoManager.delete(conn, userInfo.getUserID());

}

assertTrue(result);

}

/\*\*

\* Test of getRow method, of class UserInfoManager.

\*/

@Test

public void test\_C\_UserInfoManager\_GetRow() {

System.out.println("Testing :: UserInfoManager.getRow()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

UserInfo result = null;

if (status1 & status2) {

result = userInfoManager.getRow(conn, userInfo.getUserID());

}

assertNotNull(result);

} // End of method :: testgetRow

/\*\*

\* Test of update method, of class UserInfoManager.

\*/

@Test

public void test\_D\_UserInfoManager\_Update() {

System.out.println("Testing :: UserInfoManager.Update()");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

// Create a new user with the same userid as 'user',

// but with a new user name

UserInfo userInfoUpdated = new UserInfo(userInfo);

userInfoUpdated.setFirstname(FIRST\_NAME + "\_Update");

// Update the new user's data in the database

boolean status3 = userInfoManager.update(conn, userInfoUpdated);

UserInfo result = userInfo;

if (status1 & status2 & status3) {

result = userInfoManager.getRow(conn, userInfoUpdated.getUserID());

}

assertFalse(userInfo.getFirstname().equals(result.getFirstname()));

} // End of Method :: testUpdate()

/\*\*

\* Test of getAllRows method, of class UserInfoManager.

\*/

@Test

public void test\_E\_UserInfoManager\_getAllRows() {

System.out.println("Testing :: UserInfoManager.getAllRows()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status3 = partnerManager.insert(conn, partner);

UserInfo user1 = new UserInfo(userInfo);

UserInfo user2 = new UserInfo(userInfo);

boolean status1 = userInfoManager.insert(conn, user1);

boolean status2 = userInfoManager.insert(conn, user2);

ArrayList<UserInfo> rows = new ArrayList();

if (status1 & status2 & status3) {

// Retrieve the two inserted companies from the database

rows = new ArrayList<>(userInfoManager.getAllRows(conn));

}

int expResult = 2;

int result = rows.size();

assertTrue(expResult == result);

}

@Test

public void test\_F\_UserInfoManager\_DeleteAllRows() {

System.out.println("Testing :: UserInfoManager.DeleteAllRows");

// Making sure I first insert all the necessary rows,

// before I try to update something

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = userInfoManager.insert(conn, userInfo);

int rowsDeleted = 0;

if(status1 & status2) {

rowsDeleted = userInfoManager.deleteAllRows(conn, "yes");

}

assertTrue(rowsDeleted == 1);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper;

import java.sql.Connection;

import java.util.ArrayList;

import layer2.domain.bean.Comment;

import layer2.domain.bean.Project;

import layer3.dataSource.DBConnector;

import layer3.dataSource.DBType;

import layer3.dataSource.mapper.utility.Delete;

import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.Test;

import static org.junit.Assert.\*;

// Import TestData

import static layer3.dataSource.mapper.utility.TestData.\*;

/\*\*

\*

\* @author Athinodoros

\*/

public class commentManagerTest {

private static Connection conn;

public commentManagerTest() {

}

@BeforeClass

public static void setUpClass() {

DBConnector.getInstance().setDBType(DBType.ORACLE\_THIN\_TEST\_DATABASE);

conn = DBConnector.getInstance().getConnection();

}

@AfterClass

public static void tearDownClass() {

DBConnector.getInstance().close();

}

@Before

public void setUp() {

Delete.database(conn, "yes");

}

@After

public void tearDown() {

}

@Test

public void testInsert() {

System.out.println("Testing :: CommentManager.insert()");

//making sure I insert all the necessary rows in the other tables first

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean result = false;

if (status1 & status2) {

result = commentMananger.insert(conn, comment);

}

assertTrue(result);

}

@Test

public void testGetRow() {

System.out.println("Testing :: CommentManager.getRow()");

//making sure I first insert all the necessary rows, before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

boolean status3 = commentMananger.insert(conn, comment);

Comment result = null;

if (status1 & status2 & status3) {

result = commentMananger.getRow(conn, comment.getCommentID());

}

assertNotNull(result);

}

@Test

public void testGetAllProjectComments() {

System.out.println("Testing :: CommentManager.getAllrojectComments()");

// Making sure I first insert all the necessary rows,

// before I try to get something out

boolean status1 = partnerManager.insert(conn, partner);

boolean status2 = projectManager.insert(conn, project);

Comment comment1 = new Comment(comment);

Comment comment2 = new Comment(comment);

comment2.setComment(COMMENT + "\_Plus something added");

// Ceate two comments for the some project

boolean status3 = commentMananger.insert(conn, comment1);

boolean status4 = commentMananger.insert(conn, comment2);

ArrayList<Comment> rows = new ArrayList();

if (status1 & status2 & status3) {

// Retrieve the two inserted companies from the database

rows = new ArrayList<>( commentMananger.getAllRows(conn, comment1.getProject().getProjectID()) );

}

int expResult = 2;

int result = rows.size();

assertTrue(expResult == result);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper.utility;

import java.sql.Connection;

import layer2.domain.bean.UserAuthentication;

import layer3.dataSource.mapper.\*;

//import layer3.dataSource.mapper.CompanyManager;

//import layer3.dataSource.mapper.ProjectManager;

//import layer3.dataSource.mapper.UserManager;

/\*\*

\*

\* @author bo

\*/

public class Delete {

private static final PartnerManager partnerManager = new PartnerManager();

private static final UserInfoManager userInforManager = new UserInfoManager();

private static final UserAutenticationManager userAutenticationManager = new UserAutenticationManager();

private static final BudgetManager budgetManager = new BudgetManager();

private static final ProjectManager projectManager = new ProjectManager();

private static final CommentManager commentManager = new CommentManager();

private static final POEManager poeManager = new POEManager();

private Delete() {}

public static int database(Connection conn, String confirm ) {

// Delete all data in the database (should be inserted in drop order)

int affected = 0;

affected =+ poeManager.deleteAllRows(conn, confirm);

affected =+ commentManager.deleteAllRows(conn, confirm);

affected =+ projectManager.deleteAllRows(conn, confirm);

affected =+ budgetManager.deleteAllRows(conn, confirm);

affected =+ userAutenticationManager.deleteAllRows(conn, confirm);

affected =+ userInforManager.deleteAllRows(conn, confirm);

affected =+ partnerManager.deleteAllRows(conn, confirm);

return affected;

}

} // End of class :: Delete

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package layer3.dataSource.mapper.utility;

import java.util.Calendar;

import java.util.Date;

import java.util.Random;

import layer2.domain.bean.\*;

import layer3.dataSource.mapper.\*;

/\*\*

\*

\* @author bo

\*/

public final class TestData {

public static final POEManager poeManager = new POEManager();

public static final ProjectManager projectManager = new ProjectManager();

public static final PartnerManager partnerManager = new PartnerManager();

public static final UserInfoManager userInfoManager = new UserInfoManager();

public static final UserAutenticationManager userAutenticationManager = new UserAutenticationManager();

public static final CommentManager commentMananger = new CommentManager();

// Company

// ------

//

private static final Random random = new Random();

public static final String COMPANY\_NAME = "CompanyNameTest\_" + random.nextInt(100\_000\_000);

public static final int COMPANY\_ID = 0; // this value is probably not needed

// UserInfo

// ------

// USER\_ID will be set in the call to insetRow()

public static final int USER\_ID = 0;

public static final String FIRST\_NAME = "firstNameTest";

public static final String LAST\_NAME = "lastNameTest";

public static final String COUNTRY = "Test";

public static final Partner PARTNER = new Partner(COMPANY\_NAME, COMPANY\_ID);

public static final String U\_ROLE = "roleTest";

// UserAutentication

// ------

// - public static final int USER\_ID = 0;

public static final String UNAME = "unameTest" + random.nextInt(100\_000\_000);

public static final String PASSWORD = "passwordTest";

public static final String EMAIL = "email@Test.com";

// Project

// ----------

//

private static final Calendar cal = Calendar.getInstance();

private static final Date today = cal.getTime();

// - public static final USER\_ID = 0;

public static final String TITLE\_PROJECT = "Title\_Project";

// - public static final Partner partner;

public static final String DESCRIPTION = "Description";

public static final String STAGE = "Stage";

public static final Date SDATE = today;

public static final Date FDATE = today;

public static final int PROJECT\_BUDGET = 1000;

// Comment

// ------------

//

public static final int COMMENT\_ID = 0;

// - private static final Project project;

// - private static final UserInfo userInfo;

public static final String COMMENT = "This is a very small comment";

// POE

// ------------

//

// - public static final int POE\_ID = 0;

public static final String FILEPATH = "filePath";

// Make some beans

// ------------

//

public static final Partner partner = new Partner(COMPANY\_NAME, COMPANY\_ID);

public static final UserInfo userInfo = new UserInfo(USER\_ID, FIRST\_NAME, LAST\_NAME, COUNTRY, PARTNER, U\_ROLE);

public static final UserAuthentication userAutenticantion = new UserAuthentication(userInfo, UNAME, PASSWORD, EMAIL);

public static final Project project = new Project(USER\_ID, partner, UNAME, FIRST\_NAME, UNAME, today, today, COMPANY\_ID);

public static final Comment comment = new Comment(COMMENT\_ID, project, userInfo, COMMENT);

public static final int poeid = random.nextInt(100\_000\_000);

public static final POE poe = new POE(USER\_ID, project, COUNTRY, U\_ROLE, UNAME);

private TestData() {

//this prevents even the native class from

//calling this ctor as well :

throw new AssertionError();

}

}

alter session set nls\_date\_format = 'yyyy-mm-dd';

drop table budget;

drop table POE;

drop table comments;

drop table Project;

drop table userAthentication;

drop table userInfo;

drop table partner;

drop table dell;

DROP SEQUENCE POESequence;

DROP SEQUENCE commentSequence;

DROP SEQUENCE projectSequence;

DROP SEQUENCE UserInfoSequence;

create table budget(

quarter integer,

qyear integer,

qbudget integer,

primary key (quarter, qyear));

create table partner (

companyName varchar(30) primary key,

companyID integer);

create table userInfo (

userID integer primary key ,

firstname varchar(20),

lastname varchar(20),

country varchar(4),

companyName varchar(30) references partner(companyName) ,

urole varchar(20));

create table userAthentication (

uname varchar(30) primary key,

userID integer references userInfo(userID),

password varchar(32),

email varchar(50));

create table project (

projectID integer primary key,

-- companyName varchar(30) references Partner(companyName) references dell(companyName),

companyName varchar(30) references Partner(companyName),

title varchar(40),

description varchar(1200),

stage varchar(40),

sdate date,

fdate date,

projectBudget integer);

create table POE (

POEID integer primary key,

projectID integer references Project(projectID),

prefixs varchar(18),

fileName varchar(18),

fileBin BLOB );

create table comments (

commentID integer primary key,

projectID integer references project(projectID),

userID integer references userInfo(userID),

comments varchar(2000));

CREATE SEQUENCE projectSequence

MINVALUE 1

MAXVALUE 999999999999999999999999999

START WITH 1000

INCREMENT BY 1

CACHE 20;

CREATE SEQUENCE UserInfoSequence

MINVALUE 1

MAXVALUE 999999999999999999999999999

START WITH 1000

INCREMENT BY 1

CACHE 20;

CREATE SEQUENCE commentSequence

MINVALUE 1

MAXVALUE 999999999999999999999999999

START WITH 1000

INCREMENT BY 1

CACHE 20;

CREATE SEQUENCE POESequence

MINVALUE 1

MAXVALUE 999999999999999999999999999

START WITH 1000

INCREMENT BY 1

CACHE 20;

commit;

INSERT INTO PARTNER VALUES ('Athinodoros CORP', '1') ;

INSERT INTO PARTNER VALUES ('Bani Productions', '2') ;

INSERT INTO PARTNER VALUES ('Bo Solutions', '3') ;

insert into USERINFO VALUES (1, 'Klaus','Klauson','DK', null ,'admin' );

insert into USERINFO VALUES (2, 'Bancho','Petrov','NO','Bani Productions','partner' );

insert into USERINFO VALUES (3, 'Athinodoros','Sgouromallis','IS','Athinodoros CORP','partner' );

insert into USERINFO VALUES (4, 'Bo','Vilstrup','DK','Bo Solutions','partner' );

INSERT INTO USERATHENTICATION VALUES ( 'Klaus',1,'Klaus','klaus@Dell.com');

INSERT INTO USERATHENTICATION VALUES ( 'Bancho',2,'Bancho','Bancho@Bacholand.com');

INSERT INTO USERATHENTICATION VALUES ( 'Athinodoros',3,'Athinodoros','athinodoros.sgouromallis@hotmail.com');

INSERT INTO USERATHENTICATION VALUES ( 'Bo',4,'Bo','Bo@hotmail.com');

INSERT INTO PROJECT VALUES ( 1,'Athinodoros CORP','Title','This is a description','pending',null,null,13333);

INSERT INTO PROJECT VALUES ( 6,'Athinodoros CORP','Title','This is a description','finished',null,null,13333);

///

/\*!

\* Bootstrap v3.3.2 (http://getbootstrap.com)

\* Copyright 2011-2015 Twitter, Inc.

\* Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)

\*//\*! normalize.css v3.0.2 | MIT License | git.io/normalize \*/

html{font-family:sans-serif;

-webkit-text-size-adjust:100%;

-ms-text-size-adjust:100%}

body{margin:0}

article,aside,details,figcaption,figure,footer,header,hgroup,main,menu,nav,section,summary{

display:block}

audio,canvas,progress,video{

display:inline-block;

vertical-align:baseline}

audio:not([controls]){

display:none;height:0}

[hidden],template{display:none}

a{background-color:transparent}

a:active,a:hover{outline:0}abbr[title]{border-bottom:1px dotted}

b,strong{font-weight:700}

dfn{font-style:italic}

h1{margin:.67em 0;font-size:2em}

mark{color:#000;background:#ff0}

small{font-size:80%}

sub,sup{position:relative;

font-size:75%;

line-height:0;

vertical-align:baseline}

sup{top:-.5em}

sub{bottom:-.25em}

img{border:0}

svg:not(:root){overflow:hidden}

figure{margin:1em 40px}

hr{height:0;

-webkit-box-sizing:content-box;

-moz-box-sizing:content-box;

box-sizing:content-box}

pre{overflow:auto}

code,kbd,pre,samp{

font-family:monospace,monospace;

font-size:1em}

button,input,optgroup,select,textarea{

margin:0;

font:inherit;

color:inherit}

button{overflow:visible}

button,select{text-transform:none}

button,html input[type=button],

input[type=reset],

input[type=submit]{

-webkit-appearance:button;cursor:pointer}

button[disabled],html input[disabled]{cursor:default}

button::-moz-focus-inner,input::-moz-focus-inner{padding:0;border:0}input{line-height:normal}

input[type=checkbox],

input[type=radio]{-webkit-box-sizing:border-box;

-moz-box-sizing:border-box;box-sizing:border-box;padding:0}input[type=number]::-webkit-inner-spin-button,

input[type=number]::-webkit-outer-spin-button{height:auto}input[type=search]{-webkit-box-sizing:content-box;-moz-box-sizing:content-box;box-sizing:content-box;-webkit-appearance:textfield}input[type=search]::-webkit-search-cancel-button,input[type=search]::-webkit-search-decoration{-webkit-appearance:none}fieldset{padding:.35em .625em .75em;margin:0 2px;border:1px solid silver}legend{padding:0;border:0}textarea{overflow:auto}optgroup{font-weight:700}table{border-spacing:0;border-collapse:collapse}td,th{padding:0}/\*! Source: https://github.com/h5bp/html5-boilerplate/blob/master/src/css/main.css \*/@media print{\*,:after,:before{color:#000!important;text-shadow:none!important;background:0 0!important;-webkit-box-shadow:none!important;box-shadow:none!important}a,a:visited{text-decoration:underline}a[href]:after{content:" (" attr(href) ")"}abbr[title]:after{content:" (" attr(title) ")"}a[href^="javascript:"]:after,a[href^="#"]:after{content:""}blockquote,pre{border:1px solid #999;page-break-inside:avoid}thead{display:table-header-group}img,tr{page-break-inside:avoid}img{max-width:100%!important}h2,h3,p{orphans:3;widows:3}h2,h3{page-break-after:avoid}select{background:#fff!important}.navbar{display:none}.btn>.caret,.dropup>.btn>.caret{border-top-color:#000!important}.label{border:1px solid #000}.table{border-collapse:collapse!important}.table td,.table th{background-color:#fff!important}.table-bordered td,.table-bordered th{border:1px solid #ddd!important}}@font-face{font-family:'Glyphicons Halflings';src:url(../fonts/glyphicons-halflings-regular.eot);src:url(../fonts/glyphicons-halflings-regular.eot?#iefix) format('embedded-opentype'),url(../fonts/glyphicons-halflings-regular.woff2) format('woff2'),url(../fonts/glyphicons-halflings-regular.woff) format('woff'),url(../fonts/glyphicons-halflings-regular.ttf) format('truetype'),url(../fonts/glyphicons-halflings-regular.svg#glyphicons\_halflingsregular) format('svg')}.glyphicon{position:relative;top:1px;display:inline-block;font-family:'Glyphicons Halflings';font-style:normal;font-weight:400;line-height:1;-webkit-font-smoothing:antialiased;-moz-osx-font-smoothing:grayscale}.glyphicon-asterisk:before{content:"\2a"}.glyphicon-plus:before{content:"\2b"}.glyphicon-eur:before,.glyphicon-euro:before{content:"\20ac"}.glyphicon-minus:before{content:"\2212"}.glyphicon-cloud:before{content:"\2601"}.glyphicon-envelope:before{content:"\2709"}.glyphicon-pencil:before{content:"\270f"}.glyphicon-glass:before{content:"\e001"}.glyphicon-music:before{content:"\e002"}.glyphicon-search:before{content:"\e003"}.glyphicon-heart:before{content:"\e005"}.glyphicon-star:before{content:"\e006"}.glyphicon-star-empty:before{content:"\e007"}.glyphicon-user:before{content:"\e008"}.glyphicon-film:before{content:"\e009"}.glyphicon-th-large:before{content:"\e010"}.glyphicon-th:before{content:"\e011"}.glyphicon-th-list:before{content:"\e012"}.glyphicon-ok:before{content:"\e013"}.glyphicon-remove:before{content:"\e014"}.glyphicon-zoom-in:before{content:"\e015"}.glyphicon-zoom-out:before{content:"\e016"}.glyphicon-off:before{content:"\e017"}.glyphicon-signal:before{content:"\e018"}.glyphicon-cog:before{content:"\e019"}.glyphicon-trash:before{content:"\e020"}.glyphicon-home:before{content:"\e021"}.glyphicon-file:before{content:"\e022"}.glyphicon-time:before{content:"\e023"}.glyphicon-road:before{content:"\e024"}.glyphicon-download-alt:before{content:"\e025"}.glyphicon-download:before{content:"\e026"}.glyphicon-upload:before{content:"\e027"}.glyphicon-inbox:before{content:"\e028"}.glyphicon-play-circle:before{content:"\e029"}.glyphicon-repeat:before{content:"\e030"}.glyphicon-refresh:before{content:"\e031"}.glyphicon-list-alt:before{content:"\e032"}.glyphicon-lock:before{content:"\e033"}.glyphicon-flag:before{content:"\e034"}.glyphicon-headphones:before{content:"\e035"}.glyphicon-volume-off:before{content:"\e036"}.glyphicon-volume-down:before{content:"\e037"}.glyphicon-volume-up:before{content:"\e038"}.glyphicon-qrcode:before{content:"\e039"}.glyphicon-barcode:before{content:"\e040"}.glyphicon-tag:before{content:"\e041"}.glyphicon-tags:before{content:"\e042"}.glyphicon-book:before{content:"\e043"}.glyphicon-bookmark:before{content:"\e044"}.glyphicon-print:before{content:"\e045"}.glyphicon-camera:before{content:"\e046"}.glyphicon-font:before{content:"\e047"}.glyphicon-bold:before{content:"\e048"}.glyphicon-italic:before{content:"\e049"}.glyphicon-text-height:before{content:"\e050"}.glyphicon-text-width:before{content:"\e051"}.glyphicon-align-left:before{content:"\e052"}.glyphicon-align-center:before{content:"\e053"}.glyphicon-align-right:before{content:"\e054"}.glyphicon-align-justify:before{content:"\e055"}.glyphicon-list:before{content:"\e056"}.glyphicon-indent-left:before{content:"\e057"}.glyphicon-indent-right:before{content:"\e058"}.glyphicon-facetime-video:before{content:"\e059"}.glyphicon-picture:before{content:"\e060"}.glyphicon-map-marker:before{content:"\e062"}.glyphicon-adjust:before{content:"\e063"}.glyphicon-tint:before{content:"\e064"}.glyphicon-edit:before{content:"\e065"}.glyphicon-share:before{content:"\e066"}.glyphicon-check:before{content:"\e067"}.glyphicon-move:before{content:"\e068"}.glyphicon-step-backward:before{content:"\e069"}.glyphicon-fast-backward:before{content:"\e070"}.glyphicon-backward:before{content:"\e071"}.glyphicon-play:before{content:"\e072"}.glyphicon-pause:before{content:"\e073"}.glyphicon-stop:before{content:"\e074"}.glyphicon-forward:before{content:"\e075"}.glyphicon-fast-forward:before{content:"\e076"}.glyphicon-step-forward:before{content:"\e077"}.glyphicon-eject:before{content:"\e078"}.glyphicon-chevron-left:before{content:"\e079"}.glyphicon-chevron-right:before{content:"\e080"}.glyphicon-plus-sign:before{content:"\e081"}.glyphicon-minus-sign:before{content:"\e082"}.glyphicon-remove-sign:before{content:"\e083"}.glyphicon-ok-sign:before{content:"\e084"}.glyphicon-question-sign:before{content:"\e085"}.glyphicon-info-sign:before{content:"\e086"}.glyphicon-screenshot:before{content:"\e087"}.glyphicon-remove-circle:before{content:"\e088"}.glyphicon-ok-circle:before{content:"\e089"}.glyphicon-ban-circle:before{content:"\e090"}.glyphicon-arrow-left:before{content:"\e091"}.glyphicon-arrow-right:before{content:"\e092"}.glyphicon-arrow-up:before{content:"\e093"}.glyphicon-arrow-down:before{content:"\e094"}.glyphicon-share-alt:before{content:"\e095"}.glyphicon-resize-full:before{content:"\e096"}.glyphicon-resize-small:before{content:"\e097"}.glyphicon-exclamation-sign:before{content:"\e101"}.glyphicon-gift:before{content:"\e102"}.glyphicon-leaf:before{content:"\e103"}.glyphicon-fire:before{content:"\e104"}.glyphicon-eye-open:before{content:"\e105"}.glyphicon-eye-close:before{content:"\e106"}.glyphicon-warning-sign:before{content:"\e107"}.glyphicon-plane:before{content:"\e108"}.glyphicon-calendar:before{content:"\e109"}.glyphicon-random:before{content:"\e110"}.glyphicon-comment:before{content:"\e111"}.glyphicon-magnet:before{content:"\e112"}.glyphicon-chevron-up:before{content:"\e113"}.glyphicon-chevron-down:before{content:"\e114"}.glyphicon-retweet:before{content:"\e115"}.glyphicon-shopping-cart:before{content:"\e116"}.glyphicon-folder-close:before{content:"\e117"}.glyphicon-folder-open:before{content:"\e118"}.glyphicon-resize-vertical:before{content:"\e119"}.glyphicon-resize-horizontal:before{content:"\e120"}.glyphicon-hdd:before{content:"\e121"}.glyphicon-bullhorn:before{content:"\e122"}.glyphicon-bell:before{content:"\e123"}.glyphicon-certificate:before{content:"\e124"}.glyphicon-thumbs-up:before{content:"\e125"}.glyphicon-thumbs-down:before{content:"\e126"}.glyphicon-hand-right:before{content:"\e127"}.glyphicon-hand-left:before{content:"\e128"}.glyphicon-hand-up:before{content:"\e129"}.glyphicon-hand-down:before{content:"\e130"}.glyphicon-circle-arrow-right:before{content:"\e131"}.glyphicon-circle-arrow-left:before{content:"\e132"}.glyphicon-circle-arrow-up:before{content:"\e133"}.glyphicon-circle-arrow-down:before{content:"\e134"}.glyphicon-globe:before{content:"\e135"}.glyphicon-wrench:before{content:"\e136"}.glyphicon-tasks:before{content:"\e137"}.glyphicon-filter:before{content:"\e138"}.glyphicon-briefcase:before{content:"\e139"}.glyphicon-fullscreen:before{content:"\e140"}.glyphicon-dashboard:before{content:"\e141"}.glyphicon-paperclip:before{content:"\e142"}.glyphicon-heart-empty:before{content:"\e143"}.glyphicon-link:before{content:"\e144"}.glyphicon-phone:before{content:"\e145"}.glyphicon-pushpin:before{content:"\e146"}.glyphicon-usd:before{content:"\e148"}.glyphicon-gbp:before{content:"\e149"}.glyphicon-sort:before{content:"\e150"}.glyphicon-sort-by-alphabet:before{content:"\e151"}.glyphicon-sort-by-alphabet-alt:before{content:"\e152"}.glyphicon-sort-by-order:before{content:"\e153"}.glyphicon-sort-by-order-alt:before{content:"\e154"}.glyphicon-sort-by-attributes:before{content:"\e155"}.glyphicon-sort-by-attributes-alt:before{content:"\e156"}.glyphicon-unchecked:before{content:"\e157"}.glyphicon-expand:before{content:"\e158"}.glyphicon-collapse-down:before{content:"\e159"}.glyphicon-collapse-up:before{content:"\e160"}.glyphicon-log-in:before{content:"\e161"}.glyphicon-flash:before{content:"\e162"}.glyphicon-log-out:before{content:"\e163"}.glyphicon-new-window:before{content:"\e164"}.glyphicon-record:before{content:"\e165"}.glyphicon-save:before{content:"\e166"}.glyphicon-open:before{content:"\e167"}.glyphicon-saved:before{content:"\e168"}.glyphicon-import:before{content:"\e169"}.glyphicon-export:before{content:"\e170"}

.glyphicon-send:before{content:"\e171"}.glyphicon-floppy-disk:before{content:"\e172"}.glyphicon-floppy-saved:before{content:"\e173"}.glyphicon-floppy-remove:before{content:"\e174"}.glyphicon-floppy-save:before{content:"\e175"}.glyphicon-floppy-open:before{content:"\e176"}.glyphicon-credit-card:before{content:"\e177"}.glyphicon-transfer:before{content:"\e178"}.glyphicon-cutlery:before{content:"\e179"}.glyphicon-header:before{content:"\e180"}.glyphicon-compressed:before{content:"\e181"}.glyphicon-earphone:before{content:"\e182"}.glyphicon-phone-alt:before{content:"\e183"}.glyphicon-tower:before{content:"\e184"}.glyphicon-stats:before{content:"\e185"}.glyphicon-sd-video:before{content:"\e186"}.glyphicon-hd-video:before{content:"\e187"}.glyphicon-subtitles:before{content:"\e188"}.glyphicon-sound-stereo:before{content:"\e189"}.glyphicon-sound-dolby:before{content:"\e190"}.glyphicon-sound-5-1:before{content:"\e191"}.glyphicon-sound-6-1:before{content:"\e192"}.glyphicon-sound-7-1:before{content:"\e193"}.glyphicon-copyright-mark:before{content:"\e194"}.glyphicon-registration-mark:before{content:"\e195"}.glyphicon-cloud-download:before{content:"\e197"}.glyphicon-cloud-upload:before{content:"\e198"}.glyphicon-tree-conifer:before{content:"\e199"}.glyphicon-tree-deciduous:before{content:"\e200"}.glyphicon-cd:before{content:"\e201"}.glyphicon-save-file:before{content:"\e202"}.glyphicon-open-file:before{content:"\e203"}.glyphicon-level-up:before{content:"\e204"}.glyphicon-copy:before{content:"\e205"}.glyphicon-paste:before{content:"\e206"}.glyphicon-alert:before{content:"\e209"}.glyphicon-equalizer:before{content:"\e210"}.glyphicon-king:before{content:"\e211"}.glyphicon-queen:before{content:"\e212"}.glyphicon-pawn:before{content:"\e213"}.glyphicon-bishop:before{content:"\e214"}.glyphicon-knight:before{content:"\e215"}.glyphicon-baby-formula:before{content:"\e216"}.glyphicon-tent:before{content:"\26fa"}.glyphicon-blackboard:before{content:"\e218"}.glyphicon-bed:before{content:"\e219"}.glyphicon-apple:before{content:"\f8ff"}.glyphicon-erase:before{content:"\e221"}.glyphicon-hourglass:before{content:"\231b"}.glyphicon-lamp:before{content:"\e223"}.glyphicon-duplicate:before{content:"\e224"}.glyphicon-piggy-bank:before{content:"\e225"}.glyphicon-scissors:before{content:"\e226"}.glyphicon-bitcoin:before{content:"\e227"}.glyphicon-yen:before{content:"\00a5"}.glyphicon-ruble:before{content:"\20bd"}.glyphicon-scale:before{content:"\e230"}.glyphicon-ice-lolly:before{content:"\e231"}.glyphicon-ice-lolly-tasted:before{content:"\e232"}.glyphicon-education:before{content:"\e233"}.glyphicon-option-horizontal:before{content:"\e234"}.glyphicon-option-vertical:before{content:"\e235"}.glyphicon-menu-hamburger:before{content:"\e236"}.glyphicon-modal-window:before{content:"\e237"}.glyphicon-oil:before{content:"\e238"}.glyphicon-grain:before{content:"\e239"}.glyphicon-sunglasses:before{content:"\e240"}.glyphicon-text-size:before{content:"\e241"}.glyphicon-text-color:before{content:"\e242"}.glyphicon-text-background:before{content:"\e243"}.glyphicon-object-align-top:before{content:"\e244"}.glyphicon-object-align-bottom:before{content:"\e245"}.glyphicon-object-align-horizontal:before{content:"\e246"}.glyphicon-object-align-left:before{content:"\e247"}.glyphicon-object-align-vertical:before{content:"\e248"}.glyphicon-object-align-right:before{content:"\e249"}.glyphicon-triangle-right:before{content:"\e250"}.glyphicon-triangle-left:before{content:"\e251"}.glyphicon-triangle-bottom:before{content:"\e252"}.glyphicon-triangle-top:before{content:"\e253"}.glyphicon-console:before{content:"\e254"}.glyphicon-superscript:before{content:"\e255"}.glyphicon-subscript:before{content:"\e256"}.glyphicon-menu-left:before{content:"\e257"}.glyphicon-menu-right:before{content:"\e258"}.glyphicon-menu-down:before{content:"\e259"}.glyphicon-menu-up:before{content:"\e260"}\*{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box}:after,:before{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box}html{font-size:10px;-webkit-tap-highlight-color:rgba(0,0,0,0)}body{font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:14px;line-height:1.42857143;color:#333;background-color:#fff}button,input,select,textarea{font-family:inherit;font-size:inherit;line-height:inherit}a{color:#337ab7;text-decoration:none}a:focus,a:hover{color:#23527c;text-decoration:underline}a:focus{outline:thin dotted;outline:5px auto -webkit-focus-ring-color;outline-offset:-2px}figure{margin:0}img{vertical-align:middle}.carousel-inner>.item>a>img,.carousel-inner>.item>img,.img-responsive,.thumbnail a>img,.thumbnail>img{display:block;max-width:100%;height:auto}.img-rounded{border-radius:6px}.img-thumbnail{display:inline-block;max-width:100%;height:auto;padding:4px;line-height:1.42857143;background-color:#fff;border:1px solid #ddd;border-radius:4px;-webkit-transition:all .2s ease-in-out;-o-transition:all .2s ease-in-out;transition:all .2s ease-in-out}.img-circle{border-radius:50%}hr{margin-top:20px;margin-bottom:20px;border:0;border-top:1px solid #eee}.sr-only{position:absolute;width:1px;height:1px;padding:0;margin:-1px;overflow:hidden;clip:rect(0,0,0,0);border:0}.sr-only-focusable:active,.sr-only-focusable:focus{position:static;width:auto;height:auto;margin:0;overflow:visible;clip:auto}.h1,.h2,.h3,.h4,.h5,.h6,h1,h2,h3,h4,h5,h6{font-family:inherit;font-weight:500;line-height:1.1;color:inherit}.h1 .small,.h1 small,.h2 .small,.h2 small,.h3 .small,.h3 small,.h4 .small,.h4 small,.h5 .small,.h5 small,.h6 .small,.h6 small,h1 .small,h1 small,h2 .small,h2 small,h3 .small,h3 small,h4 .small,h4 small,h5 .small,h5 small,h6 .small,h6 small{font-weight:400;line-height:1;color:#777}.h1,.h2,.h3,h1,h2,h3{margin-top:20px;margin-bottom:10px}.h1 .small,.h1 small,.h2 .small,.h2 small,.h3 .small,.h3 small,h1 .small,h1 small,h2 .small,h2 small,h3 .small,h3 small{font-size:65%}.h4,.h5,.h6,h4,h5,h6{margin-top:10px;margin-bottom:10px}.h4 .small,.h4 small,.h5 .small,.h5 small,.h6 .small,.h6 small,h4 .small,h4 small,h5 .small,h5 small,h6 .small,h6 small{font-size:75%}.h1,h1{font-size:36px}.h2,h2{font-size:30px}.h3,h3{font-size:24px}.h4,h4{font-size:18px}.h5,h5{font-size:14px}.h6,h6{font-size:12px}p{margin:0 0 10px}

.lead{margin-left:10px;margin-bottom:20px;font-size:16px;font-weight:300;line-height:1.4}@media (min-width:768px){

.lead{font-size:18px}}

.lead a{color: black;}

.small,small{font-size:85%}.mark,mark{padding:.2em;background-color:#fcf8e3}.text-left{text-align:left}.text-right{text-align:right}.text-center{text-align:center}.text-justify{text-align:justify}.text-nowrap{white-space:nowrap}.text-lowercase{text-transform:lowercase}.text-uppercase{text-transform:uppercase}.text-capitalize{text-transform:capitalize}.text-muted{color:#777}.text-primary{color:#337ab7}a.text-primary:hover{color:#286090}.text-success{color:#3c763d}a.text-success:hover{color:#2b542c}.text-info{color:#31708f}a.text-info:hover{color:#245269}.text-warning{color:#8a6d3b}a.text-warning:hover{color:#66512c}.text-danger{color:#a94442}a.text-danger:hover{color:#843534}.bg-primary{color:#fff;background-color:#337ab7}a.bg-primary:hover{background-color:#286090}.bg-success{background-color:#dff0d8}a.bg-success:hover{background-color:#c1e2b3}.bg-info{background-color:#d9edf7}a.bg-info:hover{background-color:#afd9ee}.bg-warning{background-color:#fcf8e3}a.bg-warning:hover{background-color:#f7ecb5}.bg-danger{background-color:#f2dede}a.bg-danger:hover{background-color:#e4b9b9}.page-header{padding-bottom:9px;margin:40px 0 20px;border-bottom:1px solid #eee}ol,ul{margin-top:0;margin-bottom:10px}ol ol,ol ul,ul ol,ul ul{margin-bottom:0}.list-unstyled{padding-left:0;list-style:none}.list-inline{padding-left:0;margin-left:-5px;list-style:none}.list-inline>li{display:inline-block;padding-right:5px;padding-left:5px}dl{margin-top:0;margin-bottom:20px}dd,dt{line-height:1.42857143}dt{font-weight:700}dd{margin-left:0}@media (min-width:768px){.dl-horizontal dt{float:left;width:160px;overflow:hidden;clear:left;text-align:right;text-overflow:ellipsis;white-space:nowrap}.dl-horizontal dd{margin-left:180px}}abbr[data-original-title],abbr[title]{cursor:help;border-bottom:1px dotted #777}.initialism{font-size:90%;text-transform:uppercase}blockquote{padding:10px 20px;margin:0 0 20px;font-size:17.5px;border-left:5px solid #eee}blockquote ol:last-child,blockquote p:last-child,blockquote ul:last-child{margin-bottom:0}blockquote .small,blockquote footer,blockquote small{display:block;font-size:80%;line-height:1.42857143;color:#777}blockquote .small:before,blockquote footer:before,blockquote small:before{content:'\2014 \00A0'}.blockquote-reverse,blockquote.pull-right{padding-right:15px;padding-left:0;text-align:right;border-right:5px solid #eee;border-left:0}.blockquote-reverse .small:before,.blockquote-reverse footer:before,.blockquote-reverse small:before,blockquote.pull-right .small:before,blockquote.pull-right footer:before,blockquote.pull-right small:before{content:''}.blockquote-reverse .small:after,.blockquote-reverse footer:after,.blockquote-reverse small:after,blockquote.pull-right .small:after,blockquote.pull-right footer:after,blockquote.pull-right small:after{content:'\00A0 \2014'}address{margin-bottom:20px;font-style:normal;line-height:1.42857143}code,kbd,pre,samp{font-family:Menlo,Monaco,Consolas,"Courier New",monospace}code{padding:2px 4px;font-size:90%;color:#c7254e;background-color:#f9f2f4;border-radius:4px}kbd{padding:2px 4px;font-size:90%;color:#fff;background-color:#333;border-radius:3px;-webkit-box-shadow:inset 0 -1px 0 rgba(0,0,0,.25);box-shadow:inset 0 -1px 0 rgba(0,0,0,.25)}kbd kbd{padding:0;font-size:100%;font-weight:700;-webkit-box-shadow:none;box-shadow:none}pre{display:block;padding:9.5px;margin:0 0 10px;font-size:13px;line-height:1.42857143;color:#333;word-break:break-all;word-wrap:break-word;background-color:#f5f5f5;border:1px solid #ccc;border-radius:4px}pre code{padding:0;font-size:inherit;color:inherit;white-space:pre-wrap;background-color:transparent;border-radius:0}.pre-scrollable{max-height:340px;overflow-y:scroll}.container{padding-right:15px;padding-left:15px;margin-right:auto;margin-left:auto}@media (min-width:768px){.container{width:750px}}@media (min-width:992px){.container{width:970px}}@media (min-width:1200px){.container{width:1170px}}.container-fluid{padding-right:15px;padding-left:15px;margin-right:auto;margin-left:auto}.row{margin-right:-15px;margin-left:-15px}.col-lg-1,.col-lg-10,.col-lg-11,.col-lg-12,.col-lg-2,.col-lg-3,.col-lg-4,.col-lg-5,.col-lg-6,.col-lg-7,.col-lg-8,.col-lg-9,.col-md-1,.col-md-10,.col-md-11,.col-md-12,.col-md-2,.col-md-3,.col-md-4,.col-md-5,.col-md-6,.col-md-7,.col-md-8,.col-md-9,.col-sm-1,.col-sm-10,.col-sm-11,.col-sm-12,.col-sm-2,.col-sm-3,.col-sm-4,.col-sm-5,.col-sm-6,.col-sm-7,.col-sm-8,.col-sm-9,.col-xs-1,.col-xs-10,.col-xs-11,.col-xs-12,.col-xs-2,.col-xs-3,.col-xs-4,.col-xs-5,.col-xs-6,.col-xs-7,.col-xs-8,.col-xs-9{position:relative;min-height:1px;padding-right:15px;padding-left:15px}.col-xs-1,.col-xs-10,.col-xs-11,.col-xs-12,.col-xs-2,.col-xs-3,.col-xs-4,.col-xs-5,.col-xs-6,.col-xs-7,.col-xs-8,.col-xs-9{float:left}.col-xs-12{width:100%}.col-xs-11{width:91.66666667%}.col-xs-10{width:83.33333333%}.col-xs-9{width:75%}.col-xs-8{width:66.66666667%}.col-xs-7{width:58.33333333%}.col-xs-6{width:50%}.col-xs-5{width:41.66666667%}.col-xs-4{width:33.33333333%}.col-xs-3{width:25%}.col-xs-2{width:16.66666667%}.col-xs-1{width:8.33333333%}.col-xs-pull-12{right:100%}.col-xs-pull-11{right:91.66666667%}.col-xs-pull-10{right:83.33333333%}.col-xs-pull-9{right:75%}

.col-xs-pull-8{right:66.66666667%}.col-xs-pull-7{right:58.33333333%}.col-xs-pull-6{right:50%}.col-xs-pull-5{right:41.66666667%}.col-xs-pull-4{right:33.33333333%}.col-xs-pull-3{right:25%}.col-xs-pull-2{right:16.66666667%}.col-xs-pull-1{right:8.33333333%}.col-xs-pull-0{right:auto}.col-xs-push-12{left:100%}.col-xs-push-11{left:91.66666667%}.col-xs-push-10{left:83.33333333%}.col-xs-push-9{left:75%}.col-xs-push-8{left:66.66666667%}.col-xs-push-7{left:58.33333333%}.col-xs-push-6{left:50%}.col-xs-push-5{left:41.66666667%}.col-xs-push-4{left:33.33333333%}.col-xs-push-3{left:25%}.col-xs-push-2{left:16.66666667%}.col-xs-push-1{left:8.33333333%}.col-xs-push-0{left:auto}.col-xs-offset-12{margin-left:100%}.col-xs-offset-11{margin-left:91.66666667%}.col-xs-offset-10{margin-left:83.33333333%}.col-xs-offset-9{margin-left:75%}.col-xs-offset-8{margin-left:66.66666667%}.col-xs-offset-7{margin-left:58.33333333%}.col-xs-offset-6{margin-left:50%}.col-xs-offset-5{margin-left:41.66666667%}.col-xs-offset-4{margin-left:33.33333333%}.col-xs-offset-3{margin-left:25%}.col-xs-offset-2{margin-left:16.66666667%}.col-xs-offset-1{margin-left:8.33333333%}.col-xs-offset-0{margin-left:0}@media (min-width:768px){.col-sm-1,.col-sm-10,.col-sm-11,.col-sm-12,.col-sm-2,.col-sm-3,.col-sm-4,.col-sm-5,.col-sm-6,.col-sm-7,.col-sm-8,.col-sm-9{float:left}.col-sm-12{width:100%}.col-sm-11{width:91.66666667%}.col-sm-10{width:83.33333333%}.col-sm-9{width:75%}.col-sm-8{width:66.66666667%}.col-sm-7{width:58.33333333%}.col-sm-6{width:50%}.col-sm-5{width:41.66666667%}.col-sm-4{width:33.33333333%}.col-sm-3{width:25%}.col-sm-2{width:16.66666667%}.col-sm-1{width:8.33333333%}.col-sm-pull-12{right:100%}.col-sm-pull-11{right:91.66666667%}.col-sm-pull-10{right:83.33333333%}.col-sm-pull-9{right:75%}.col-sm-pull-8{right:66.66666667%}.col-sm-pull-7{right:58.33333333%}.col-sm-pull-6{right:50%}.col-sm-pull-5{right:41.66666667%}.col-sm-pull-4{right:33.33333333%}.col-sm-pull-3{right:25%}.col-sm-pull-2{right:16.66666667%}.col-sm-pull-1{right:8.33333333%}.col-sm-pull-0{right:auto}.col-sm-push-12{left:100%}.col-sm-push-11{left:91.66666667%}.col-sm-push-10{left:83.33333333%}.col-sm-push-9{left:75%}.col-sm-push-8{left:66.66666667%}.col-sm-push-7{left:58.33333333%}.col-sm-push-6{left:50%}.col-sm-push-5{left:41.66666667%}.col-sm-push-4{left:33.33333333%}.col-sm-push-3{left:25%}.col-sm-push-2{left:16.66666667%}.col-sm-push-1{left:8.33333333%}.col-sm-push-0{left:auto}.col-sm-offset-12{margin-left:100%}.col-sm-offset-11{margin-left:91.66666667%}.col-sm-offset-10{margin-left:83.33333333%}.col-sm-offset-9{margin-left:75%}.col-sm-offset-8{margin-left:66.66666667%}.col-sm-offset-7{margin-left:58.33333333%}.col-sm-offset-6{margin-left:50%}.col-sm-offset-5{margin-left:41.66666667%}.col-sm-offset-4{margin-left:33.33333333%}.col-sm-offset-3{margin-left:25%}.col-sm-offset-2{margin-left:16.66666667%}.col-sm-offset-1{margin-left:8.33333333%}.col-sm-offset-0{margin-left:0}}@media (min-width:992px){.col-md-1,.col-md-10,.col-md-11,.col-md-12,.col-md-2,.col-md-3,.col-md-4,.col-md-5,.col-md-6,.col-md-7,.col-md-8,.col-md-9{float:left}.col-md-12{width:100%}.col-md-11{width:91.66666667%}.col-md-10{width:83.33333333%}.col-md-9{width:75%}.col-md-8{width:66.66666667%}.col-md-7{width:58.33333333%}.col-md-6{width:50%}.col-md-5{width:41.66666667%}.col-md-4{width:33.33333333%}.col-md-3{width:25%}.col-md-2{width:16.66666667%}.col-md-1{width:8.33333333%}.col-md-pull-12{right:100%}.col-md-pull-11{right:91.66666667%}.col-md-pull-10{right:83.33333333%}.col-md-pull-9{right:75%}.col-md-pull-8{right:66.66666667%}.col-md-pull-7{right:58.33333333%}.col-md-pull-6{right:50%}.col-md-pull-5{right:41.66666667%}.col-md-pull-4{right:33.33333333%}.col-md-pull-3{right:25%}.col-md-pull-2{right:16.66666667%}.col-md-pull-1{right:8.33333333%}.col-md-pull-0{right:auto}.col-md-push-12{left:100%}.col-md-push-11{left:91.66666667%}.col-md-push-10{left:83.33333333%}.col-md-push-9{left:75%}.col-md-push-8{left:66.66666667%}.col-md-push-7{left:58.33333333%}.col-md-push-6{left:50%}.col-md-push-5{left:41.66666667%}.col-md-push-4{left:33.33333333%}.col-md-push-3{left:25%}.col-md-push-2{left:16.66666667%}.col-md-push-1{left:8.33333333%}.col-md-push-0{left:auto}.col-md-offset-12{margin-left:100%}.col-md-offset-11{margin-left:91.66666667%}.col-md-offset-10{margin-left:83.33333333%}.col-md-offset-9{margin-left:75%}.col-md-offset-8{margin-left:66.66666667%}.col-md-offset-7{margin-left:58.33333333%}.col-md-offset-6{margin-left:50%}.col-md-offset-5{margin-left:41.66666667%}.col-md-offset-4{margin-left:33.33333333%}.col-md-offset-3{margin-left:25%}.col-md-offset-2{margin-left:16.66666667%}.col-md-offset-1{margin-left:8.33333333%}.col-md-offset-0{margin-left:0}}@media (min-width:1200px){.col-lg-1,.col-lg-10,.col-lg-11,.col-lg-12,.col-lg-2,.col-lg-3,.col-lg-4,.col-lg-5,.col-lg-6,.col-lg-7,.col-lg-8,.col-lg-9{float:left}.col-lg-12{width:100%}.col-lg-11{width:91.66666667%}.col-lg-10{width:83.33333333%}.col-lg-9{width:75%}.col-lg-8{width:66.66666667%}.col-lg-7{width:58.33333333%}.col-lg-6{width:50%}.col-lg-5{width:41.66666667%}.col-lg-4{width:33.33333333%}.col-lg-3{width:25%}.col-lg-2{width:16.66666667%}.col-lg-1{width:8.33333333%}.col-lg-pull-12{right:100%}.col-lg-pull-11{right:91.66666667%}.col-lg-pull-10{right:83.33333333%}.col-lg-pull-9{right:75%}.col-lg-pull-8{right:66.66666667%}.col-lg-pull-7{right:58.33333333%}.col-lg-pull-6{right:50%}.col-lg-pull-5{right:41.66666667%}.col-lg-pull-4{right:33.33333333%}.col-lg-pull-3{right:25%}.col-lg-pull-2{right:16.66666667%}.col-lg-pull-1{right:8.33333333%}.col-lg-pull-0{right:auto}.col-lg-push-12{left:100%}.col-lg-push-11{left:91.66666667%}.col-lg-push-10{left:83.33333333%}.col-lg-push-9{left:75%}.col-lg-push-8{left:66.66666667%}.col-lg-push-7{left:58.33333333%}.col-lg-push-6{left:50%}.col-lg-push-5{left:41.66666667%}.col-lg-push-4{left:33.33333333%}.col-lg-push-3{left:25%}.col-lg-push-2{left:16.66666667%}.col-lg-push-1{left:8.33333333%}.col-lg-push-0{left:auto}.col-lg-offset-12{margin-left:100%}.col-lg-offset-11{margin-left:91.66666667%}.col-lg-offset-10{margin-left:83.33333333%}.col-lg-offset-9{margin-left:75%}.col-lg-offset-8{margin-left:66.66666667%}.col-lg-offset-7{margin-left:58.33333333%}.col-lg-offset-6{margin-left:50%}.col-lg-offset-5{margin-left:41.66666667%}.col-lg-offset-4{margin-left:33.33333333%}.col-lg-offset-3{margin-left:25%}.col-lg-offset-2{margin-left:16.66666667%}.col-lg-offset-1{margin-left:8.33333333%}.col-lg-offset-0{margin-left:0}}table{background-color:transparent}caption{padding-top:8px;padding-bottom:8px;color:#777;text-align:left}th{text-align:left}.table{width:100%;max-width:100%;margin-bottom:20px}.table>tbody>tr>td,.table>tbody>tr>th,.table>tfoot>tr>td,.table>tfoot>tr>th,.table>thead>tr>td,.table>thead>tr>th{padding:8px;line-height:1.42857143;vertical-align:top;border-top:1px solid #ddd}.table>thead>tr>th{vertical-align:bottom;border-bottom:2px solid #ddd}.table>caption+thead>tr:first-child>td,.table>caption+thead>tr:first-child>th,.table>colgroup+thead>tr:first-child>td,.table>colgroup+thead>tr:first-child>th,.table>thead:first-child>tr:first-child>td,.table>thead:first-child>tr:first-child>th{border-top:0}.table>tbody+tbody{border-top:2px solid #ddd}.table .table{background-color:#fff}.table-condensed>tbody>tr>td,.table-condensed>tbody>tr>th,.table-condensed>tfoot>tr>td,.table-condensed>tfoot>tr>th,.table-condensed>thead>tr>td,.table-condensed>thead>tr>th{padding:5px}.table-bordered{border:1px solid #ddd}.table-bordered>tbody>tr>td,.table-bordered>tbody>tr>th,.table-bordered>tfoot>tr>td,.table-bordered>tfoot>tr>th,.table-bordered>thead>tr>td,.table-bordered>thead>tr>th{border:1px solid #ddd}.table-bordered>thead>tr>td,.table-bordered>thead>tr>th{border-bottom-width:2px}

.table-striped>tbody>tr:nth-of-type(odd){background-color:#f9f9f9}

.table-hover>tbody>tr:hover{background-color:#f5f5f5}table col[class\*=col-]{position:static;display:table-column;float:none}table td[class\*=col-],table th[class\*=col-]{position:static;display:table-cell;float:none}.table>tbody>tr.active>td,.table>tbody>tr.active>th,.table>tbody>tr>td.active,.table>tbody>tr>th.active,.table>tfoot>tr.active>td,.table>tfoot>tr.active>th,.table>tfoot>tr>td.active,.table>tfoot>tr>th.active,.table>thead>tr.active>td,.table>thead>tr.active>th,.table>thead>tr>td.active,.table>thead>tr>th.active{background-color:#f5f5f5}.table-hover>tbody>tr.active:hover>td,.table-hover>tbody>tr.active:hover>th,.table-hover>tbody>tr:hover>.active,.table-hover>tbody>tr>td.active:hover,.table-hover>tbody>tr>th.active:hover{background-color:#e8e8e8}.table>tbody>tr.success>td,.table>tbody>tr.success>th,.table>tbody>tr>td.success,.table>tbody>tr>th.success,.table>tfoot>tr.success>td,.table>tfoot>tr.success>th,.table>tfoot>tr>td.success,.table>tfoot>tr>th.success,.table>thead>tr.success>td,.table>thead>tr.success>th,.table>thead>tr>td.success,.table>thead>tr>th.success{background-color:#dff0d8}.table-hover>tbody>tr.success:hover>td,.table-hover>tbody>tr.success:hover>th,.table-hover>tbody>tr:hover>.success,.table-hover>tbody>tr>td.success:hover,.table-hover>tbody>tr>th.success:hover{background-color:#d0e9c6}.table>tbody>tr.info>td,.table>tbody>tr.info>th,.table>tbody>tr>td.info,.table>tbody>tr>th.info,.table>tfoot>tr.info>td,.table>tfoot>tr.info>th,.table>tfoot>tr>td.info,.table>tfoot>tr>th.info,.table>thead>tr.info>td,.table>thead>tr.info>th,.table>thead>tr>td.info,.table>thead>tr>th.info{background-color:#d9edf7}.table-hover>tbody>tr.info:hover>td,.table-hover>tbody>tr.info:hover>th,.table-hover>tbody>tr:hover>.info,.table-hover>tbody>tr>td.info:hover,.table-hover>tbody>tr>th.info:hover{background-color:#c4e3f3}.table>tbody>tr.warning>td,.table>tbody>tr.warning>th,.table>tbody>tr>td.warning,.table>tbody>tr>th.warning,.table>tfoot>tr.warning>td,.table>tfoot>tr.warning>th,.table>tfoot>tr>td.warning,.table>tfoot>tr>th.warning,.table>thead>tr.warning>td,.table>thead>tr.warning>th,.table>thead>tr>td.warning,.table>thead>tr>th.warning{background-color:#fcf8e3}.table-hover>tbody>tr.warning:hover>td,.table-hover>tbody>tr.warning:hover>th,.table-hover>tbody>tr:hover>.warning,.table-hover>tbody>tr>td.warning:hover,.table-hover>tbody>tr>th.warning:hover{background-color:#faf2cc}.table>tbody>tr.danger>td,.table>tbody>tr.danger>th,.table>tbody>tr>td.danger,.table>tbody>tr>th.danger,.table>tfoot>tr.danger>td,.table>tfoot>tr.danger>th,.table>tfoot>tr>td.danger,.table>tfoot>tr>th.danger,.table>thead>tr.danger>td,.table>thead>tr.danger>th,.table>thead>tr>td.danger,.table>thead>tr>th.danger{background-color:#f2dede}.table-hover>tbody>tr.danger:hover>td,.table-hover>tbody>tr.danger:hover>th,.table-hover>tbody>tr:hover>.danger,.table-hover>tbody>tr>td.danger:hover,.table-hover>tbody>tr>th.danger:hover{background-color:#ebcccc}.table-responsive{min-height:.01%;overflow-x:auto}@media screen and (max-width:767px){.table-responsive{width:100%;margin-bottom:15px;overflow-y:hidden;-ms-overflow-style:-ms-autohiding-scrollbar;border:1px solid #ddd}.table-responsive>.table{margin-bottom:0}.table-responsive>.table>tbody>tr>td,.table-responsive>.table>tbody>tr>th,.table-responsive>.table>tfoot>tr>td,.table-responsive>.table>tfoot>tr>th,.table-responsive>.table>thead>tr>td,.table-responsive>.table>thead>tr>th{white-space:nowrap}.table-responsive>.table-bordered{border:0}.table-responsive>.table-bordered>tbody>tr>td:first-child,.table-responsive>.table-bordered>tbody>tr>th:first-child,.table-responsive>.table-bordered>tfoot>tr>td:first-child,.table-responsive>.table-bordered>tfoot>tr>th:first-child,.table-responsive>.table-bordered>thead>tr>td:first-child,.table-responsive>.table-bordered>thead>tr>th:first-child{border-left:0}.table-responsive>.table-bordered>tbody>tr>td:last-child,.table-responsive>.table-bordered>tbody>tr>th:last-child,.table-responsive>.table-bordered>tfoot>tr>td:last-child,.table-responsive>.table-bordered>tfoot>tr>th:last-child,.table-responsive>.table-bordered>thead>tr>td:last-child,.table-responsive>.table-bordered>thead>tr>th:last-child{border-right:0}.table-responsive>.table-bordered>tbody>tr:last-child>td,.table-responsive>.table-bordered>tbody>tr:last-child>th,.table-responsive>.table-bordered>tfoot>tr:last-child>td,.table-responsive>.table-bordered>tfoot>tr:last-child>th{border-bottom:0}}fieldset{min-width:0;padding:0;margin:0;border:0}legend{display:block;width:100%;padding:0;margin-bottom:20px;font-size:21px;line-height:inherit;color:#333;border:0;border-bottom:1px solid #e5e5e5}label{display:inline-block;max-width:100%;margin-bottom:5px;font-weight:700}input[type=search]{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box}input[type=checkbox],input[type=radio]{margin:4px 0 0;margin-top:1px \9;line-height:normal}input[type=file]{display:block}input[type=range]{display:block;width:100%}select[multiple],select[size]{height:auto}input[type=file]:focus,input[type=checkbox]:focus,input[type=radio]:focus{outline:thin dotted;outline:5px auto -webkit-focus-ring-color;outline-offset:-2px}output{display:block;padding-top:7px;font-size:14px;line-height:1.42857143;color:#555}.form-control{display:block;width:100%;height:34px;padding:6px 12px;font-size:14px;line-height:1.42857143;color:#555;background-color:#fff;background-image:none;border:1px solid #ccc;border-radius:4px;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075);-webkit-transition:border-color ease-in-out .15s,-webkit-box-shadow ease-in-out .15s;-o-transition:border-color ease-in-out .15s,box-shadow ease-in-out .15s;transition:border-color ease-in-out .15s,box-shadow ease-in-out .15s}.form-control:focus{border-color:#66afe9;outline:0;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 8px rgba(102,175,233,.6);box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 8px rgba(102,175,233,.6)}.form-control::-moz-placeholder{color:#999;opacity:1}.form-control:-ms-input-placeholder{color:#999}.form-control::-webkit-input-placeholder{color:#999}.form-control[disabled],.form-control[readonly],fieldset[disabled] .form-control{cursor:not-allowed;background-color:#eee;opacity:1}textarea.form-control{height:auto}input[type=search]{-webkit-appearance:none}@media screen and (-webkit-min-device-pixel-ratio:0){input[type=date],input[type=time],input[type=datetime-local],input[type=month]{line-height:34px}.input-group-sm input[type=date],.input-group-sm input[type=time],.input-group-sm input[type=datetime-local],.input-group-sm input[type=month],input[type=date].input-sm,input[type=time].input-sm,input[type=datetime-local].input-sm,input[type=month].input-sm{line-height:30px}.input-group-lg input[type=date],.input-group-lg input[type=time],.input-group-lg input[type=datetime-local],.input-group-lg input[type=month],input[type=date].input-lg,input[type=time].input-lg,input[type=datetime-local].input-lg,input[type=month].input-lg{line-height:46px}}.form-group{margin-bottom:15px}.checkbox,.radio{position:relative;display:block;margin-top:10px;margin-bottom:10px}.checkbox label,.radio label{min-height:20px;padding-left:20px;margin-bottom:0;font-weight:400;cursor:pointer}.checkbox input[type=checkbox],.checkbox-inline input[type=checkbox],.radio input[type=radio],.radio-inline input[type=radio]{position:absolute;margin-top:4px \9;margin-left:-20px}.checkbox+.checkbox,.radio+.radio{margin-top:-5px}.checkbox-inline,.radio-inline{display:inline-block;padding-left:20px;margin-bottom:0;font-weight:400;vertical-align:middle;cursor:pointer}.checkbox-inline+.checkbox-inline,.radio-inline+.radio-inline{margin-top:0;margin-left:10px}fieldset[disabled] input[type=checkbox],fieldset[disabled] input[type=radio],input[type=checkbox].disabled,input[type=checkbox][disabled],input[type=radio].disabled,input[type=radio][disabled]{cursor:not-allowed}.checkbox-inline.disabled,.radio-inline.disabled,fieldset[disabled] .checkbox-inline,fieldset[disabled] .radio-inline{cursor:not-allowed}.checkbox.disabled label,.radio.disabled label,fieldset[disabled] .checkbox label,fieldset[disabled] .radio label{cursor:not-allowed}.form-control-static{padding-top:7px;padding-bottom:7px;margin-bottom:0}.form-control-static.input-lg,.form-control-static.input-sm{padding-right:0;padding-left:0}.input-sm{height:30px;padding:5px 10px;font-size:12px;line-height:1.5;border-radius:3px}select.input-sm{height:30px;line-height:30px}select[multiple].input-sm,textarea.input-sm{height:auto}.form-group-sm .form-control{height:30px;padding:5px 10px;font-size:12px;line-height:1.5;border-radius:3px}select.form-group-sm .form-control{height:30px;line-height:30px}select[multiple].form-group-sm .form-control,textarea.form-group-sm .form-control{height:auto}.form-group-sm .form-control-static{height:30px;padding:5px 10px;font-size:12px;line-height:1.5}.input-lg{height:46px;padding:10px 16px;font-size:18px;line-height:1.3333333;border-radius:6px}select.input-lg{height:46px;line-height:46px}select[multiple].input-lg,textarea.input-lg{height:auto}.form-group-lg .form-control{height:46px;padding:10px 16px;font-size:18px;line-height:1.3333333;border-radius:6px}select.form-group-lg .form-control{height:46px;line-height:46px}select[multiple].form-group-lg .form-control,textarea.form-group-lg .form-control{height:auto}.form-group-lg .form-control-static{height:46px;padding:10px 16px;font-size:18px;line-height:1.3333333}.has-feedback{position:relative}.has-feedback .form-control{padding-right:42.5px}.form-control-feedback{position:absolute;top:0;right:0;z-index:2;display:block;width:34px;height:34px;line-height:34px;text-align:center;pointer-events:none}.input-lg+.form-control-feedback{width:46px;height:46px;line-height:46px}.input-sm+.form-control-feedback{width:30px;height:30px;line-height:30px}.has-success .checkbox,.has-success .checkbox-inline,.has-success .control-label,.has-success .help-block,.has-success .radio,.has-success .radio-inline,.has-success.checkbox label,.has-success.checkbox-inline label,.has-success.radio label,.has-success.radio-inline label{color:#3c763d}.has-success .form-control{border-color:#3c763d;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-success .form-control:focus{border-color:#2b542c;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #67b168;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #67b168}.has-success .input-group-addon{color:#3c763d;background-color:#dff0d8;border-color:#3c763d}.has-success .form-control-feedback{color:#3c763d}.has-warning .checkbox,.has-warning .checkbox-inline,.has-warning .control-label,.has-warning .help-block,.has-warning .radio,.has-warning .radio-inline,.has-warning.checkbox label,.has-warning.checkbox-inline label,.has-warning.radio label,.has-warning.radio-inline label{color:#8a6d3b}.has-warning .form-control{border-color:#8a6d3b;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-warning .form-control:focus{border-color:#66512c;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #c0a16b;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #c0a16b}.has-warning .input-group-addon{color:#8a6d3b;background-color:#fcf8e3;border-color:#8a6d3b}.has-warning .form-control-feedback{color:#8a6d3b}.has-error .checkbox,.has-error .checkbox-inline,.has-error .control-label,.has-error .help-block,.has-error .radio,.has-error .radio-inline,.has-error.checkbox label,.has-error.checkbox-inline label,.has-error.radio label,.has-error.radio-inline label{color:#a94442}.has-error .form-control{border-color:#a94442;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-error .form-control:focus{border-color:#843534;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #ce8483;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #ce8483}.has-error .input-group-addon{color:#a94442;background-color:#f2dede;border-color:#a94442}.has-error .form-control-feedback{color:#a94442}.has-feedback label~.form-control-feedback{top:25px}.has-feedback label.sr-only~.form-control-feedback{top:0}.help-block{display:block;margin-top:5px;margin-bottom:10px;color:#737373}@media (min-width:768px){.form-inline .form-group{display:inline-block;margin-bottom:0;vertical-align:middle}.form-inline .form-control{display:inline-block;width:auto;vertical-align:middle}.form-inline .form-control-static{display:inline-block}.form-inline .input-group{display:inline-table;vertical-align:middle}.form-inline .input-group .form-control,.form-inline .input-group .input-group-addon,.form-inline .input-group .input-group-btn{width:auto}.form-inline .input-group>.form-control{width:100%}.form-inline .control-label{margin-bottom:0;vertical-align:middle}.form-inline .checkbox,.form-inline .radio{display:inline-block;margin-top:0;margin-bottom:0;vertical-align:middle}.form-inline .checkbox label,.form-inline .radio label{padding-left:0}.form-inline .checkbox input[type=checkbox],.form-inline .radio input[type=radio]{position:relative;margin-left:0}.form-inline .has-feedback .form-control-feedback{top:0}}.form-horizontal .checkbox,.form-horizontal .checkbox-inline,.form-horizontal .radio,.form-horizontal .radio-inline{padding-top:7px;margin-top:0;margin-bottom:0}.form-horizontal .checkbox,.form-horizontal .radio{min-height:27px}.form-horizontal .form-group{margin-right:-15px;margin-left:-15px}@media (min-width:768px){.form-horizontal .control-label{padding-top:7px;margin-bottom:0;text-align:right}}.form-horizontal .has-feedback .form-control-feedback{right:15px}@media (min-width:768px){.form-horizontal .form-group-lg .control-label{padding-top:14.33px}}@media (min-width:768px){.form-horizontal .form-group-sm .control-label{padding-top:6px}}.btn{display:inline-block;padding:6px 12px;margin-bottom:0;font-size:14px;font-weight:400;line-height:1.42857143;text-align:center;white-space:nowrap;vertical-align:middle;-ms-touch-action:manipulation;touch-action:manipulation;cursor:pointer;-webkit-user-select:none;-moz-user-select:none;-ms-user-select:none;user-select:none;background-image:none;border:1px solid transparent;border-radius:4px}.btn.active.focus,.btn.active:focus,.btn.focus,.btn:active.focus,.btn:active:focus,.btn:focus{outline:thin dotted;outline:5px auto -webkit-focus-ring-color;outline-offset:-2px}.btn.focus,.btn:focus,.btn:hover{color:#333;text-decoration:none}.btn.active,.btn:active{background-image:none;outline:0;-webkit-box-shadow:inset 0 3px 5px rgba(0,0,0,.125);box-shadow:inset 0 3px 5px rgba(0,0,0,.125)}.btn.disabled,.btn[disabled],fieldset[disabled] .btn{pointer-events:none;cursor:not-allowed;filter:alpha(opacity=65);-webkit-box-shadow:none;box-shadow:none;opacity:.65}.btn-default{color:#333;background-color:#fff;border-color:#ccc}.btn-default.active,.btn-default.focus,.btn-default:active,.btn-default:focus,.btn-default:hover,.open>.dropdown-toggle.btn-default{color:#333;background-color:#e6e6e6;border-color:#adadad}.btn-default.active,.btn-default:active,.open>.dropdown-toggle.btn-default{background-image:none}.btn-default.disabled,.btn-default.disabled.active,.btn-default.disabled.focus,.btn-default.disabled:active,.btn-default.disabled:focus,.btn-default.disabled:hover,.btn-default[disabled],.btn-default[disabled].active,.btn-default[disabled].focus,.btn-default[disabled]:active,.btn-default[disabled]:focus,.btn-default[disabled]:hover,fieldset[disabled] .btn-default,fieldset[disabled] .btn-default.active,fieldset[disabled] .btn-default.focus,fieldset[disabled] .btn-default:active,fieldset[disabled] .btn-default:focus,fieldset[disabled] .btn-default:hover{background-color:#fff;border-color:#ccc}.btn-default .badge{color:#fff;background-color:#333}.btn-primary{color:#fff;background-color:#337ab7;border-color:#2e6da4}.btn-primary.active,.btn-primary.focus,.btn-primary:active,.btn-primary:focus,.btn-primary:hover,.open>.dropdown-toggle.btn-primary{color:#fff;background-color:#286090;border-color:#204d74}.btn-primary.active,.btn-primary:active,.open>.dropdown-toggle.btn-primary{background-image:none}.btn-primary.disabled,.btn-primary.disabled.active,.btn-primary.disabled.focus,.btn-primary.disabled:active,.btn-primary.disabled:focus,.btn-primary.disabled:hover,.btn-primary[disabled],.btn-primary[disabled].active,.btn-primary[disabled].focus,.btn-primary[disabled]:active,.btn-primary[disabled]:focus,.btn-primary[disabled]:hover,fieldset[disabled] .btn-primary,fieldset[disabled] .btn-primary.active,fieldset[disabled] .btn-primary.focus,fieldset[disabled] .btn-primary:active,fieldset[disabled] .btn-primary:focus,fieldset[disabled] .btn-primary:hover{background-color:#337ab7;border-color:#2e6da4}.btn-primary .badge{color:#337ab7;background-color:#fff}.btn-success{color:#fff;background-color:#5cb85c;border-color:#4cae4c}.btn-success.active,.btn-success.focus,.btn-success:active,.btn-success:focus,.btn-success:hover,.open>.dropdown-toggle.btn-success{color:#fff;background-color:#449d44;border-color:#398439}.btn-success.active,.btn-success:active,.open>.dropdown-toggle.btn-success{background-image:none}.btn-success.disabled,.btn-success.disabled.active,.btn-success.disabled.focus,.btn-success.disabled:active,.btn-success.disabled:focus,.btn-success.disabled:hover,.btn-success[disabled],.btn-success[disabled].active,.btn-success[disabled].focus,.btn-success[disabled]:active,.btn-success[disabled]:focus,.btn-success[disabled]:hover,fieldset[disabled] .btn-success,fieldset[disabled] .btn-success.active,fieldset[disabled] .btn-success.focus,fieldset[disabled] .btn-success:active,fieldset[disabled] .btn-success:focus,fieldset[disabled] .btn-success:hover{background-color:#5cb85c;border-color:#4cae4c}.btn-success .badge{color:#5cb85c;background-color:#fff}.btn-info{color:#fff;background-color:#5bc0de;border-color:#46b8da}.btn-info.active,.btn-info.focus,.btn-info:active,.btn-info:focus,.btn-info:hover,.open>.dropdown-toggle.btn-info{color:#fff;background-color:#31b0d5;border-color:#269abc}.btn-info.active,.btn-info:active,.open>.dropdown-toggle.btn-info{background-image:none}.btn-info.disabled,.btn-info.disabled.active,.btn-info.disabled.focus,.btn-info.disabled:active,.btn-info.disabled:focus,.btn-info.disabled:hover,.btn-info[disabled],.btn-info[disabled].active,.btn-info[disabled].focus,.btn-info[disabled]:active,.btn-info[disabled]:focus,.btn-info[disabled]:hover,fieldset[disabled] .btn-info,fieldset[disabled] .btn-info.active,fieldset[disabled] .btn-info.focus,fieldset[disabled] .btn-info:active,fieldset[disabled] .btn-info:focus,fieldset[disabled] .btn-info:hover{background-color:#5bc0de;border-color:#46b8da}.btn-info .badge{color:#5bc0de;background-color:#fff}.btn-warning{color:#fff;background-color:#f0ad4e;border-color:#eea236}.btn-warning.active,.btn-warning.focus,.btn-warning:active,.btn-warning:focus,.btn-warning:hover,.open>.dropdown-toggle.btn-warning{color:#fff;background-color:#ec971f;border-color:#d58512}.btn-warning.active,.btn-warning:active,.open>.dropdown-toggle.btn-warning{background-image:none}.btn-warning.disabled,.btn-warning.disabled.active,.btn-warning.disabled.focus,.btn-warning.disabled:active,.btn-warning.disabled:focus,.btn-warning.disabled:hover,.btn-warning[disabled],.btn-warning[disabled].active,.btn-warning[disabled].focus,.btn-warning[disabled]:active,.btn-warning[disabled]:focus,.btn-warning[disabled]:hover,fieldset[disabled] .btn-warning,fieldset[disabled] .btn-warning.active,fieldset[disabled] .btn-warning.focus,fieldset[disabled] .btn-warning:active,fieldset[disabled] .btn-warning:focus,fieldset[disabled] .btn-warning:hover{background-color:#f0ad4e;border-color:#eea236}.btn-warning .badge{color:#f0ad4e;background-color:#fff}.btn-danger{color:#fff;background-color:#d9534f;border-color:#d43f3a}

.btn-danger.active,.btn-danger.focus,.btn-danger:active,.btn-danger:focus,.btn-danger:hover,.open>.dropdown-toggle.btn-danger{color:#fff;background-color:#c9302c;border-color:#ac2925}.btn-danger.active,.btn-danger:active,.open>.dropdown-toggle.btn-danger{background-image:none}.btn-danger.disabled,.btn-danger.disabled.active,.btn-danger.disabled.focus,.btn-danger.disabled:active,.btn-danger.disabled:focus,.btn-danger.disabled:hover,.btn-danger[disabled],.btn-danger[disabled].active,.btn-danger[disabled].focus,.btn-danger[disabled]:active,.btn-danger[disabled]:focus,.btn-danger[disabled]:hover,fieldset[disabled] .btn-danger,fieldset[disabled] .btn-danger.active,fieldset[disabled] .btn-danger.focus,fieldset[disabled] .btn-danger:active,fieldset[disabled] .btn-danger:focus,fieldset[disabled] .btn-danger:hover{background-color:#d9534f;border-color:#d43f3a}.btn-danger .badge{color:#d9534f;background-color:#fff}.btn-link{font-weight:400;color:#337ab7;border-radius:0}.btn-link,.btn-link.active,.btn-link:active,.btn-link[disabled],fieldset[disabled] .btn-link{background-color:transparent;-webkit-box-shadow:none;box-shadow:none}.btn-link,.btn-link:active,.btn-link:focus,.btn-link:hover{border-color:transparent}.btn-link:focus,.btn-link:hover{color:#23527c;text-decoration:underline;background-color:transparent}.btn-link[disabled]:focus,.btn-link[disabled]:hover,fieldset[disabled] .btn-link:focus,fieldset[disabled] .btn-link:hover{color:#777;text-decoration:none}.btn-group-lg>.btn,.btn-lg{padding:10px 16px;font-size:18px;line-height:1.3333333;border-radius:6px}.btn-group-sm>.btn,.btn-sm{padding:5px 10px;font-size:12px;line-height:1.5;border-radius:3px}.btn-group-xs>.btn,.btn-xs{padding:1px 5px;font-size:12px;line-height:1.5;border-radius:3px}.btn-block{display:block;width:100%}.btn-block+.btn-block{margin-top:5px}input[type=button].btn-block,input[type=reset].btn-block,input[type=submit].btn-block{width:100%}.fade{opacity:0;-webkit-transition:opacity .15s linear;-o-transition:opacity .15s linear;transition:opacity .15s linear}.fade.in{opacity:1}.collapse{display:none;visibility:hidden}.collapse.in{display:block;visibility:visible}tr.collapse.in{display:table-row}tbody.collapse.in{display:table-row-group}.collapsing{position:relative;height:0;overflow:hidden;-webkit-transition-timing-function:ease;-o-transition-timing-function:ease;transition-timing-function:ease;-webkit-transition-duration:.35s;-o-transition-duration:.35s;transition-duration:.35s;-webkit-transition-property:height,visibility;-o-transition-property:height,visibility;transition-property:height,visibility}.caret{display:inline-block;width:0;height:0;margin-left:2px;vertical-align:middle;border-top:4px solid;border-right:4px solid transparent;border-left:4px solid transparent}.dropdown,.dropup{position:relative}.dropdown-toggle:focus{outline:0}.dropdown-menu{position:absolute;top:100%;left:0;z-index:1000;display:none;float:left;min-width:160px;padding:5px 0;margin:2px 0 0;font-size:14px;text-align:left;list-style:none;background-color:#fff;-webkit-background-clip:padding-box;background-clip:padding-box;border:1px solid #ccc;border:1px solid rgba(0,0,0,.15);border-radius:4px;-webkit-box-shadow:0 6px 12px rgba(0,0,0,.175);box-shadow:0 6px 12px rgba(0,0,0,.175)}.dropdown-menu.pull-right{right:0;left:auto}.dropdown-menu .divider{height:1px;margin:9px 0;overflow:hidden;background-color:#e5e5e5}.dropdown-menu>li>a{display:block;padding:3px 20px;clear:both;font-weight:400;line-height:1.42857143;color:#333;white-space:nowrap}.dropdown-menu>li>a:focus,.dropdown-menu>li>a:hover{color:#262626;text-decoration:none;background-color:#f5f5f5}.dropdown-menu>.active>a,.dropdown-menu>.active>a:focus,.dropdown-menu>.active>a:hover{color:#fff;text-decoration:none;background-color:#337ab7;outline:0}.dropdown-menu>.disabled>a,.dropdown-menu>.disabled>a:focus,.dropdown-menu>.disabled>a:hover{color:#777}.dropdown-menu>.disabled>a:focus,.dropdown-menu>.disabled>a:hover{text-decoration:none;cursor:not-allowed;background-color:transparent;background-image:none;filter:progid:DXImageTransform.Microsoft.gradient(enabled=false)}.open>.dropdown-menu{display:block}.open>a{outline:0}.dropdown-menu-right{right:0;left:auto}.dropdown-menu-left{right:auto;left:0}.dropdown-header{display:block;padding:3px 20px;font-size:12px;line-height:1.42857143;color:#777;white-space:nowrap}.dropdown-backdrop{position:fixed;top:0;right:0;bottom:0;left:0;z-index:990}.pull-right>.dropdown-menu{right:0;left:auto}.dropup .caret,.navbar-fixed-bottom .dropdown .caret{content:"";border-top:0;border-bottom:4px solid}.dropup .dropdown-menu,.navbar-fixed-bottom .dropdown .dropdown-menu{top:auto;bottom:100%;margin-bottom:2px}@media (min-width:768px){.navbar-right .dropdown-menu{right:0;left:auto}.navbar-right .dropdown-menu-left{right:auto;left:0}}.btn-group,.btn-group-vertical{position:relative;display:inline-block;vertical-align:middle}.btn-group-vertical>.btn,.btn-group>.btn{position:relative;float:left}.btn-group-vertical>.btn.active,.btn-group-vertical>.btn:active,.btn-group-vertical>.btn:focus,.btn-group-vertical>.btn:hover,.btn-group>.btn.active,.btn-group>.btn:active,.btn-group>.btn:focus,.btn-group>.btn:hover{z-index:2}.btn-group .btn+.btn,.btn-group .btn+.btn-group,.btn-group .btn-group+.btn,.btn-group .btn-group+.btn-group{margin-left:-1px}.btn-toolbar{margin-left:-5px}.btn-toolbar .btn-group,.btn-toolbar .input-group{float:left}.btn-toolbar>.btn,.btn-toolbar>.btn-group,.btn-toolbar>.input-group{margin-left:5px}.btn-group>.btn:not(:first-child):not(:last-child):not(.dropdown-toggle){border-radius:0}.btn-group>.btn:first-child{margin-left:0}.btn-group>.btn:first-child:not(:last-child):not(.dropdown-toggle){border-top-right-radius:0;border-bottom-right-radius:0}.btn-group>.btn:last-child:not(:first-child),.btn-group>.dropdown-toggle:not(:first-child){border-top-left-radius:0;border-bottom-left-radius:0}.btn-group>.btn-group{float:left}.btn-group>.btn-group:not(:first-child):not(:last-child)>.btn{border-radius:0}.btn-group>.btn-group:first-child:not(:last-child)>.btn:last-child,.btn-group>.btn-group:first-child:not(:last-child)>.dropdown-toggle{border-top-right-radius:0;border-bottom-right-radius:0}.btn-group>.btn-group:last-child:not(:first-child)>.btn:first-child{border-top-left-radius:0;border-bottom-left-radius:0}.btn-group .dropdown-toggle:active,.btn-group.open .dropdown-toggle{outline:0}.btn-group>.btn+.dropdown-toggle{padding-right:8px;padding-left:8px}.btn-group>.btn-lg+.dropdown-toggle{padding-right:12px;padding-left:12px}.btn-group.open .dropdown-toggle{-webkit-box-shadow:inset 0 3px 5px rgba(0,0,0,.125);box-shadow:inset 0 3px 5px rgba(0,0,0,.125)}.btn-group.open .dropdown-toggle.btn-link{-webkit-box-shadow:none;box-shadow:none}.btn .caret{margin-left:0}.btn-lg .caret{border-width:5px 5px 0;border-bottom-width:0}.dropup .btn-lg .caret{border-width:0 5px 5px}.btn-group-vertical>.btn,.btn-group-vertical>.btn-group,.btn-group-vertical>.btn-group>.btn{display:block;float:none;width:100%;max-width:100%}.btn-group-vertical>.btn-group>.btn{float:none}.btn-group-vertical>.btn+.btn,.btn-group-vertical>.btn+.btn-group,.btn-group-vertical>.btn-group+.btn,.btn-group-vertical>.btn-group+.btn-group{margin-top:-1px;margin-left:0}.btn-group-vertical>.btn:not(:first-child):not(:last-child){border-radius:0}.btn-group-vertical>.btn:first-child:not(:last-child){border-top-right-radius:4px;border-bottom-right-radius:0;border-bottom-left-radius:0}.btn-group-vertical>.btn:last-child:not(:first-child){border-top-left-radius:0;border-top-right-radius:0;border-bottom-left-radius:4px}.btn-group-vertical>.btn-group:not(:first-child):not(:last-child)>.btn{border-radius:0}.btn-group-vertical>.btn-group:first-child:not(:last-child)>.btn:last-child,.btn-group-vertical>.btn-group:first-child:not(:last-child)>.dropdown-toggle{border-bottom-right-radius:0;border-bottom-left-radius:0}.btn-group-vertical>.btn-group:last-child:not(:first-child)>.btn:first-child{border-top-left-radius:0;border-top-right-radius:0}.btn-group-justified{display:table;width:100%;table-layout:fixed;border-collapse:separate}.btn-group-justified>.btn,.btn-group-justified>.btn-group{display:table-cell;float:none;width:1%}.btn-group-justified>.btn-group .btn{width:100%}.btn-group-justified>.btn-group .dropdown-menu{left:auto}[data-toggle=buttons]>.btn input[type=checkbox],[data-toggle=buttons]>.btn input[type=radio],[data-toggle=buttons]>.btn-group>.btn input[type=checkbox],[data-toggle=buttons]>.btn-group>.btn input[type=radio]{position:absolute;clip:rect(0,0,0,0);pointer-events:none}.input-group{position:relative;display:table;border-collapse:separate}.input-group[class\*=col-]{float:none;padding-right:0;padding-left:0}

.input-group .form-control{position:relative;z-index:2;float:left;width:100%;margin-bottom:0}.input-group-lg>.form-control,.input-group-lg>.input-group-addon,.input-group-lg>.input-group-btn>.btn{height:46px;padding:10px 16px;font-size:18px;line-height:1.3333333;border-radius:6px}select.input-group-lg>.form-control,select.input-group-lg>.input-group-addon,select.input-group-lg>.input-group-btn>.btn{height:46px;line-height:46px}select[multiple].input-group-lg>.form-control,select[multiple].input-group-lg>.input-group-addon,select[multiple].input-group-lg>.input-group-btn>.btn,textarea.input-group-lg>.form-control,textarea.input-group-lg>.input-group-addon,textarea.input-group-lg>.input-group-btn>.btn{height:auto}.input-group-sm>.form-control,.input-group-sm>.input-group-addon,.input-group-sm>.input-group-btn>.btn{height:30px;padding:5px 10px;font-size:12px;line-height:1.5;border-radius:3px}select.input-group-sm>.form-control,select.input-group-sm>.input-group-addon,select.input-group-sm>.input-group-btn>.btn{height:30px;line-height:30px}select[multiple].input-group-sm>.form-control,select[multiple].input-group-sm>.input-group-addon,select[multiple].input-group-sm>.input-group-btn>.btn,textarea.input-group-sm>.form-control,textarea.input-group-sm>.input-group-addon,textarea.input-group-sm>.input-group-btn>.btn{height:auto}.input-group .form-control,.input-group-addon,.input-group-btn{display:table-cell}.input-group .form-control:not(:first-child):not(:last-child),.input-group-addon:not(:first-child):not(:last-child),.input-group-btn:not(:first-child):not(:last-child){border-radius:0}.input-group-addon,.input-group-btn{width:1%;white-space:nowrap;vertical-align:middle}.input-group-addon{padding:6px 12px;font-size:14px;font-weight:400;line-height:1;color:#555;text-align:center;background-color:#eee;border:1px solid #ccc;border-radius:4px}.input-group-addon.input-sm{padding:5px 10px;font-size:12px;border-radius:3px}.input-group-addon.input-lg{padding:10px 16px;font-size:18px;border-radius:6px}.input-group-addon input[type=checkbox],.input-group-addon input[type=radio]{margin-top:0}.input-group .form-control:first-child,.input-group-addon:first-child,.input-group-btn:first-child>.btn,.input-group-btn:first-child>.btn-group>.btn,.input-group-btn:first-child>.dropdown-toggle,.input-group-btn:last-child>.btn-group:not(:last-child)>.btn,.input-group-btn:last-child>.btn:not(:last-child):not(.dropdown-toggle){border-top-right-radius:0;border-bottom-right-radius:0}.input-group-addon:first-child{border-right:0}.input-group .form-control:last-child,.input-group-addon:last-child,.input-group-btn:first-child>.btn-group:not(:first-child)>.btn,.input-group-btn:first-child>.btn:not(:first-child),.input-group-btn:last-child>.btn,.input-group-btn:last-child>.btn-group>.btn,.input-group-btn:last-child>.dropdown-toggle{border-top-left-radius:0;border-bottom-left-radius:0}.input-group-addon:last-child{border-left:0}.input-group-btn{position:relative;font-size:0;white-space:nowrap}.input-group-btn>.btn{position:relative}.input-group-btn>.btn+.btn{margin-left:-1px}.input-group-btn>.btn:active,.input-group-btn>.btn:focus,.input-group-btn>.btn:hover{z-index:2}.input-group-btn:first-child>.btn,.input-group-btn:first-child>.btn-group{margin-right:-1px}.input-group-btn:last-child>.btn,.input-group-btn:last-child>.btn-group{margin-left:-1px}.nav{padding-left:0;margin-bottom:0;list-style:none}.nav>li{position:relative;display:block}.nav>li>a{position:relative;display:block;padding:10px 15px}.nav>li>a:focus,.nav>li>a:hover{text-decoration:none;background-color:#eee}.nav>li.disabled>a{color:#777}.nav>li.disabled>a:focus,.nav>li.disabled>a:hover{color:#777;text-decoration:none;cursor:not-allowed;background-color:transparent}.nav .open>a,.nav .open>a:focus,.nav .open>a:hover{background-color:#eee;border-color:#337ab7}.nav .nav-divider{height:1px;margin:9px 0;overflow:hidden;background-color:#e5e5e5}.nav>li>a>img{max-width:none}.nav-tabs{border-bottom:1px solid #ddd}.nav-tabs>li{float:left;margin-bottom:-1px}.nav-tabs>li>a{margin-right:2px;line-height:1.42857143;border:1px solid transparent;border-radius:4px 4px 0 0}.nav-tabs>li>a:hover{border-color:#eee #eee #ddd}.nav-tabs>li.active>a,.nav-tabs>li.active>a:focus,.nav-tabs>li.active>a:hover{color:#555;cursor:default;background-color:#fff;border:1px solid #ddd;border-bottom-color:transparent}.nav-tabs.nav-justified{width:100%;border-bottom:0}.nav-tabs.nav-justified>li{float:none}.nav-tabs.nav-justified>li>a{margin-bottom:5px;text-align:center}.nav-tabs.nav-justified>.dropdown .dropdown-menu{top:auto;left:auto}@media (min-width:768px){.nav-tabs.nav-justified>li{display:table-cell;width:1%}.nav-tabs.nav-justified>li>a{margin-bottom:0}}.nav-tabs.nav-justified>li>a{margin-right:0;border-radius:4px}.nav-tabs.nav-justified>.active>a,.nav-tabs.nav-justified>.active>a:focus,.nav-tabs.nav-justified>.active>a:hover{border:1px solid #ddd}@media (min-width:768px){.nav-tabs.nav-justified>li>a{border-bottom:1px solid #ddd;border-radius:4px 4px 0 0}.nav-tabs.nav-justified>.active>a,.nav-tabs.nav-justified>.active>a:focus,.nav-tabs.nav-justified>.active>a:hover{border-bottom-color:#fff}}.nav-pills>li{float:left}.nav-pills>li>a{border-radius:4px}.nav-pills>li+li{margin-left:2px}.nav-pills>li.active>a,.nav-pills>li.active>a:focus,.nav-pills>li.active>a:hover{color:#fff;background-color:#337ab7}.nav-stacked>li{float:none}.nav-stacked>li+li{margin-top:2px;margin-left:0}.nav-justified{width:100%}.nav-justified>li{float:none}.nav-justified>li>a{margin-bottom:5px;text-align:center}.nav-justified>.dropdown .dropdown-menu{top:auto;left:auto}@media (min-width:768px){.nav-justified>li{display:table-cell;width:1%}.nav-justified>li>a{margin-bottom:0}}.nav-tabs-justified{border-bottom:0}.nav-tabs-justified>li>a{margin-right:0;border-radius:4px}.nav-tabs-justified>.active>a,.nav-tabs-justified>.active>a:focus,.nav-tabs-justified>.active>a:hover{border:1px solid #ddd}@media (min-width:768px){.nav-tabs-justified>li>a{border-bottom:1px solid #ddd;border-radius:4px 4px 0 0}.nav-tabs-justified>.active>a,.nav-tabs-justified>.active>a:focus,.nav-tabs-justified>.active>a:hover{border-bottom-color:#fff}}.tab-content>.tab-pane{display:none;visibility:hidden}.tab-content>.active{display:block;visibility:visible}.nav-tabs .dropdown-menu{margin-top:-1px;border-top-left-radius:0;border-top-right-radius:0}.navbar{position:relative;min-height:50px;margin-bottom:20px;border:1px solid transparent}@media (min-width:768px){.navbar{border-radius:4px}}@media (min-width:768px){.navbar-header{float:left}}.navbar-collapse{padding-right:15px;padding-left:15px;overflow-x:visible;-webkit-overflow-scrolling:touch;border-top:1px solid transparent;-webkit-box-shadow:inset 0 1px 0 rgba(255,255,255,.1);box-shadow:inset 0 1px 0 rgba(255,255,255,.1)}.navbar-collapse.in{overflow-y:auto}@media (min-width:768px){.navbar-collapse{width:auto;border-top:0;-webkit-box-shadow:none;box-shadow:none}.navbar-collapse.collapse{display:block!important;height:auto!important;padding-bottom:0;overflow:visible!important;visibility:visible!important}.navbar-collapse.in{overflow-y:visible}.navbar-fixed-bottom .navbar-collapse,.navbar-fixed-top .navbar-collapse,.navbar-static-top .navbar-collapse{padding-right:0;padding-left:0}}.navbar-fixed-bottom .navbar-collapse,.navbar-fixed-top .navbar-collapse{max-height:340px}@media (max-device-width:480px) and (orientation:landscape){.navbar-fixed-bottom .navbar-collapse,.navbar-fixed-top .navbar-collapse{max-height:200px}}.container-fluid>.navbar-collapse,.container-fluid>.navbar-header,.container>.navbar-collapse,.container>.navbar-header{margin-right:-15px;margin-left:-15px}@media (min-width:768px){.container-fluid>.navbar-collapse,.container-fluid>.navbar-header,.container>.navbar-collapse,.container>.navbar-header{margin-right:0;margin-left:0}}.navbar-static-top{z-index:1000;border-width:0 0 1px}@media (min-width:768px){.navbar-static-top{border-radius:0}}.navbar-fixed-bottom,.navbar-fixed-top{position:fixed;right:0;left:0;z-index:1030}@media (min-width:768px){.navbar-fixed-bottom,.navbar-fixed-top{border-radius:0}}.navbar-fixed-top{top:0;border-width:0 0 1px}.navbar-fixed-bottom{bottom:0;margin-bottom:0;border-width:1px 0 0}.navbar-brand{float:left;height:50px;padding:15px 15px;font-size:18px;line-height:20px}.navbar-brand:focus,.navbar-brand:hover{text-decoration:none}.navbar-brand>img{display:block}@media (min-width:768px){.navbar>.container .navbar-brand,.navbar>.container-fluid .navbar-brand{margin-left:-15px}}.navbar-toggle{position:relative;float:right;padding:9px 10px;margin-top:8px;margin-right:15px;margin-bottom:8px;background-color:transparent;background-image:none;border:1px solid transparent;border-radius:4px}.navbar-toggle:focus{outline:0}.navbar-toggle .icon-bar{display:block;width:22px;height:2px;border-radius:1px}.navbar-toggle .icon-bar+.icon-bar{margin-top:4px}@media (min-width:768px){.navbar-toggle{display:none}}.navbar-nav{margin:7.5px -15px}.navbar-nav>li>a{padding-top:10px;padding-bottom:10px;line-height:20px}@media (max-width:767px){.navbar-nav .open .dropdown-menu{position:static;float:none;width:auto;margin-top:0;background-color:transparent;border:0;-webkit-box-shadow:none;box-shadow:none}.navbar-nav .open .dropdown-menu .dropdown-header,.navbar-nav .open .dropdown-menu>li>a{padding:5px 15px 5px 25px}.navbar-nav .open .dropdown-menu>li>a{line-height:20px}.navbar-nav .open .dropdown-menu>li>a:focus,.navbar-nav .open .dropdown-menu>li>a:hover{background-image:none}}@media (min-width:768px){.navbar-nav{float:left;margin:0}.navbar-nav>li{float:left}.navbar-nav>li>a{padding-top:15px;padding-bottom:15px}}.navbar-form{padding:10px 15px;margin-top:8px;margin-right:-15px;margin-bottom:8px;margin-left:-15px;border-top:1px solid transparent;border-bottom:1px solid transparent;-webkit-box-shadow:inset 0 1px 0 rgba(255,255,255,.1),0 1px 0 rgba(255,255,255,.1);box-shadow:inset 0 1px 0 rgba(255,255,255,.1),0 1px 0 rgba(255,255,255,.1)}@media (min-width:768px){.navbar-form .form-group{display:inline-block;margin-bottom:0;vertical-align:middle}.navbar-form .form-control{display:inline-block;width:auto;vertical-align:middle}.navbar-form .form-control-static{display:inline-block}.navbar-form .input-group{display:inline-table;vertical-align:middle}.navbar-form .input-group .form-control,.navbar-form .input-group .input-group-addon,.navbar-form .input-group .input-group-btn{width:auto}.navbar-form .input-group>.form-control{width:100%}.navbar-form .control-label{margin-bottom:0;vertical-align:middle}.navbar-form .checkbox,.navbar-form .radio{display:inline-block;margin-top:0;margin-bottom:0;vertical-align:middle}.navbar-form .checkbox label,.navbar-form .radio label{padding-left:0}.navbar-form .checkbox input[type=checkbox],.navbar-form .radio input[type=radio]{position:relative;margin-left:0}.navbar-form .has-feedback .form-control-feedback{top:0}}@media (max-width:767px){.navbar-form .form-group{margin-bottom:5px}.navbar-form .form-group:last-child{margin-bottom:0}}@media (min-width:768px){.navbar-form{width:auto;padding-top:0;padding-bottom:0;margin-right:0;margin-left:0;border:0;-webkit-box-shadow:none;box-shadow:none}}.navbar-nav>li>.dropdown-menu{margin-top:0;border-top-left-radius:0;border-top-right-radius:0}.navbar-fixed-bottom .navbar-nav>li>.dropdown-menu{margin-bottom:0;border-top-left-radius:4px;border-top-right-radius:4px;border-bottom-right-radius:0;border-bottom-left-radius:0}.navbar-btn{margin-top:8px;margin-bottom:8px}.navbar-btn.btn-sm{margin-top:10px;margin-bottom:10px}.navbar-btn.btn-xs{margin-top:14px;margin-bottom:14px}.navbar-text{margin-top:15px;margin-bottom:15px}@media (min-width:768px){.navbar-text{float:left;margin-right:15px;margin-left:15px}}@media (min-width:768px){.navbar-left{float:left!important}.navbar-right{float:right!important;margin-right:-15px}.navbar-right~.navbar-right{margin-right:0}}.navbar-default{background-color:#f8f8f8;border-color:#e7e7e7}.navbar-default .navbar-brand{color:#777}.navbar-default .navbar-brand:focus,.navbar-default .navbar-brand:hover{color:#5e5e5e;background-color:transparent}.navbar-default .navbar-text{color:#777}.navbar-default .navbar-nav>li>a{color:#777}.navbar-default .navbar-nav>li>a:focus,.navbar-default .navbar-nav>li>a:hover{color:#333;background-color:transparent}.navbar-default .navbar-nav>.active>a,.navbar-default .navbar-nav>.active>a:focus,.navbar-default .navbar-nav>.active>a:hover{color:#555;background-color:#e7e7e7}.navbar-default .navbar-nav>.disabled>a,.navbar-default .navbar-nav>.disabled>a:focus,.navbar-default .navbar-nav>.disabled>a:hover{color:#ccc;background-color:transparent}.navbar-default .navbar-toggle{border-color:#ddd}.navbar-default .navbar-toggle:focus,.navbar-default .navbar-toggle:hover{background-color:#ddd}.navbar-default .navbar-toggle .icon-bar{background-color:#888}.navbar-default .navbar-collapse,.navbar-default .navbar-form{border-color:#e7e7e7}.navbar-default .navbar-nav>.open>a,.navbar-default .navbar-nav>.open>a:focus,.navbar-default .navbar-nav>.open>a:hover{color:#555;background-color:#e7e7e7}@media (max-width:767px){.navbar-default .navbar-nav .open .dropdown-menu>li>a{color:#777}.navbar-default .navbar-nav .open .dropdown-menu>li>a:focus,.navbar-default .navbar-nav .open .dropdown-menu>li>a:hover{color:#333;background-color:transparent}.navbar-default .navbar-nav .open .dropdown-menu>.active>a,.navbar-default .navbar-nav .open .dropdown-menu>.active>a:focus,.navbar-default .navbar-nav .open .dropdown-menu>.active>a:hover{color:#555;background-color:#e7e7e7}.navbar-default .navbar-nav .open .dropdown-menu>.disabled>a,.navbar-default .navbar-nav .open .dropdown-menu>.disabled>a:focus,.navbar-default .navbar-nav .open .dropdown-menu>.disabled>a:hover{color:#ccc;background-color:transparent}}.navbar-default .navbar-link{color:#777}.navbar-default .navbar-link:hover{color:#333}.navbar-default .btn-link{color:#777}.navbar-default .btn-link:focus,.navbar-default .btn-link:hover{color:#333}.navbar-default .btn-link[disabled]:focus,.navbar-default .btn-link[disabled]:hover,fieldset[disabled] .navbar-default .btn-link:focus,fieldset[disabled] .navbar-default .btn-link:hover{color:#ccc}.navbar-inverse{background-color:#222;border-color:#080808}.navbar-inverse .navbar-brand{color:#9d9d9d}.navbar-inverse .navbar-brand:focus,.navbar-inverse .navbar-brand:hover{color:#fff;background-color:transparent}.navbar-inverse .navbar-text{color:#9d9d9d}.navbar-inverse .navbar-nav>li>a{color:#9d9d9d}.navbar-inverse .navbar-nav>li>a:focus,.navbar-inverse .navbar-nav>li>a:hover{color:#fff;background-color:transparent}.navbar-inverse .navbar-nav>.active>a,.navbar-inverse .navbar-nav>.active>a:focus,.navbar-inverse .navbar-nav>.active>a:hover{color:#fff;background-color:#080808}.navbar-inverse .navbar-nav>.disabled>a,.navbar-inverse .navbar-nav>.disabled>a:focus,.navbar-inverse .navbar-nav>.disabled>a:hover{color:#444;background-color:transparent}.navbar-inverse .navbar-toggle{border-color:#333}.navbar-inverse .navbar-toggle:focus,.navbar-inverse .navbar-toggle:hover{background-color:#333}.navbar-inverse .navbar-toggle .icon-bar{background-color:#fff}.navbar-inverse .navbar-collapse,.navbar-inverse .navbar-form{border-color:#101010}.navbar-inverse .navbar-nav>.open>a,.navbar-inverse .navbar-nav>.open>a:focus,.navbar-inverse .navbar-nav>.open>a:hover{color:#fff;background-color:#080808}@media (max-width:767px){.navbar-inverse .navbar-nav .open .dropdown-menu>.dropdown-header{border-color:#080808}.navbar-inverse .navbar-nav .open .dropdown-menu .divider{background-color:#080808}.navbar-inverse .navbar-nav .open .dropdown-menu>li>a{color:#9d9d9d}.navbar-inverse .navbar-nav .open .dropdown-menu>li>a:focus,.navbar-inverse .navbar-nav .open .dropdown-menu>li>a:hover{color:#fff;background-color:transparent}.navbar-inverse .navbar-nav .open .dropdown-menu>.active>a,.navbar-inverse .navbar-nav .open .dropdown-menu>.active>a:focus,.navbar-inverse .navbar-nav .open .dropdown-menu>.active>a:hover{color:#fff;background-color:#080808}.navbar-inverse .navbar-nav .open .dropdown-menu>.disabled>a,.navbar-inverse .navbar-nav .open .dropdown-menu>.disabled>a:focus,.navbar-inverse .navbar-nav .open .dropdown-menu>.disabled>a:hover{color:#444;background-color:transparent}}.navbar-inverse .navbar-link{color:#9d9d9d}.navbar-inverse .navbar-link:hover{color:#fff}.navbar-inverse .btn-link{color:#9d9d9d}.navbar-inverse .btn-link:focus,.navbar-inverse .btn-link:hover{color:#fff}.navbar-inverse .btn-link[disabled]:focus,.navbar-inverse .btn-link[disabled]:hover,fieldset[disabled] .navbar-inverse .btn-link:focus,fieldset[disabled] .navbar-inverse .btn-link:hover{color:#444}.breadcrumb{padding:8px 15px;margin-bottom:20px;list-style:none;background-color:#f5f5f5;border-radius:4px}.breadcrumb>li{display:inline-block}.breadcrumb>li+li:before{padding:0 5px;color:#ccc;content:"/\00a0"}.breadcrumb>.active{color:#777}.pagination{display:inline-block;padding-left:0;margin:20px 0;border-radius:4px}.pagination>li{display:inline}.pagination>li>a,.pagination>li>span{position:relative;float:left;padding:6px 12px;margin-left:-1px;line-height:1.42857143;color:#337ab7;text-decoration:none;background-color:#fff;border:1px solid #ddd}.pagination>li:first-child>a,.pagination>li:first-child>span{margin-left:0;border-top-left-radius:4px;border-bottom-left-radius:4px}.pagination>li:last-child>a,.pagination>li:last-child>span{border-top-right-radius:4px;border-bottom-right-radius:4px}.pagination>li>a:focus,.pagination>li>a:hover,.pagination>li>span:focus,.pagination>li>span:hover{color:#23527c;background-color:#eee;border-color:#ddd}.pagination>.active>a,.pagination>.active>a:focus,.pagination>.active>a:hover,.pagination>.active>span,.pagination>.active>span:focus,.pagination>.active>span:hover{z-index:2;color:#fff;cursor:default;background-color:#337ab7;border-color:#337ab7}.pagination>.disabled>a,.pagination>.disabled>a:focus,.pagination>.disabled>a:hover,.pagination>.disabled>span,.pagination>.disabled>span:focus,.pagination>.disabled>span:hover{color:#777;cursor:not-allowed;background-color:#fff;border-color:#ddd}.pagination-lg>li>a,.pagination-lg>li>span{padding:10px 16px;font-size:18px}.pagination-lg>li:first-child>a,.pagination-lg>li:first-child>span{border-top-left-radius:6px;border-bottom-left-radius:6px}.pagination-lg>li:last-child>a,.pagination-lg>li:last-child>span{border-top-right-radius:6px;border-bottom-right-radius:6px}.pagination-sm>li>a,.pagination-sm>li>span{padding:5px 10px;font-size:12px}.pagination-sm>li:first-child>a,.pagination-sm>li:first-child>span{border-top-left-radius:3px;border-bottom-left-radius:3px}.pagination-sm>li:last-child>a,.pagination-sm>li:last-child>span{border-top-right-radius:3px;border-bottom-right-radius:3px}.pager{padding-left:0;margin:20px 0;text-align:center;list-style:none}.pager li{display:inline}.pager li>a,.pager li>span{display:inline-block;padding:5px 14px;background-color:#fff;border:1px solid #ddd;border-radius:15px}.pager li>a:focus,.pager li>a:hover{text-decoration:none;background-color:#eee}.pager .next>a,.pager .next>span{float:right}.pager .previous>a,.pager .previous>span{float:left}.pager .disabled>a,.pager .disabled>a:focus,.pager .disabled>a:hover,.pager .disabled>span{color:#777;cursor:not-allowed;background-color:#fff}.label{display:inline;padding:.2em .6em .3em;font-size:75%;font-weight:700;line-height:1;color:#fff;text-align:center;white-space:nowrap;vertical-align:baseline;border-radius:.25em}a.label:focus,a.label:hover{color:#fff;text-decoration:none;cursor:pointer}.label:empty{display:none}.btn .label{position:relative;top:-1px}.label-default{background-color:#777}.label-default[href]:focus,.label-default[href]:hover{background-color:#5e5e5e}.label-primary{background-color:#337ab7}.label-primary[href]:focus,.label-primary[href]:hover{background-color:#286090}.label-success{background-color:#5cb85c}.label-success[href]:focus,.label-success[href]:hover{background-color:#449d44}.label-info{background-color:#5bc0de}.label-info[href]:focus,.label-info[href]:hover{background-color:#31b0d5}.label-warning{background-color:#f0ad4e}.label-warning[href]:focus,.label-warning[href]:hover{background-color:#ec971f}.label-danger{background-color:#d9534f}.label-danger[href]:focus,.label-danger[href]:hover{background-color:#c9302c}.badge{display:inline-block;min-width:10px;padding:3px 7px;font-size:12px;font-weight:700;line-height:1;color:#fff;text-align:center;white-space:nowrap;vertical-align:baseline;background-color:#777;border-radius:10px}.badge:empty{display:none}.btn .badge{position:relative;top:-1px}.btn-xs .badge{top:0;padding:1px 5px}a.badge:focus,a.badge:hover{color:#fff;text-decoration:none;cursor:pointer}.list-group-item.active>.badge,.nav-pills>.active>a>.badge{color:#337ab7;background-color:#fff}.list-group-item>.badge{float:right}.list-group-item>.badge+.badge{margin-right:5px}.nav-pills>li>a>.badge{margin-left:3px}.jumbotron{padding:30px 15px;margin-bottom:30px;color:inherit;background-color:#eee}.jumbotron .h1,.jumbotron h1{color:inherit}.jumbotron p{margin-bottom:15px;font-size:21px;font-weight:200}.jumbotron>hr{border-top-color:#d5d5d5}.container .jumbotron,.container-fluid .jumbotron{border-radius:6px}.jumbotron .container{max-width:100%}@media screen and (min-width:768px){.jumbotron{padding:48px 0}.container .jumbotron,.container-fluid .jumbotron{padding-right:60px;padding-left:60px}.jumbotron .h1,.jumbotron h1{font-size:63px}}.thumbnail{display:block;padding:4px;margin-bottom:20px;line-height:1.42857143;background-color:#fff;border:1px solid #ddd;border-radius:4px;-webkit-transition:border .2s ease-in-out;-o-transition:border .2s ease-in-out;transition:border .2s ease-in-out}.thumbnail a>img,.thumbnail>img{margin-right:auto;margin-left:auto}a.thumbnail.active,a.thumbnail:focus,a.thumbnail:hover{border-color:#337ab7}.thumbnail .caption{padding:9px;color:#333}.alert{padding:15px;margin-bottom:20px;border:1px solid transparent;border-radius:4px}.alert h4{margin-top:0;color:inherit}.alert .alert-link{font-weight:700}.alert>p,.alert>ul{margin-bottom:0}.alert>p+p{margin-top:5px}.alert-dismissable,.alert-dismissible{padding-right:35px}.alert-dismissable .close,.alert-dismissible .close{position:relative;top:-2px;right:-21px;color:inherit}.alert-success{color:#3c763d;background-color:#dff0d8;border-color:#d6e9c6}.alert-success hr{border-top-color:#c9e2b3}.alert-success .alert-link{color:#2b542c}.alert-info{color:#31708f;background-color:#d9edf7;border-color:#bce8f1}.alert-info hr{border-top-color:#a6e1ec}.alert-info .alert-link{color:#245269}.alert-warning{color:#8a6d3b;background-color:#fcf8e3;border-color:#faebcc}.alert-warning hr{border-top-color:#f7e1b5}.alert-warning .alert-link{color:#66512c}.alert-danger{color:#a94442;background-color:#f2dede;border-color:#ebccd1}.alert-danger hr{border-top-color:#e4b9c0}.alert-danger .alert-link{color:#843534}@-webkit-keyframes progress-bar-stripes{from{background-position:40px 0}to{background-position:0 0}}@-o-keyframes progress-bar-stripes{from{background-position:40px 0}to{background-position:0 0}}@keyframes progress-bar-stripes{from{background-position:40px 0}to{background-position:0 0}}.progress{height:20px;margin-bottom:20px;overflow:hidden;background-color:#f5f5f5;border-radius:4px;-webkit-box-shadow:inset 0 1px 2px rgba(0,0,0,.1);box-shadow:inset 0 1px 2px rgba(0,0,0,.1)}.progress-bar{float:left;width:0;height:100%;font-size:12px;line-height:20px;color:#fff;text-align:center;background-color:#337ab7;-webkit-box-shadow:inset 0 -1px 0 rgba(0,0,0,.15);box-shadow:inset 0 -1px 0 rgba(0,0,0,.15);-webkit-transition:width .6s ease;-o-transition:width .6s ease;transition:width .6s ease}.progress-bar-striped,.progress-striped .progress-bar{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);-webkit-background-size:40px 40px;background-size:40px 40px}.progress-bar.active,.progress.active .progress-bar{-webkit-animation:progress-bar-stripes 2s linear infinite;-o-animation:progress-bar-stripes 2s linear infinite;animation:progress-bar-stripes 2s linear infinite}.progress-bar-success{background-color:#5cb85c}.progress-striped .progress-bar-success{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent)}.progress-bar-info{background-color:#5bc0de}.progress-striped .progress-bar-info{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent)}.progress-bar-warning{background-color:#f0ad4e}.progress-striped .progress-bar-warning{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent)}.progress-bar-danger{background-color:#d9534f}.progress-striped .progress-bar-danger{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent)}.media{margin-top:15px}.media:first-child{margin-top:0}.media,.media-body{overflow:hidden;zoom:1}.media-body{width:10000px}.media-object{display:block}.media-right,.media>.pull-right{padding-left:10px}.media-left,.media>.pull-left{padding-right:10px}.media-body,.media-left,.media-right{display:table-cell;vertical-align:top}.media-middle{vertical-align:middle}.media-bottom{vertical-align:bottom}.media-heading{margin-top:0;margin-bottom:5px}.media-list{padding-left:0;list-style:none}.list-group{padding-left:0;margin-bottom:20px}.list-group-item{position:relative;display:block;padding:10px 15px;margin-bottom:-1px;background-color:#fff;border:1px solid #ddd}.list-group-item:first-child{border-top-left-radius:4px;border-top-right-radius:4px}.list-group-item:last-child{margin-bottom:0;border-bottom-right-radius:4px;border-bottom-left-radius:4px}a.list-group-item{color:#555}a.list-group-item .list-group-item-heading{color:#333}a.list-group-item:focus,a.list-group-item:hover{color:#555;text-decoration:none;background-color:#f5f5f5}.list-group-item.disabled,.list-group-item.disabled:focus,.list-group-item.disabled:hover{color:#777;cursor:not-allowed;background-color:#eee}.list-group-item.disabled .list-group-item-heading,.list-group-item.disabled:focus .list-group-item-heading,.list-group-item.disabled:hover .list-group-item-heading{color:inherit}.list-group-item.disabled .list-group-item-text,.list-group-item.disabled:focus .list-group-item-text,.list-group-item.disabled:hover .list-group-item-text{color:#777}.list-group-item.active,.list-group-item.active:focus,

.list-group-item.active:hover{z-index:2;color:#fff;background-color:#337ab7;border-color:#337ab7}

.list-group-item.active .list-group-item-heading,.list-group-item.active .list-group-item-heading>.small,.list-group-item.active .list-group-item-heading>small,.list-group-item.active:focus .list-group-item-heading,.list-group-item.active:focus .list-group-item-heading>.small,.list-group-item.active:focus

.list-group-item-heading>small,.list-group-item.active:hover

.list-group-item-heading,.list-group-item.active:hover .list-group-item-heading>.small,.list-group-item.active:hover .list-group-item-heading>small{color:inherit}.list-group-item.active .list-group-item-text,.list-group-item.active:focus .list-group-item-text,.list-group-item.active:hover .list-group-item-text{color:#c7ddef}.list-group-item-success{color:#3c763d;background-color:#dff0d8}a.list-group-item-success{color:#3c763d}a.list-group-item-success .list-group-item-heading{color:inherit}a.list-group-item-success:focus,a.list-group-item-success:hover{color:#3c763d;background-color:#d0e9c6}a.list-group-item-success.active,a.list-group-item-success.active:focus,a.list-group-item-success.active:hover{color:#fff;background-color:#3c763d;border-color:#3c763d}.list-group-item-info{color:#31708f;background-color:#d9edf7}a.list-group-item-info{color:#31708f}a.list-group-item-info .list-group-item-heading{color:inherit}a.list-group-item-info:focus,a.list-group-item-info:hover{color:#31708f;background-color:#c4e3f3}a.list-group-item-info.active,a.list-group-item-info.active:focus,a.list-group-item-info.active:hover{color:#fff;background-color:#31708f;border-color:#31708f}.list-group-item-warning{color:#8a6d3b;background-color:#fcf8e3}a.list-group-item-warning{color:#8a6d3b}a.list-group-item-warning .list-group-item-heading{color:inherit}a.list-group-item-warning:focus,a.list-group-item-warning:hover{color:#8a6d3b;background-color:#faf2cc}a.list-group-item-warning.active,a.list-group-item-warning.active:focus,a.list-group-item-warning.active:hover{color:#fff;background-color:#8a6d3b;border-color:#8a6d3b}.list-group-item-danger{color:#a94442;background-color:#f2dede}a.list-group-item-danger{color:#a94442}a.list-group-item-danger .list-group-item-heading{color:inherit}a.list-group-item-danger:focus,a.list-group-item-danger:hover{color:#a94442;background-color:#ebcccc}a.list-group-item-danger.active,a.list-group-item-danger.active:focus,a.list-group-item-danger.active:hover{color:#fff;background-color:#a94442;border-color:#a94442}.list-group-item-heading{margin-top:0;margin-bottom:5px}.list-group-item-text{margin-bottom:0;line-height:1.3}.panel{margin-bottom:20px;background-color:#fff;border:1px solid transparent;border-radius:4px;-webkit-box-shadow:0 1px 1px rgba(0,0,0,.05);box-shadow:0 1px 1px rgba(0,0,0,.05)}.panel-body{padding:15px}.panel-heading{padding:10px 15px;border-bottom:1px solid transparent;border-top-left-radius:3px;border-top-right-radius:3px}.panel-heading>.dropdown .dropdown-toggle{color:inherit}.panel-title{margin-top:0;margin-bottom:0;font-size:16px;color:inherit}.panel-title>.small,.panel-title>.small>a,.panel-title>a,.panel-title>small,.panel-title>small>a{color:inherit}.panel-footer{padding:10px 15px;background-color:#f5f5f5;border-top:1px solid #ddd;border-bottom-right-radius:3px;border-bottom-left-radius:3px}.panel>.list-group,.panel>.panel-collapse>.list-group{margin-bottom:0}.panel>.list-group .list-group-item,.panel>.panel-collapse>.list-group .list-group-item{border-width:1px 0;border-radius:0}.panel>.list-group:first-child .list-group-item:first-child,.panel>.panel-collapse>.list-group:first-child .list-group-item:first-child{border-top:0;border-top-left-radius:3px;border-top-right-radius:3px}.panel>.list-group:last-child .list-group-item:last-child,.panel>.panel-collapse>.list-group:last-child .list-group-item:last-child{border-bottom:0;border-bottom-right-radius:3px;border-bottom-left-radius:3px}.panel-heading+.list-group .list-group-item:first-child{border-top-width:0}.list-group+.panel-footer{border-top-width:0}.panel>.panel-collapse>.table,.panel>.table,.panel>.table-responsive>.table{margin-bottom:0}.panel>.panel-collapse>.table caption,.panel>.table caption,.panel>.table-responsive>.table caption{padding-right:15px;padding-left:15px}.panel>.table-responsive:first-child>.table:first-child,.panel>.table:first-child{border-top-left-radius:3px;border-top-right-radius:3px}.panel>.table-responsive:first-child>.table:first-child>tbody:first-child>tr:first-child,.panel>.table-responsive:first-child>.table:first-child>thead:first-child>tr:first-child,.panel>.table:first-child>tbody:first-child>tr:first-child,.panel>.table:first-child>thead:first-child>tr:first-child{border-top-left-radius:3px;border-top-right-radius:3px}.panel>.table-responsive:first-child>.table:first-child>tbody:first-child>tr:first-child td:first-child,.panel>.table-responsive:first-child>.table:first-child>tbody:first-child>tr:first-child th:first-child,.panel>.table-responsive:first-child>.table:first-child>thead:first-child>tr:first-child td:first-child,.panel>.table-responsive:first-child>.table:first-child>thead:first-child>tr:first-child th:first-child,.panel>.table:first-child>tbody:first-child>tr:first-child td:first-child,.panel>.table:first-child>tbody:first-child>tr:first-child th:first-child,.panel>.table:first-child>thead:first-child>tr:first-child td:first-child,.panel>.table:first-child>thead:first-child>tr:first-child th:first-child{border-top-left-radius:3px}.panel>.table-responsive:first-child>.table:first-child>tbody:first-child>tr:first-child td:last-child,.panel>.table-responsive:first-child>.table:first-child>tbody:first-child>tr:first-child th:last-child,.panel>.table-responsive:first-child>.table:first-child>thead:first-child>tr:first-child td:last-child,.panel>.table-responsive:first-child>.table:first-child>thead:first-child>tr:first-child th:last-child,.panel>.table:first-child>tbody:first-child>tr:first-child td:last-child,.panel>.table:first-child>tbody:first-child>tr:first-child th:last-child,.panel>.table:first-child>thead:first-child>tr:first-child td:last-child,.panel>.table:first-child>thead:first-child>tr:first-child th:last-child{border-top-right-radius:3px}.panel>.table-responsive:last-child>.table:last-child,.panel>.table:last-child{border-bottom-right-radius:3px;border-bottom-left-radius:3px}.panel>.table-responsive:last-child>.table:last-child>tbody:last-child>tr:last-child,.panel>.table-responsive:last-child>.table:last-child>tfoot:last-child>tr:last-child,.panel>.table:last-child>tbody:last-child>tr:last-child,.panel>.table:last-child>tfoot:last-child>tr:last-child{border-bottom-right-radius:3px;border-bottom-left-radius:3px}.panel>.table-responsive:last-child>.table:last-child>tbody:last-child>tr:last-child td:first-child,.panel>.table-responsive:last-child>.table:last-child>tbody:last-child>tr:last-child th:first-child,.panel>.table-responsive:last-child>.table:last-child>tfoot:last-child>tr:last-child td:first-child,.panel>.table-responsive:last-child>.table:last-child>tfoot:last-child>tr:last-child th:first-child,.panel>.table:last-child>tbody:last-child>tr:last-child td:first-child,.panel>.table:last-child>tbody:last-child>tr:last-child th:first-child,.panel>.table:last-child>tfoot:last-child>tr:last-child td:first-child,.panel>.table:last-child>tfoot:last-child>tr:last-child th:first-child{border-bottom-left-radius:3px}.panel>.table-responsive:last-child>.table:last-child>tbody:last-child>tr:last-child td:last-child,.panel>.table-responsive:last-child>.table:last-child>tbody:last-child>tr:last-child th:last-child,.panel>.table-responsive:last-child>.table:last-child>tfoot:last-child>tr:last-child td:last-child,.panel>.table-responsive:last-child>.table:last-child>tfoot:last-child>tr:last-child th:last-child,.panel>.table:last-child>tbody:last-child>tr:last-child td:last-child,.panel>.table:last-child>tbody:last-child>tr:last-child th:last-child,.panel>.table:last-child>tfoot:last-child>tr:last-child td:last-child,.panel>.table:last-child>tfoot:last-child>tr:last-child th:last-child{border-bottom-right-radius:3px}.panel>.panel-body+.table,.panel>.panel-body+.table-responsive,.panel>.table+.panel-body,.panel>.table-responsive+.panel-body{border-top:1px solid #ddd}.panel>.table>tbody:first-child>tr:first-child td,.panel>.table>tbody:first-child>tr:first-child th{border-top:0}.panel>.table-bordered,.panel>.table-responsive>.table-bordered{border:0}.panel>.table-bordered>tbody>tr>td:first-child,.panel>.table-bordered>tbody>tr>th:first-child,.panel>.table-bordered>tfoot>tr>td:first-child,.panel>.table-bordered>tfoot>tr>th:first-child,.panel>.table-bordered>thead>tr>td:first-child,.panel>.table-bordered>thead>tr>th:first-child,.panel>.table-responsive>.table-bordered>tbody>tr>td:first-child,.panel>.table-responsive>.table-bordered>tbody>tr>th:first-child,.panel>.table-responsive>.table-bordered>tfoot>tr>td:first-child,.panel>.table-responsive>.table-bordered>tfoot>tr>th:first-child,.panel>.table-responsive>.table-bordered>thead>tr>td:first-child,.panel>.table-responsive>.table-bordered>thead>tr>th:first-child{border-left:0}.panel>.table-bordered>tbody>tr>td:last-child,.panel>.table-bordered>tbody>tr>th:last-child,.panel>.table-bordered>tfoot>tr>td:last-child,.panel>.table-bordered>tfoot>tr>th:last-child,.panel>.table-bordered>thead>tr>td:last-child,.panel>.table-bordered>thead>tr>th:last-child,.panel>.table-responsive>.table-bordered>tbody>tr>td:last-child,.panel>.table-responsive>.table-bordered>tbody>tr>th:last-child,.panel>.table-responsive>.table-bordered>tfoot>tr>td:last-child,.panel>.table-responsive>.table-bordered>tfoot>tr>th:last-child,.panel>.table-responsive>.table-bordered>thead>tr>td:last-child,.panel>.table-responsive>.table-bordered>thead>tr>th:last-child{border-right:0}.panel>.table-bordered>tbody>tr:first-child>td,.panel>.table-bordered>tbody>tr:first-child>th,.panel>.table-bordered>thead>tr:first-child>td,.panel>.table-bordered>thead>tr:first-child>th,.panel>.table-responsive>.table-bordered>tbody>tr:first-child>td,.panel>.table-responsive>.table-bordered>tbody>tr:first-child>th,.panel>.table-responsive>.table-bordered>thead>tr:first-child>td,.panel>.table-responsive>.table-bordered>thead>tr:first-child>th{border-bottom:0}.panel>.table-bordered>tbody>tr:last-child>td,.panel>.table-bordered>tbody>tr:last-child>th,.panel>.table-bordered>tfoot>tr:last-child>td,.panel>.table-bordered>tfoot>tr:last-child>th,.panel>.table-responsive>.table-bordered>tbody>tr:last-child>td,.panel>.table-responsive>.table-bordered>tbody>tr:last-child>th,.panel>.table-responsive>.table-bordered>tfoot>tr:last-child>td,.panel>.table-responsive>.table-bordered>tfoot>tr:last-child>th{border-bottom:0}.panel>.table-responsive{margin-bottom:0;border:0}.panel-group{margin-bottom:20px}.panel-group .panel{margin-bottom:0;border-radius:4px}.panel-group .panel+.panel{margin-top:5px}.panel-group .panel-heading{border-bottom:0}.panel-group .panel-heading+.panel-collapse>.list-group,.panel-group .panel-heading+.panel-collapse>.panel-body{border-top:1px solid #ddd}.panel-group .panel-footer{border-top:0}.panel-group .panel-footer+.panel-collapse .panel-body{border-bottom:1px solid #ddd}.panel-default{border-color:#ddd}.panel-default>.panel-heading{color:#333;background-color:#f5f5f5;border-color:#ddd}.panel-default>.panel-heading+.panel-collapse>.panel-body{border-top-color:#ddd}.panel-default>.panel-heading .badge{color:#f5f5f5;background-color:#333}.panel-default>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#ddd}.panel-primary{border-color:#337ab7}.panel-primary>.panel-heading{color:#fff;background-color:#337ab7;border-color:#337ab7}.panel-primary>.panel-heading+.panel-collapse>.panel-body{border-top-color:#337ab7}.panel-primary>.panel-heading .badge{color:#337ab7;background-color:#fff}.panel-primary>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#337ab7}.panel-success{border-color:#d6e9c6}.panel-success>.panel-heading{color:#3c763d;background-color:#dff0d8;border-color:#d6e9c6}.panel-success>.panel-heading+.panel-collapse>.panel-body{border-top-color:#d6e9c6}.panel-success>.panel-heading .badge{color:#dff0d8;background-color:#3c763d}.panel-success>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#d6e9c6}.panel-info{border-color:#bce8f1}.panel-info>.panel-heading{color:#31708f;background-color:#d9edf7;border-color:#bce8f1}.panel-info>.panel-heading+.panel-collapse>.panel-body{border-top-color:#bce8f1}.panel-info>.panel-heading .badge{color:#d9edf7;background-color:#31708f}.panel-info>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#bce8f1}.panel-warning{border-color:#faebcc}.panel-warning>.panel-heading{color:#8a6d3b;background-color:#fcf8e3;border-color:#faebcc}.panel-warning>.panel-heading+.panel-collapse>.panel-body{border-top-color:#faebcc}.panel-warning>.panel-heading .badge{color:#fcf8e3;background-color:#8a6d3b}.panel-warning>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#faebcc}.panel-danger{border-color:#ebccd1}.panel-danger>.panel-heading{color:#a94442;background-color:#f2dede;border-color:#ebccd1}.panel-danger>.panel-heading+.panel-collapse>.panel-body{border-top-color:#ebccd1}.panel-danger>.panel-heading .badge{color:#f2dede;background-color:#a94442}.panel-danger>.panel-footer+.panel-collapse>.panel-body{border-bottom-color:#ebccd1}.embed-responsive{position:relative;display:block;height:0;padding:0;overflow:hidden}.embed-responsive .embed-responsive-item,.embed-responsive embed,.embed-responsive iframe,.embed-responsive object,.embed-responsive video{position:absolute;top:0;bottom:0;left:0;width:100%;height:100%;border:0}.embed-responsive.embed-responsive-16by9{padding-bottom:56.25%}.embed-responsive.embed-responsive-4by3{padding-bottom:75%}.well{min-height:20px;padding:19px;margin-bottom:20px;background-color:#f5f5f5;border:1px solid #e3e3e3;border-radius:4px;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.05);box-shadow:inset 0 1px 1px rgba(0,0,0,.05)}.well blockquote{border-color:#ddd;border-color:rgba(0,0,0,.15)}.well-lg{padding:24px;border-radius:6px}.well-sm{padding:9px;border-radius:3px}.close{float:right;font-size:21px;font-weight:700;line-height:1;color:#000;text-shadow:0 1px 0 #fff;filter:alpha(opacity=20);opacity:.2}.close:focus,.close:hover{color:#000;text-decoration:none;cursor:pointer;filter:alpha(opacity=50);opacity:.5}button.close{-webkit-appearance:none;padding:0;cursor:pointer;background:0 0;border:0}.modal-open{overflow:hidden}.modal{position:fixed;top:0;right:0;bottom:0;left:0;z-index:1040;display:none;overflow:hidden;-webkit-overflow-scrolling:touch;outline:0}.modal.fade .modal-dialog{-webkit-transition:-webkit-transform .3s ease-out;-o-transition:-o-transform .3s ease-out;transition:transform .3s ease-out;-webkit-transform:translate(0,-25%);-ms-transform:translate(0,-25%);-o-transform:translate(0,-25%);transform:translate(0,-25%)}.modal.in .modal-dialog{-webkit-transform:translate(0,0);-ms-transform:translate(0,0);-o-transform:translate(0,0);transform:translate(0,0)}.modal-open .modal{overflow-x:hidden;overflow-y:auto}.modal-dialog{position:relative;width:auto;margin:10px}.modal-content{position:relative;background-color:#fff;-webkit-background-clip:padding-box;background-clip:padding-box;border:1px solid #999;border:1px solid rgba(0,0,0,.2);border-radius:6px;outline:0;-webkit-box-shadow:0 3px 9px rgba(0,0,0,.5);box-shadow:0 3px 9px rgba(0,0,0,.5)}.modal-backdrop{position:absolute;top:0;right:0;left:0;background-color:#000}.modal-backdrop.fade{filter:alpha(opacity=0);opacity:0}.modal-backdrop.in{filter:alpha(opacity=50);opacity:.5}.modal-header{min-height:16.43px;padding:15px;border-bottom:1px solid #e5e5e5}.modal-header .close{margin-top:-2px}.modal-title{margin:0;line-height:1.42857143}.modal-body{position:relative;padding:15px}.modal-footer{padding:15px;text-align:right;border-top:1px solid #e5e5e5}.modal-footer .btn+.btn{margin-bottom:0;margin-left:5px}.modal-footer .btn-group .btn+.btn{margin-left:-1px}.modal-footer .btn-block+.btn-block{margin-left:0}.modal-scrollbar-measure{position:absolute;top:-9999px;width:50px;height:50px;overflow:scroll}@media (min-width:768px){.modal-dialog{width:600px;margin:30px auto}.modal-content{-webkit-box-shadow:0 5px 15px rgba(0,0,0,.5);box-shadow:0 5px 15px rgba(0,0,0,.5)}.modal-sm{width:300px}}@media (min-width:992px){.modal-lg{width:900px}}.tooltip{position:absolute;z-index:1070;display:block;font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:12px;font-weight:400;line-height:1.4;visibility:visible;filter:alpha(opacity=0);opacity:0}.tooltip.in{filter:alpha(opacity=90);opacity:.9}.tooltip.top{padding:5px 0;margin-top:-3px}.tooltip.right{padding:0 5px;margin-left:3px}.tooltip.bottom{padding:5px 0;margin-top:3px}.tooltip.left{padding:0 5px;margin-left:-3px}.tooltip-inner{max-width:200px;padding:3px 8px;color:#fff;text-align:center;text-decoration:none;background-color:#000;border-radius:4px}.tooltip-arrow{position:absolute;width:0;height:0;border-color:transparent;border-style:solid}.tooltip.top .tooltip-arrow{bottom:0;left:50%;margin-left:-5px;border-width:5px 5px 0;border-top-color:#000}.tooltip.top-left .tooltip-arrow{right:5px;bottom:0;margin-bottom:-5px;border-width:5px 5px 0;border-top-color:#000}.tooltip.top-right .tooltip-arrow{bottom:0;left:5px;margin-bottom:-5px;border-width:5px 5px 0;border-top-color:#000}.tooltip.right .tooltip-arrow{top:50%;left:0;margin-top:-5px;border-width:5px 5px 5px 0;border-right-color:#000}.tooltip.left .tooltip-arrow{top:50%;right:0;margin-top:-5px;border-width:5px 0 5px 5px;border-left-color:#000}.tooltip.bottom .tooltip-arrow{top:0;left:50%;margin-left:-5px;border-width:0 5px 5px;border-bottom-color:#000}.tooltip.bottom-left .tooltip-arrow{top:0;right:5px;margin-top:-5px;border-width:0 5px 5px;border-bottom-color:#000}.tooltip.bottom-right .tooltip-arrow{top:0;left:5px;margin-top:-5px;border-width:0 5px 5px;border-bottom-color:#000}.popover{position:absolute;top:0;left:0;z-index:1060;display:none;max-width:276px;padding:1px;font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:14px;font-weight:400;line-height:1.42857143;text-align:left;white-space:normal;background-color:#fff;-webkit-background-clip:padding-box;background-clip:padding-box;border:1px solid #ccc;border:1px solid rgba(0,0,0,.2);border-radius:6px;-webkit-box-shadow:0 5px 10px rgba(0,0,0,.2);box-shadow:0 5px 10px rgba(0,0,0,.2)}.popover.top{margin-top:-10px}.popover.right{margin-left:10px}.popover.bottom{margin-top:10px}.popover.left{margin-left:-10px}.popover-title{padding:8px 14px;margin:0;font-size:14px;background-color:#f7f7f7;border-bottom:1px solid #ebebeb;border-radius:5px 5px 0 0}.popover-content{padding:9px 14px}.popover>.arrow,.popover>.arrow:after{position:absolute;display:block;width:0;height:0;border-color:transparent;border-style:solid}.popover>.arrow{border-width:11px}.popover>.arrow:after{content:"";border-width:10px}.popover.top>.arrow{bottom:-11px;left:50%;margin-left:-11px;border-top-color:#999;border-top-color:rgba(0,0,0,.25);border-bottom-width:0}.popover.top>.arrow:after{bottom:1px;margin-left:-10px;content:" ";border-top-color:#fff;border-bottom-width:0}.popover.right>.arrow{top:50%;left:-11px;margin-top:-11px;border-right-color:#999;border-right-color:rgba(0,0,0,.25);border-left-width:0}.popover.right>.arrow:after{bottom:-10px;left:1px;content:" ";border-right-color:#fff;border-left-width:0}.popover.bottom>.arrow{top:-11px;left:50%;margin-left:-11px;border-top-width:0;border-bottom-color:#999;border-bottom-color:rgba(0,0,0,.25)}.popover.bottom>.arrow:after{top:1px;margin-left:-10px;content:" ";border-top-width:0;border-bottom-color:#fff}.popover.left>.arrow{top:50%;right:-11px;margin-top:-11px;border-right-width:0;border-left-color:#999;border-left-color:rgba(0,0,0,.25)}.popover.left>.arrow:after{right:1px;bottom:-10px;content:" ";border-right-width:0;border-left-color:#fff}.carousel{position:relative}.carousel-inner{position:relative;width:100%;overflow:hidden}.carousel-inner>.item{position:relative;display:none;-webkit-transition:.6s ease-in-out left;-o-transition:.6s ease-in-out left;transition:.6s ease-in-out left}.carousel-inner>.item>a>img,.carousel-inner>.item>img{line-height:1}@media all and (transform-3d),(-webkit-transform-3d){.carousel-inner>.item{-webkit-transition:-webkit-transform .6s ease-in-out;-o-transition:-o-transform .6s ease-in-out;transition:transform .6s ease-in-out;-webkit-backface-visibility:hidden;backface-visibility:hidden;-webkit-perspective:1000;perspective:1000}.carousel-inner>.item.active.right,.carousel-inner>.item.next{left:0;-webkit-transform:translate3d(100%,0,0);transform:translate3d(100%,0,0)}.carousel-inner>.item.active.left,.carousel-inner>.item.prev{left:0;-webkit-transform:translate3d(-100%,0,0);transform:translate3d(-100%,0,0)}.carousel-inner>.item.active,.carousel-inner>.item.next.left,.carousel-inner>.item.prev.right{left:0;-webkit-transform:translate3d(0,0,0);transform:translate3d(0,0,0)}}.carousel-inner>.active,.carousel-inner>.next,.carousel-inner>.prev{display:block}.carousel-inner>.active{left:0}.carousel-inner>.next,.carousel-inner>.prev{position:absolute;top:0;width:100%}.carousel-inner>.next{left:100%}.carousel-inner>.prev{left:-100%}.carousel-inner>.next.left,.carousel-inner>.prev.right{left:0}.carousel-inner>.active.left{left:-100%}.carousel-inner>.active.right{left:100%}.carousel-control{position:absolute;top:0;bottom:0;left:0;width:15%;font-size:20px;color:#fff;text-align:center;text-shadow:0 1px 2px rgba(0,0,0,.6);filter:alpha(opacity=50);opacity:.5}.carousel-control.left{background-image:-webkit-linear-gradient(left,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);background-image:-o-linear-gradient(left,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);background-image:-webkit-gradient(linear,left top,right top,from(rgba(0,0,0,.5)),to(rgba(0,0,0,.0001)));background-image:linear-gradient(to right,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#80000000', endColorstr='#00000000', GradientType=1);background-repeat:repeat-x}.carousel-control.right{right:0;left:auto;background-image:-webkit-linear-gradient(left,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);background-image:-o-linear-gradient(left,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);background-image:-webkit-gradient(linear,left top,right top,from(rgba(0,0,0,.0001)),to(rgba(0,0,0,.5)));background-image:linear-gradient(to right,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#00000000', endColorstr='#80000000', GradientType=1);background-repeat:repeat-x}.carousel-control:focus,.carousel-control:hover{color:#fff;text-decoration:none;filter:alpha(opacity=90);outline:0;opacity:.9}.carousel-control .glyphicon-chevron-left,.carousel-control .glyphicon-chevron-right,.carousel-control .icon-next,.carousel-control .icon-prev{position:absolute;top:50%;z-index:5;display:inline-block}.carousel-control .glyphicon-chevron-left,.carousel-control .icon-prev{left:50%;margin-left:-10px}.carousel-control .glyphicon-chevron-right,.carousel-control .icon-next{right:50%;margin-right:-10px}.carousel-control .icon-next,.carousel-control .icon-prev{width:20px;height:20px;margin-top:-10px;font-family:serif;line-height:1}.carousel-control .icon-prev:before{content:'\2039'}.carousel-control .icon-next:before{content:'\203a'}.carousel-indicators{position:absolute;bottom:10px;left:50%;z-index:15;width:60%;padding-left:0;margin-left:-30%;text-align:center;list-style:none}.carousel-indicators li{display:inline-block;width:10px;height:10px;margin:1px;text-indent:-999px;cursor:pointer;background-color:#000 \9;background-color:rgba(0,0,0,0);border:1px solid #fff;border-radius:10px}.carousel-indicators .active{width:12px;height:12px;margin:0;background-color:#fff}.carousel-caption{position:absolute;right:15%;bottom:20px;left:15%;z-index:10;padding-top:20px;padding-bottom:20px;color:#fff;text-align:center;text-shadow:0 1px 2px rgba(0,0,0,.6)}.carousel-caption .btn{text-shadow:none}@media screen and (min-width:768px){.carousel-control .glyphicon-chevron-left,.carousel-control .glyphicon-chevron-right,.carousel-control .icon-next,.carousel-control .icon-prev{width:30px;height:30px;margin-top:-15px;font-size:30px}.carousel-control .glyphicon-chevron-left,.carousel-control .icon-prev{margin-left:-15px}.carousel-control .glyphicon-chevron-right,.carousel-control .icon-next{margin-right:-15px}.carousel-caption{right:20%;left:20%;padding-bottom:30px}.carousel-indicators{bottom:20px}}.btn-group-vertical>.btn-group:after,.btn-group-vertical>.btn-group:before,.btn-toolbar:after,.btn-toolbar:before,.clearfix:after,.clearfix:before,.container-fluid:after,.container-fluid:before,.container:after,.container:before,.dl-horizontal dd:after,.dl-horizontal dd:before,.form-horizontal .form-group:after,.form-horizontal .form-group:before,.modal-footer:after,.modal-footer:before,.nav:after,.nav:before,.navbar-collapse:after,.navbar-collapse:before,.navbar-header:after,.navbar-header:before,.navbar:after,.navbar:before,.pager:after,.pager:before,.panel-body:after,.panel-body:before,.row:after,.row:before{display:table;content:" "}.btn-group-vertical>.btn-group:after,.btn-toolbar:after,.clearfix:after,.container-fluid:after,.container:after,.dl-horizontal dd:after,.form-horizontal .form-group:after,.modal-footer:after,.nav:after,.navbar-collapse:after,.navbar-header:after,.navbar:after,.pager:after,.panel-body:after,.row:after{clear:both}.center-block{display:block;margin-right:auto;margin-left:auto}.pull-right{float:right!important}.pull-left{float:left!important}.hide{display:none!important}.show{display:block!important}.invisible{visibility:hidden}.text-hide{font:0/0 a;color:transparent;text-shadow:none;background-color:transparent;border:0}.hidden{display:none!important;visibility:hidden!important}.affix{position:fixed}@-ms-viewport{width:device-width}.visible-lg,.visible-md,.visible-sm,.visible-xs{display:none!important}.visible-lg-block,.visible-lg-inline,.visible-lg-inline-block,.visible-md-block,.visible-md-inline,.visible-md-inline-block,.visible-sm-block,.visible-sm-inline,.visible-sm-inline-block,.visible-xs-block,.visible-xs-inline,.visible-xs-inline-block{display:none!important}@media (max-width:767px){.visible-xs{display:block!important}table.visible-xs{display:table}tr.visible-xs{display:table-row!important}td.visible-xs,th.visible-xs{display:table-cell!important}}@media (max-width:767px){.visible-xs-block{display:block!important}}@media (max-width:767px){.visible-xs-inline{display:inline!important}}@media (max-width:767px){.visible-xs-inline-block{display:inline-block!important}}@media (min-width:768px) and (max-width:991px){.visible-sm{display:block!important}table.visible-sm{display:table}tr.visible-sm{display:table-row!important}td.visible-sm,th.visible-sm{display:table-cell!important}}@media (min-width:768px) and (max-width:991px){.visible-sm-block{display:block!important}}@media (min-width:768px) and (max-width:991px){.visible-sm-inline{display:inline!important}}@media (min-width:768px) and (max-width:991px){.visible-sm-inline-block{display:inline-block!important}}@media (min-width:992px) and (max-width:1199px){.visible-md{display:block!important}table.visible-md{display:table}tr.visible-md{display:table-row!important}td.visible-md,th.visible-md{display:table-cell!important}}@media (min-width:992px) and (max-width:1199px){.visible-md-block{display:block!important}}@media (min-width:992px) and (max-width:1199px){.visible-md-inline{display:inline!important}}@media (min-width:992px) and (max-width:1199px){.visible-md-inline-block{display:inline-block!important}}@media (min-width:1200px){.visible-lg{display:block!important}table.visible-lg{display:table}tr.visible-lg{display:table-row!important}td.visible-lg,th.visible-lg{display:table-cell!important}}@media (min-width:1200px){.visible-lg-block{display:block!important}}@media (min-width:1200px){.visible-lg-inline{display:inline!important}}@media (min-width:1200px){.visible-lg-inline-block{display:inline-block!important}}@media (max-width:767px){.hidden-xs{display:none!important}}@media (min-width:768px) and (max-width:991px){.hidden-sm{display:none!important}}@media (min-width:992px) and (max-width:1199px){.hidden-md{display:none!important}}@media (min-width:1200px){.hidden-lg{display:none!important}}.visible-print{display:none!important}@media print{.visible-print{display:block!important}table.visible-print{display:table}tr.visible-print{display:table-row!important}td.visible-print,th.visible-print{display:table-cell!important}}.visible-print-block{display:none!important}@media print{.visible-print-block{display:block!important}}.visible-print-inline{display:none!important}@media print{.visible-print-inline{display:inline!important}}.visible-print-inline-block{display:none!important}@media print{.visible-print-inline-block{display:inline-block!important}}@media print{.hidden-print{display:none!important}}

/\*

\* Globals

\*/

/\* Links \*/

a,

a:focus,

a:hover {

color: #fff;

}

/\* Custom default button \*/

.btn-default,

.btn-default:hover,

.btn-default:focus {

color: #333;

text-shadow: none; /\* Prevent inheritence from `body` \*/

background-color: #fff;

border: 1px solid #fff;

}

/\*

\* Base structure

\*/

html,

body {

height: 100%;

background-color: #333;

background-image: url("http://i61.tinypic.com/1z6v2hz.jpg");

}

body {

color: #fff;

text-align: center;

text-shadow: 0 1px 3px rgba(0,0,0,.5);

}

/\* Extra markup and styles for table-esque vertical and horizontal centering \*/

.site-wrapper {

display: table;

width: 100%;

height: 100%; /\* For at least Firefox \*/

min-height: 100%;

-webkit-box-shadow: inset 0 0 100px rgba(0,0,0,.5);

box-shadow: inset 0 0 100px rgba(0,0,0,.5);

}

.site-wrapper-inner {

display: table-cell;

vertical-align: top;

}

.cover-container {

margin-right: auto;

margin-left: auto;

}

/\* Padding for spacing \*/

.inner {

padding: 30px;

}

/\*

\* Header

\*/

.masthead-brand {

margin-top: 10px;

margin-bottom: 10px;

}

.masthead-nav > li {

display: inline-block;

}

.masthead-nav > li + li {

margin-left: 20px;

}

.masthead-nav > li > a {

padding-right: 0;

padding-left: 0;

font-size: 16px;

font-weight: bold;

color: #fff; /\* IE8 proofing \*/

color: rgba(255,255,255,.75);

border-bottom: 2px solid transparent;

}

.masthead-nav > li > a:hover,

.masthead-nav > li > a:focus {

background-color: transparent;

border-bottom-color: #a9a9a9;

border-bottom-color: rgba(255,255,255,.25);

}

.masthead-nav > .active > a,

.masthead-nav > .active > a:hover,

.masthead-nav > .active > a:focus {

color: #fff;

border-bottom-color: #fff;

}

@media (min-width: 768px) {

.masthead-brand {

float: left;

}

.masthead-nav {

float: right;

}

}

/\*

\* Cover

\*/

.cover {

padding: 0 20px;

}

.cover .btn-lg {

padding: 10px 20px;

font-weight: bold;

}

/\*

\* Footer

\*/

.mastfoot {

color: #999; /\* IE8 proofing \*/

color: rgba(255,255,255,.5);

}

/\*

\* Affix and center

\*/

@media (min-width: 768px) {

/\* Pull out the header and footer \*/

.masthead {

position: fixed;

top: 0;

}

.mastfoot {

position: fixed;

bottom: 0;

}

/\* Start the vertical centering \*/

.site-wrapper-inner {

vertical-align: middle;

}

/\* Handle the widths \*/

.masthead,

.mastfoot,

.cover-container {

width: 100%; /\* Must be percentage or pixels for horizontal alignment \*/

}

}

@media (min-width: 992px) {

.masthead,

.mastfoot,

.cover-container {

width: 700px;

}

}

.input{

color: black;

}

/\*

\* Base structure

\*/

/\* Move down content because we have a fixed navbar that is 50px tall \*/

body {

padding-top: 50px;

background-image: url("http://www.eternaladoptions.org/sites/default/files/Spots-Background-Light-Bright-Pale.jpg");

}

/\* This is the css for the scroolbar table \*/

.scrollit {

overflow:scroll;

height:400px;

width: 950px;

}

/\*

\* Global add-ons

\*/

.sub-header {

padding-bottom: 10px;

border-bottom: 1px solid #eee;

}

/\*

\* Top navigation

\* Hide default border to remove 1px line.

\*/

.navbar-fixed-top {

border: 0;

}

/\*

\* Sidebar

\*/

/\* Hide for mobile, show later \*/

.sidebar {

display: none;

}

@media (min-width: 768px) {

.sidebar {

position: fixed;

bottom: 0;

left: 0;

z-index: 1000;

display: block;

/\* padding: 20px;\*/

overflow-x: hidden;

overflow-y: auto; /\* Scrollable contents if viewport is shorter than content. \*/

background-color: #f5f5f5;

border-right: 1px solid #eee;

}

}

/\* Sidebar navigation \*/

.nav-sidebar {

margin-right: -21px; /\* 20px padding + 1px border \*/

margin-bottom: 20px;

margin-left: -20px;

}

.nav-sidebar > li > a {

padding-right: 20px;

padding-left: 20px;

}

.nav-sidebar > .active > a,

.nav-sidebar > .active > a:hover,

.nav-sidebar > .active > a:focus {

color: #fff;

background-color: #428bca;

}

/\*

\* Main content

\*/

.main {

padding: 20px;

}

@media (min-width: 768px) {

.main {

padding-right: 40px;

padding-left: 40px;

}

}

.main .page-header {

margin-top: 0;

}

/\*

\* Placeholder dashboard ideas

\*/

.placeholders {

margin-bottom: 30px;

text-align: center;

}

.placeholders h4 {

margin-bottom: 0;

}

.placeholder {

margin-bottom: 20px;

}

.placeholder img {

display: inline-block;

border-radius: 50%;

}