**ngày 11/1:**

* cách chèn 1 ô vào trong bảng đã tồn tại

var tableRef = document.getElementById('myTable').getElementsByTagName('tbody')[0];

// Insert a row in the table at the last row

var newRow = tableRef.insertRow(tableRef.rows.length);

// Insert a cell in the row at index 0

var newCell = newRow.insertCell(0);

// Append a text node to the cell

var newText = document.createTextNode('New row');

newCell.appendChild(newText);

* Cách chèn unicode gõ tiếng Việt vào java

"jdbc:mysql://localhost:3306/quanlythuvien?useUnicode=yes&characterEncoding=UTF-8", "root", "1234");

Thêm 2 dòng này vào: (tùy theo chức năng của chương trình mà thêm vào cho hợp lý)

request.setCharacterEncoding("UTF-8");

response.setCharacterEncoding("UTF-8");

* Cách thay đổi định dạng cột trong mysql để gõ tiếng việt

ALTER TABLE <table\_name> MODIFY <column\_name> VARCHAR(255) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;

* Cách sửa lỗi : peration not allowed after ResultSet closed

A Statement object can have only one active ResultSet, so when you execute rs2 = stmt.executeQuery(sql), the first ResultSet (rs) gets closed.

Create two Statement objects, one for rs and another for rs2.

* Cách kết nối tới bảng excel để làm việc với excel trong eclipse

1. Download thư viện poi-3.17 và add vào eclipse

-Kết nối với cơ sở dữ liệu mySql bằng jsp

Thư viện : <%@ page import=*"java.sql.\*"*%>

Connection conn = **null**;

Statement stmt = **null**;

ResultSet rs = **null**;

**try** {

Class.forName("com.mysql.jdbc.Driver");

conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/thongtin\_phanhoi?useUnicode=yes&characterEncoding=UTF-8", "root", "1234");

stmt = conn.createStatement();

stmt.executeUpdate("INSERT INTO thongtin...");

//rs = stmt.executeQuery("SELECT \* FROM thongtin");

} **catch** (Exception e) {

e.printStackTrace();

} **finally** {

**if** (stmt != **null**)

stmt.close();

**if** (conn != **null**)

conn.close();

}

Nếu kết nối với cơ sở dữ liệu Sqlite thì thay thế

Class.forName("com.mysql.jdbc.Driver"); -> Class.*forName*("org.sqlite.JDBC");

conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/thongtin\_phanhoi?useUnicode=yes&characterEncoding=UTF-8", "root", "1234"); -> conn = DriverManager.*getConnection*("jdbc:sqlite:D:/DINHNV/My Data/SQLite DB/database/FullCalendar.db");

* Cách upload file lên thư mục trong eclipse

+ Download  thư viện commons-fileupload và apache common-io và add vào thư viện trong eclipse

+ thêm đường dẫn lưu file vào file web.xml

<context-param>

<description>Location to store uploaded file</description>

<param-name>file-upload</param-name>

<param-value>

C:\Users\DinhNV\workspace\UploadFile\WebContent\uploaded

</param-value>

</context-param>

+ Tạo file uploadFile.html

<html>

<head>

<title>File Uploading Form</title>

</head>

<body>

<h3>File Upload:</h3>

Select a file to upload: <br />

<form action = *"uploadfile.jsp"* method = *"post"*

enctype = *"multipart/form-data"*>

<input type = *"file"* name = *"file"* size = *"50"* /><br>

<br />

<input type = *"submit"* value = *"Upload Files"* />

</form>

</body>

</html>

+ tạo file uploadfile.jsp

<%@ page import = *"java.io.\*,java.util.\*, javax.servlet.\*"* %>

<%@ page import = *"javax.servlet.http.\*"* %>

<%@ page import = *"org.apache.commons.fileupload.\*"* %>

<%@ page import = *"org.apache.commons.fileupload.disk.\*"* %>

<%@ page import = *"org.apache.commons.fileupload.servlet.\*"* %>

<%@ page import = *"org.apache.commons.io.output.\*"* %>

<%

File file ;

**int** maxFileSize = 5000 \* 1024;

**int** maxMemSize = 5000 \* 1024;

ServletContext context = pageContext.getServletContext();

String filePath = context.getInitParameter("file-upload");

// Verify the content type

String contentType = request.getContentType();

**if** ((contentType.indexOf("multipart/form-data") >= 0)) {

DiskFileItemFactory factory = **new** DiskFileItemFactory();

// maximum size that will be stored in memory

factory.setSizeThreshold(maxMemSize);

// Location to save data that is larger than maxMemSize.

factory.setRepository(**new** File("c:\\temp"));

// Create a new file upload handler

ServletFileUpload upload = **new** ServletFileUpload(factory);

// maximum file size to be uploaded.

upload.setSizeMax( maxFileSize );

**try** {

// Parse the request to get file items.

List fileItems = upload.parseRequest(request);

// Process the uploaded file items

Iterator i = fileItems.iterator();

out.println("<html>");

out.println("<head>");

out.println("<title>JSP File upload</title>");

out.println("</head>");

out.println("<body>");

**while** ( i.hasNext () ) {

FileItem fi = (FileItem)i.next();

**if** ( !fi.isFormField () ) {

// Get the uploaded file parameters

String fieldName = fi.getFieldName();

String fileName = fi.getName();

**boolean** isInMemory = fi.isInMemory();

**long** sizeInBytes = fi.getSize();

// Write the file

**if**( fileName.lastIndexOf("\\") >= 0 ) {

file = **new** File( filePath + fileName.substring( fileName.lastIndexOf("\\"))) ;

} **else** {

file = **new** File( filePath + fileName.substring(fileName.lastIndexOf("\\")+1)) ;

}

fi.write( file ) ;

out.println("Uploaded Filename: " + filePath + fileName + "<br>");

}

}

out.println("</body>");

out.println("</html>");

} **catch**(Exception ex) {

System.out.println(ex);

}

} **else** {

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet upload</title>");

out.println("</head>");

out.println("<body>");

out.println("<p>No file uploaded</p>");

out.println("</body>");

out.println("</html>");

}

%>

* Cách chèn trình soạn thảo văn bản vào trang web

+ tải thư viện ckeditor và add vào eclipse

+ thêm đường dẫn này vào tag <head>

<script type=*"text/javascript"* src=*"ckeditor/ckeditor.js"*> </script>

+ thêm đoạn code:

<h1>Bình Luận</h1>

<h4 style="color:*red*">Vui lòng đăng nhập để bình luận</h4>

<form action=*"luuDB.jsp"*>

<textarea name=*"ten"* id=*"ten"*></textarea>

<script>CKEDITOR.replace('ten');</script>

<input type=*"submit"* value=*"Nhập"*>

</form>

* Cách Tạo đánh giá bằng Jquery Raty

+ Tải thư viện Jquery Raty và add vào eclipse

+ Khai báo các thư viện vào thẻ <head>

+ Đoạn code để đánh giá

<div id=*"1"*>Đánh giá</div>

<p style="display: *none*"></p>

+ Chèn đoạn script này vào sau thẻ <body>

<script>

$('#1').raty({

cancel : **true**,

cancelPlace : 'right',

number : 18,

click : **function**(score, evt) {

$('#rate').attr("value", score);

}

});

</script>

* Kiểm tra xem một table có tồn tại dữ liệu không?

public class kiemTraTonTai {

public static boolean tonTai(String id) throws SQLException

{

Connection conn = null;

Statement stmt = null;

ResultSet rs=null;

boolean a=false;

try {

Class.forName("com.mysql.jdbc.Driver");

conn = DriverManager.getConnection(

"jdbc:mysql://localhost:3306/ontap", "root", "1234");

stmt = conn.createStatement();

rs = stmt.executeQuery("SELECT \* FROM user\_danhgia where id\_hinhanh="+id+"");

if(rs.next()) a=true;

} catch (Exception se) {

se.printStackTrace();

} finally {

if (rs != null)

rs.close();

if (stmt != null)

stmt.close();

if (conn != null)

conn.close();

}

return a;

}

}

* Kết nối để lấy giá trị từ form sau đó insert vào db

<%@ page import="java.sql.\*"%>

<%@ page import=*"java.io.\*,java.util.\*, javax.servlet.\*"*%>

<%@ page import=*"javax.servlet.http.\*"*%>

<%@ page import=*"org.apache.commons.fileupload.\*"*%>

<%@ page import=*"org.apache.commons.fileupload.disk.\*"*%>

<%@ page import=*"org.apache.commons.fileupload.servlet.\*"*%>

<%@ page import=*"org.apache.commons.io.output.\*"*%>

<%

**int** maxFileSize = 5000 \* 1024;

**int** maxMemSize = 5000 \* 1024;

String a = "", b = "";

String contentType = request.getContentType();

**if** ((contentType.indexOf("multipart/form-data") >= 0)) {

DiskFileItemFactory factory = **new** DiskFileItemFactory();

factory.setSizeThreshold(maxMemSize);

factory.setRepository(**new** File("c:\\temp"));

ServletFileUpload upload = **new** ServletFileUpload(factory);

upload.setSizeMax(maxFileSize);

**try** {

List fileItems = upload.parseRequest(request);

Iterator i = fileItems.iterator();

**while** (i.hasNext()) {

FileItem fi = (FileItem) i.next();

**if** (fi.isFormField()) { // khi no la form

**if** ("1".equals(fi.getFieldName())) {

a = fi.getString();

}

**if** ("2".equals(fi.getFieldName())) {

b = fi.getString();

}

}

}

Statement stmt = **null**;

**try** {

Class.forName("com.mysql.jdbc.Driver");

conn = DriverManager

.getConnection(

"jdbc:mysql://localhost:3306/ontap?useUnicode=yes&characterEncoding=UTF-8",

"root", "1234");

stmt = conn.createStatement();

stmt.executeUpdate("INSERT INTO user\_danhgia(danh\_gia, nhan\_xet) value(\""+ a + "\", \""+ b + "\")");

} **catch** (Exception e) {

e.printStackTrace();

} **finally** {

**if** (stmt != **null**)

stmt.close();

**if** (conn != **null**)

conn.close();

}

out.println("Đã lưu vào DB!<br>");

} **catch** (Exception ex) {

System.out.println(ex);

}

}

%>

* Tìm hiểu play framework
* Cài đặt Play Framework:

1. Tải play Framework ở địa chỉ https://www.playframework.com/download#starters (chú ý version)
2. Giải nén vào thư mục trong máy tính
3. Cài biến môi trường cho play framework
4. Tạo project mới:
   1. Mở cmd
   2. Gõ cd [đường dẫn đến nơi lưu project]
   3. Gõ play new <tên project> để tạo project mới
   4. Truy cập vào project vừa tạo, gõ play run để chạy project
   5. Truy cập trình duyệt, <http://localhost:9000/>

* Import Play Framework vào eclipse
  + Tạo project trong cmd
  + Thay đổi cấu hình project, play eclipsify
  + Import Play vào eclipse: File/Import…->General->Existing………->next, chọn đường dẫn đến project vừa tạo-> finish
  + Chức năng của các file trong project vừa tạo và import vào eclipse
    - Application.java: Định nghĩa các hàm sẽ hiển thị lên trình duyệt
    - Index.html: hiển thị những gì lên trình duyệt
    - Cof/routes: Điều khiển và quản lý những gì được hiển thị lên trình duyệt
* Cách khởi động một trang web:
  + - 1. Khởi động server Tomcat
      2. Chạy file đã lập trình với đường dẫn: [http://localhost:8080/kpp/<ten\_file>.jsp](http://localhost:8080/kpp/%3cten_file%3e.jsp)

File đặt cùng thư mục với Tomcat

* Ghi chú: có thể chia nhỏ chương trình bằng cách đặt các file .jsp, sau đó include vào chương trình chính bằng cách: [%@include file = *"script.jsp"*%](mailto:%25@include%20file%20=%20%22script.jsp%22%25)
* Biến **session** là biến không cần khai báo, được dùng chung cho tất cả các trang
* Khối lệnh:
* String sql = "{?=call kpp\_owner.pkg\_web.get\_branch\_list(?)}";
* CallableStatement stm = conn.prepareCall(sql);
* stm.registerOutParameter(1, OracleTypes.CURSOR);
* stm.setString(2, user\_name);
* stm.execute();

Tương đương với lệnh select \* from trong mysql

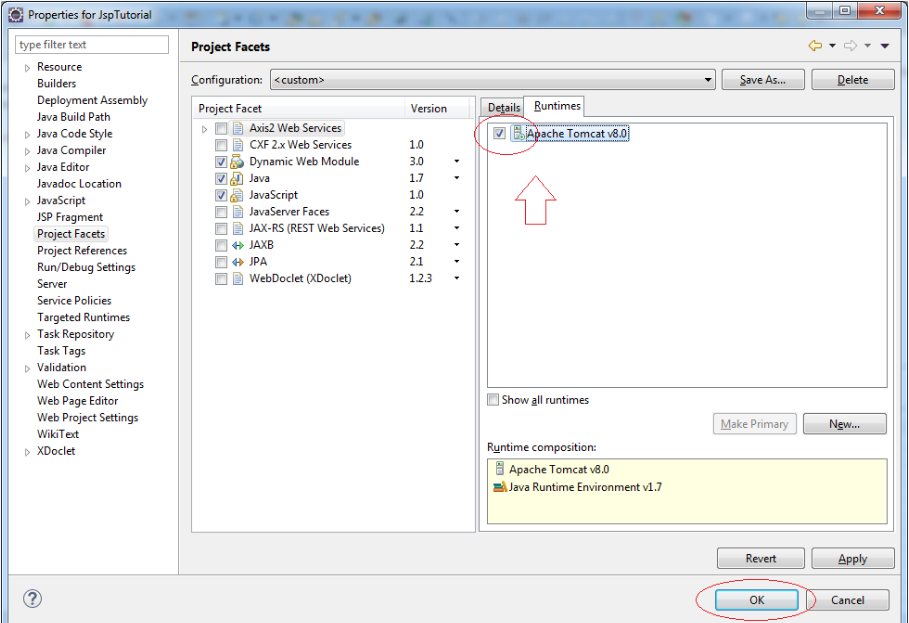
**Các lệnh trong jquery**

* Sự kiện **change** trong jquery là sự kiện mà sẽ bị tác động khi thay đổi giá trị của nó
  + Ví dụ: https://api.jquery.com/change/
* **Trigger** trong jquery, khi thực hiện sự kiện này thì sự kiện khác cũng được thực hiện theo

Ví dụ: <http://api.jquery.com/trigger/>

* **parseInt** trong java, chuyển đổi một chuỗi sang số nguyên hệ thập phân

ví dụ: <https://www.w3schools.com/jsref/jsref_parseint.asp>

* thuộc tính **val()** để get, set giá trị cho thuộc tính trong jquery
  + ví dụ: <https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_html_val_set>
* $(document).ready(function(){} tương đương với hàm $(function(){}
  + Ví dụ: <https://api.jquery.com/ready/>
* **Empty** trong jquery, xóa tất cả các phần tử có liên quan
  + Ví dụ: <https://api.jquery.com/empty/>
* **Append** trong jquery, nối phần tử vào, bao gồm cả các tag và định dạng
  + Ví dụ: <http://api.jquery.com/append/>
* **hasOwnProperty** kiểm tra xem đối tượng có chứa thuộc tính nào đó không?
  + Ví dụ: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/hasOwnProperty>
* **Serialize(),** mã hóa tập hợp các phần tử của form thành kiểu String để submit
  + Ví dụ: <http://api.jquery.com/serialize/>
* **On()** method: <https://www.w3schools.com/jquery/event_on.asp>
* Ví dụ tổng hợp trigger, on, alert, val:
* .jqGrid(): tạo bảng trong html
  + <https://free-jqgrid.github.io/getting-started/index.html#the_first_grid_code>
* Cách thêm thư viện Tomcat: 

Các lệnh trong HTML

* Thuộc tính rel="stylesheet": Liên kết đến 1 style sheet ở bên ngoài
  + <https://www.w3schools.com/tags/tag_link.asp>
* Thuộc tính rel="icon": xác định biểu tượng icon cho trang web
  + <https://www.w3schools.com/tags/att_link_sizes.asp>
* Thẻ <meta charset="utf-8">: Khai báo bảng mã cho website

**Học Bootstrap**

* Để sử dụng được Bootstrap, ta phải khai báo link: <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
* Chia lưới: Grid Basic
* Trình bày cho đẹp: Typography
* class="container": định dạng văn bản có khoảng cách trái và phải
* class="panel panel-default": định dạng đoạn văn bản có hộp bọc có đường viền và padding xung quanh nó
* class="panel-body": nội dung bên trong hộp panel
* <ul class="nav nav-tabs">
* <li class="active"><a data-toggle="tab" href="baocao-ty-le-msale">Báo cáo tỷ lệ Dinh</a></li>
* </ul>
  + : khai báo thanh điều hướng, chọn trước cột có thuộc tính class=”active”
* data-toggle="tab": để làm cho tab chuyển đổi
* class="tab-content": nội dung của tab
* class="tab-pane fade in active": làm cho tab mờ và sáng lên khi click vào, thêm thuộc tính in active để chọn tab hiện tại

**học .jqGrid()**

* iconSet: "fontAwesome": làm cho các biểu tượng đẹp hơn
* rownumbers: true: tạo thêm 1 cột số thứ tự ở phía trước bảng
* **idPrefix: "g1\_":** sử dụng dữ liệu có cùng giá trị id trong nhiều lưới trên trang
* sortname: "invdate" and sortorder: "desc": để sắp xếp lại các hàng theo thứ tự
* caption: "The grid…”: đặt tên cho bảng

**các thuộc tính trong jqGrid:**

* cmTemplate: Xác định một tập các thuộc tính ghi đè các giá trị mặc định trong colModel.
* cellattr: Hàm này thêm thuộc tính vào ô trong quá trình tạo dữ liệu - tức là linh động. Ví dụ, tất cả các thuộc tính hợp lệ cho ô của bảng có thể được sử dụng hoặc một thuộc tính kiểu với các thuộc tính khác nhau. Hàm nên trả về chuỗi.
  + Nếu đặt hàm này vào thuộc tính cmTemplate: có tác dụng trên cả hàng trong bảng
  + Nếu đặt hàm này vào thuộc tính colModel: chỉ có tác dụng trên 1 ô trong bảng
  + Định nghĩa hàm như sau:

function tongAttr2(rowId, val, rawObject) {

if (rawObject != null && Number(rawObject.sl\_dbh) < 20 ) {

return "style='font-weight:bold; background-color:gray'";

}

}

* Đoạn lệnh để trộn các ô trong 1 hàng: đoạn này nằm trong thuộc tính loadComplete

$(this).jqGrid('setGroupHeaders', {

useColSpanStyle: true,

groupHeaders:[

{startColumnName: 'w1\_th', numberOfColumns: 3, titleText: 'Tuần 1<br>'},

{startColumnName: 'w2\_th', numberOfColumns: 3, titleText: 'Tuần 2<br>'}

]

});

* Đoạn lệnh để tự động reset lại sau khi load xong, đoạn này nằm trong thuộc tính loadComplete

$('#btnXem').button('reset');

* viewrecords: true, Nếu true, jqGrid hiển thị số bắt đầu và kết thúc số liệu trong lưới,
* loadonce: true, Nếu true, lưới tải dữ liệu từ máy chủ chỉ một lần
* footerrow: true, Nếu thiết lập là true thì bạn sẽ đặt một bảng footer với một hàng dưới các bản ghi gird và phía trên máy nhắn tin. Số cột bằng các cột được chỉ định trong colModel

CÁCH XUẤT FILE EXCEL

* Add các thư viện và import vào eclipse:
* Tạo một đối tượng Map beans = **new** HashMap(); và thêm các giá trị beans.put("lst", lst); để xuất file
* Tạo một thư mục excel\_template trong đường dẫn chứa file, sau đó tạo file excel mẫu để add các giá trị vào
* Đây là đoạn lệnh để xuất file excel
* // Thực hiện xuất file excel
* String rootPath = session.getServletContext().getRealPath("");
* String templateFileFullName = rootPath + "excel\_template/BC\_CS\_MSALE.xls";
* **try** {
* XLSTransformer transformer = **new** XLSTransformer();
* InputStream inputStream = **new** BufferedInputStream(**new** FileInputStream(templateFileFullName));
* Workbook workbook = transformer.transformXLS(inputStream, beans);
* workbook.write(o);
* inputStream.close();
* o.flush();
* o.close();
* } **catch** (Exception e) {
* System.out.println(e.getMessage());
* } **finally** {
* conn.close();
* }

**Ngày 15/3: Học FullCalendar**

Hiện tiêu đề cho lịch

header: {

left: 'prev, today, next',

center: 'title',

right: 'month,agendaWeek,agendaDay,listMonth'

},

eventLimit: true, //giới hạn số sự kiện hiển thị trong 1 ngày, nếu nhiều sự kiện không thể chứa hết trong ô thì sẽ được hiển thị thêm ở đường link

defaultView: chế độ hiển thị mặc định

listDayFormat: false,//nếu false hiện thị thứ ở bên trái

listDayAltFormat : true,//nếu false hiện ngày tháng năm ở bên phải

noEventsMessage: "không có sự kiện nào"//hiển thị thông báo nếu không có sự kiện để hiển thị

defaultView: 'agendaWeek',//hiện thị theo kế hoạch tuần

allDaySlot: true,//hiện thị các sự kiện cả ngày hay không?

allDayText: 'cả ngày',//tên của sự kiện cả ngày

slotEventOverlap: false,//hiển thị các sự kiện có chồng lên nhau không?

agendaEventMinHeight: 100,//độ cao hiển thị của sự kiện

defaultView: 'basicDay',//chế độ hiện thị mặc định là hiển thị theo ngày

selectable: false,//cho phép chọn hay không?

weekends: false,//hiện ngày cuối tuần

hiddenDays: [0,4],//ẩn ngày chủ nhật và thứ 5

columnHeader : true,// có hiện thứ mấy hay không?

columnHeaderFormat : 'ddd M/D',//Định dạng kiểu hiện thị thứ, ngày trong tuần

resources: Định nghĩa các tài nguyên trong lịch.

select: Một phương pháp để lập trình chọn một khoảng thời gian.(lấy 1 khoảng thời gian)

dayClick: Được kích hoạt khi người dùng nhấp vào một ngày hoặc một thời gian.

editable: true, cho phép sự kiện kéo được

aspectRatio: Đặt tỷ lệ co chiều rộng / chiều cao của lịch.

columnHeaderText: function(mom) {

if (mom.weekday() === 5) {

return 'Đây là thứ 6';

}

else {

return mom.format('LLL');

}

},//Lập trình để Định dạng cách hiện thị thứ trong tuần

columnHeaderHtml: function(mom) {

if (mom.weekday() === 5) {

return '<b><u>Friday!</u></b>';

}

else {

return '<i>' + mom.format('LLL') + '</i>';

}

},//Lập trình để Định dạng cách hiện thị thứ trong tuần có chèn thêm các thẻ HTML

groupByResource: true, //cho phép hiển thị riêng các Room A, B, C, D hay không?

Ví dụ: <https://fullcalendar.io/docs/groupByResource>

scrollTime: Xác định khoảng tiến về phía trước của thanh cuộn ban đầu.

defaultView: Chế độ xem ban đầu khi lịch tải.

views: Định nghĩa một số chế độ xem lịch

resourceLabelText: Văn bản sẽ xuất hiện phía trên danh sách các tài nguyên.

Events: Định nghĩa các sự kiện

selectHelper: Có nên vẽ một sự kiện "giữ chỗ" trong khi người dùng đang kéo hay không.

slotDuration: '00:20': chia mỗi khoảng thời gian trong 1h ra 20 phút

navLinks: true : Nhấp vào bất kỳ ngày nào trong lịch và đến nơi bạn cần

resourceText: Xác định văn bản được hiển thị đầu mỗi tài nguyên.

resourceRender: Xác định thuộc tính các ô của tài nguyên

resourceRender: function(resource, leftCells, rightCells) {

if (resource.id == 'b') {

leftCells.css('background-color', 'red');

rightCells.css('background-color', 'gray');

}

}

Cài đặt (Set) các option trong Lịch

$('#select-G').on('click', function() {

$('#calendar').fullCalendar('select', '2018-03-07T02:00:00', '2018-03-07T07:00:00', 'g');

});

resourceColumns: Chia cột tài nguyên thành các cột nhỏ hơn và định nghĩa chúng

resourceOrder: Xác định trình tự sắp xếp của danh sách tài nguyên.

---------------------ngày 16/3--------------------------

eventOverlap: sự kiện sẽ chiếm hết ô và có cho phép các sự kiện trùng lặp thời gian với nhau không?

resources (as a JSON feed): có thể lấy resources từ một file khác, không hỗ trợ ở trình duyệt Cốc cốc

events: 'usa\_\_en@holiday.calendar.google.com', sự kiện của google

googleCalendarApiKey: 'AIzaSyDcnW6WejpTOCffshGDDb4neIrXVUA1EAE', key sự kiện của google

eventClick: function(calEvent, jsEvent, view) {} hàm xảy ra khi kích vào sự kiện

window.open(calEvent.url, 'gcalevent', 'width=700,height=600'); mở cửa sổ mới khi kích vào sự kiện

$(this).css('border-color', 'red'); thay đổi màu viền của ô sự kiện hiện tại

Làm cho sự kiện bên ngoài có thể kéo được vào lịch

// make the event draggable using jQuery UI

$(this).draggable({

zIndex: 999,

revert: true, // will cause the event to go back to its

revertDuration: 0 // original position after the drag

});

Cú pháp data trong jquery, để set giá trị cho một thẻ nào đó trong html, phải khai báo thẻ đó trong đoạn code

Ví dụ: <https://api.jquery.com/data/>

Droppable: Xác định xem các dragg jQuery UI bên ngoài có thể được kéo vào lịch không?.

Để phân chia thành 2 cột trong trang lịch, cần phải có đoạn <style>

title: $.trim($(this).text()), tên của sự kiện ngoài được lấy theo tên của phần tử text

$(this).data('event', { }); event phải đặt là event

revert: true, sự kiện có trở lại chỗ cũ khi đã kéo đi không?

revertDuration: Thời gian cần cho sự kiện để trở lại vị trí ban đầu sau khi kéo không thành công.

removeResource: xóa tài nguyên

addResource: thêm tài nguyên

gộp nhóm tài nguyên:

{

group: true,

labelText: 'Building',

field: 'building'

},

rendering: 'background': sự kiện màu nền (không có mô tả sự kiện, chỉ có màu nền thôi

Ngày 19/3:Học SQLITE tại trang <https://www.sqlite.org/index.html>

1. Download sqlite3 tại trang <https://www.sqlite.org/index.html>
2. Tạo thư mục sqlite2 tại ổ đĩa C và copy file sqlite.exe vào
3. Mở cửa sổ CMD và gõ câu lệnh để làm việc với file sqlite3
4. Tạo các ví dụ về DB mẫu và test

Ngày 20/3: Tiếp tục tìm hiểu về Sqlite và làm các bài tập

1. Tạo thư mục và copy file sqlite3.exe vào
2. Mở cmd và chạy file sqlite3.exe vừa copy vào
3. Tải và cài đặt trình duyệt FireFox
4. Tìm SQLite Manager và add vào FireFox
5. Khởi động lại FireFox, Kích vào biểu tượng Sqlite Manager góc trên bên phải trình duyệt
6. Làm các ví dụ lưu sự kiện vào DB và hiển thị ra trình duyệt, dùng thư viện FullCalendar

Tài liệu FullCalendar hay: https://stackoverflow.com/questions/29595441/bootstrap-fullcalendar-uncaught-typeerror-undefined-is-not-a-function

Học AJAX

<https://www.w3schools.com/jquery/jquery_ajax_intro.asp>

**Ngày 22/3: học W3CSS**

Đoạn JS điều khiển những hình ảnh hiện luân phiên cách nhau 2 giây

<script>

var myIndex = 1;

carousel();

function carousel() {

var i;

var x = document.getElementsByClassName("mySlides");

for (i = 0; i < x.length; i++) {

x[i].style.display = "none";

}

myIndex++;

if (myIndex > x.length) {myIndex = 1}

x[myIndex-1].style.display = "block";

setTimeout(carousel, 2000);

}

</script>

Lỗi getOutputStream() has already been called for this response:

=>Xóa phần khai báo HTML đi là không lỗi nữa

Input mà yêu cầu phải nhập dữ liệu => thêm thuộc tính **required**

*w3-modal:* Một phương thức là một hộp thoại / cửa sổ popup được hiển thị trên đầu trang hiện tại:

**--Ngày 27/3----**

The **w3-container** class adds a 16px left and right, top and bottom padding to any HTML element.