

05_03 Creating the detail page of the agenda section

In this exercise, you will create the detail page of the *agenda* drill down. Use the following step-by-step instructions. If there is something unclear, don't hesitate to refer to the final version of the site you find in the 'final' folder of the exercises.

Before you get started, you need to run a configuration file

This configuration file will change the dates of the events so that these events will take place in the future, regardless of when you are taking this course.

1. Browse to <http://localhost:8600/cfTraining/setup.cfm>.
2. You should have a message telling you that the event dates have been updated. Click on the link to go to the agenda page.
3. The agenda page should now contain a list of 7 events taking place in the future.

This is the master page of the data drill down that you developed in one of the previous videos. Regardless of when you are taking this course, this page now lists events you can work with.

Detect the presence of the url.eventID parameter in the URL

In this particular example, the master page and the detail page of the data drill down are handled in the same `.cfm` file. Consequently, you need to find a way for the 'agenda.cfm' page to detect the presence of the 'url.eventID' variable and to act accordingly.

1. Right after the `<div id="calendarContent">` (on my computer, on line 38), open a `<cfif>` tag that checks if the 'url.eventID' variable exists.

```
<div id="calendarContent">  
    <cfif isDefined('url.eventID')>  
        <cfquery datasource="hdStreet" name="rsCurrentEvents">  
code continues....
```

Remember that the name of the variable must be passed as a string to the 'isDefined()' function. That's why you need to enclose 'url.eventID' in quotes (single or double).

2. Immediately below the `<cfif>`, write a `<cfelse>` tag.

```
<div id="calendarContent">
  <cfif isDefined('url.eventID')>
    <cfelse>
      <cfquery datasource="hdStreet" name="rsCurrentEvents">
code continues.....
```

3. Close the `<cfif>` just before the `<div id="calendarContent">` closes.

```
      <td><a href="agenda.cfm?eventID=#fld_eventID#">Read
        More</a></td>
    </tr>
  </cfoutput>
</table>
</cfif>
</cfif>
</div>
<div id="calendarSideBar">
```

For the sake of clarity, add some ColdFusion comments in your code

To add a comment, programming languages use specific symbols to mark the beginning and the end of a comment.

HTML uses '`<!--`' to mark the beginning of a comment and '`-->`' to mark the end of a comment.

```
<!-- I'm an HTML comment -->
```

ColdFusion uses something very similar for its commenting system. '`<!---`' marks the beginning of a ColdFusion comment and '`--->`' marks the end of a ColdFusion comment.

```
<!---I'm a ColdFusion comment--->
```

Did you see the difference between a ColdFusion comment and an HTML comment? A ColdFusion comment has one more '-' than an HTML comment.

1. I suggest you take some time to comment your code as illustrated below.

```
<div id="calendarContent">
  <cfif isDefined('url.eventID')>
    <!---Output a single agenda if url.eventID is defined--->
  <cfelse>
    <!---Output the upcoming event table--->
    The code for the events table goes here
  </cfif>
</div>
```

In the <cfif> block, create the 'rsSingleEvent' query

1. Just below the first comment, create a <cfquery> tag block to query the 'hdStreet' datasource. Name your query 'rsSingleEvent'.

```
<!---Output a single event if url.eventID is defined--->
<cfquery datasource="hdStreet" name="rsSingleEvent">
</cfquery>
```

2. In the body of the <cfquery> tag, select 'fld_eventID', 'fld_eventName', 'fld_eventDateTime', 'fld_eventLocation', 'fld_eventVenue' and 'fld_eventDescription' from the 'tbl_events' table.
3. Add a WHERE clause to select the only event whose ID matches the value of the 'url.eventID' parameter.

Make sure your <cfquery> tag block is the same as below.

```
<cfquery datasource="hdStreet" name="rsSingleEvent">
  SELECT FLD_EVENTID, FLD_EVENTNAME, FLD_EVENTDATETIME,
  FLD_EVENTLOCATION, FLD_EVENTVENUE, FLD_EVENTDESCRIPTION
  FROM TBL_EVENTS
  WHERE FLD_EVENTID = #url.eventID#
</cfquery>
```

4. Write a <cfdump /> below the <cfquery/>.

```
<cfdump var="#rsSingleEvent#" label="single event query" />
```

5. Run the page
6. Click on one of the 'Read more' link. The query should be dumped in the main area of the page.
7. Test another query by hitting the 'back' button of your browser and clicking on another 'Read More' link.

Output the event's name and description in the main area of the page

Now that the data is available to the page, you can output the event's detail in the main area of the page

1. Delete or comment out your latest `<cfdump />`. Replace it by a `<cfoutput>` tag block.
2. Output the name of the event in a `<h1>` tag.
3. Output the description of the event just below the title.
4. Just below the `</cfoutput>` tag create a link that goes back to the 'agenda.cfm' page.

Your code should look like the following

```
</cfquery>
<cfoutput>
  <h1>#rsSingleEvent.fld_eventName#</h1>
  #rsSingleEvent.fld_eventDescription#
</cfoutput>
<a href="agenda.cfm">Go back to the agenda</a>
```

Notice that the link points to 'agenda.cfm' without the 'eventID' url parameter.

5. Save and run the page in the web browser.
6. Test a few events and make sure the *"Back to the agenda"* link works as expected.

Output the event's details in the side bar

You have outputted the event's title and the event's description in the main area of the page. Now, you will output the remaining event's data in the sidebar of the page.

1. Create a `<cfif>` tag block with a `<cfelse>` just after the `<div id="calendarSideBar">` opens (line 76 on my computer).
2. Close the `<cfif>` block just before the `<div id="calendarSideBar">` closes.

```
<div id="calendarSideBar">
  <cfif isDefined('url.eventID')>
  <cfelse>
```

code of the sidebar of the master page as written in a previous video

</cfif>

</div>

3. Create a `<cfoutput>` tag block in the `<cfif>`, above the `<cfelse>` tag.
4. Just below the `<cfoutput>` add a `<h1>` title that read, *"Event Details"*.
5. Copy the `<div id="EventDetails">` block that is below the `<cfelse>` and paste it in the `<cfoutput>` block you created in step 3.
6. In the copied "EventDetails" div, update the existing code so that the data comes from the 'rsSingleEvent' query and not from the 'rsCurrentEvents' anymore.

Your final code should look like the following

```
<div id="calendarSideBar">
  <cfif isDefined('url.eventID')>
    <cfoutput>
      <h1>Event details</h1>
      <div id="EventDetails">
        <p id="eventDate"><span
          id="month">#dateFormat(rsSingleEvent.fld_eventDateTime,
            'mmm') #</span> <span
            id="days">#dateFormat(rsSingleEvent.fld_eventDateTime,
              'dd') #</span></p>
        <h2>#rsSingleEvent.fld_eventName#</h2>
      </div>
      <p> #rsSingleEvent.fld_eventLocation# <br />
        #rsSingleEvent.fld_eventvenue# </p>
    </cfoutput>
  <cfelse>
code continues....
```

Save and test the page. The links in the upcoming events table should be activated and you should be able to go back and forth from the table to the individual events!

Some housekeeping!

Now that you can dynamically display all the events, the 'events' directory of your site and all the '.html' files it contains are no longer needed. Therefore, you can safely delete the 'events' folder of your ColdFusion project.