CPRG256

FINAL PROJECT

DUE DATE: 10:00 AM MONDAY APRIL 15, 2019. NO LATE SUBMISSIONS WILL BE ACCEPTED. ONLY STUDENTS WHO HAVE SUBMITTED THEIR PROJECT WILL BE ALLOWED TO DEMONSTRATE. THIS IS AN INDIVIDUAL ASSIGNMENT SUBMISSION

Create a directory ‘cprg256/finalproject’. Create the required subdirectories for the following applications.

PART A:

Create a subdirectory titled ‘welldata’.

You will create an ‘index.html’ page that will display and search for data as described below.

You have been provided with two files: ‘welldata.xml’ and ‘productiondata.xml’. The ‘welldata.xml’ contains the following data:

* Location of the well using the Alberta Township System (See below for details)
  + Depth of the well in meters
  + Perforation depth (top) in meters
  + Perforation zone in meters
  + Pump stroke length
  + Strokes per minute

A well location in accordance with the Alberta Township System shown below:

Example: B15-98-17-W5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Section |  | Township |  | Range |  | Meridian |
| B15 | - | 98 | - | 17 | - | W5 |
| VALID DATA RANGE:  Section: A-D and 1-16  Township: 1-126  Range: 1-24  Meridian: W and 4-6 | | | | | | |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

The ‘productiondata.xml’ file contains the following data:

* + A well location
  + Date
  + Oil production in m3/day
  + Water production in m3/day
  + Gas production in 103 m3/day

Your application will do the following:

Search by specific location of a well.

The user can enter the full location using the Alberta Township System e.g. B15-98-17-W5. Your application will search ‘welldata.xml’ to find the oil well. If the oil well is found, it searches ‘productiondata.xml’ for the production information for that well. The data is then displayed on the web page in separate <form> or <div> element. If no well is found, an error message will be printed to the page.

Any data input must be validated according to the Valid Data Range listed above.

Include buttons that will start the search or clear data from the form.

The application must use AJAX to process the xml files and display the search responses.

(20 Marks)

PART B:

Create a subdirectory titled ‘rentalcar’. Your application will require ‘rentalclients.json’.

Your task is to create the Web Application for ‘Dodgy Brakes Car Rental’. The index.html page will have a ‘header’ and ‘footer’ that will display a company logo and the current date and time. Include a suitable background photo for the page and a link to ‘rental.html’.

The ‘rental.html’ page, has the following functionality:

Your page will include a ‘search’ field and a ‘car rental’ form. The ‘car rental’ form will be un-editable until a client is found and selected.

The ‘search’ field will search ‘rentalclient.json’ by last name. As you enter letters, the search function will find and display all the clients whose last names start with those letters.

Example: If you enter ‘B’, all the clients with a last name’ that starts with ‘B’ will be displayed. When you type ‘Br’ all the clients with a last name’ that starts with ‘Br’ are displayed.

NOTE: Only display the Last and First Names of the clients.

You can then select a client from the list. This ‘event’ will make the ‘car rental’ form editable.

After a client is selected, the client information will be displayed in the ‘car rental’ form. E.g. last name, first name, address, state/prov, email and phone.

The client can then choose to rent a vehicle. The rental choices are:

Compact $15/day

Mid-size $20/day

Luxury $35/day

Van/Truck $40/day

Note: Include a picture of a vehicle next to each of the rental choices.

Options: Roof Rack or Bicycle Rack extra $5/day

GPS extra $10

Child Seat free

The client can rent from 1 to 30 days.

After this information is selected, the total of the rental is calculated and the complete client and rental information will be displayed on the page. This can appear on the page or in an ‘alert’ field.

(20 Marks)

PART C:

Create a subdirectory titled ‘finalquiz’. You are to create a web application that will display a quiz and create the necessary JavaScript to complete the following:

* Process the ‘FinalQuiz.xml’ file to display the multiple choice questions within the index.html page. Each question has 4 possible answers.
* Create a function that will grade the quiz and display the grade out of 5. Within ‘FinalQuiz.xml’ is an element ‘rightanswers’ that has a comma delimited string with the correct answers. Use CSS to create an appropriate interface.

Here is an example of what the display could look like:

Question 1:

In a switch statement, the \_\_\_\_\_\_\_\_ case clause is used to process exceptional conditions and is usually listed last.

* A) break
* B) default
* C) else
* D) then

At the end of the quiz, create a button ‘Grade Quiz’, that will display the number of correct answers out of five. E.g. ‘Grade 3/5’

(20 Marks)

REQUIREMENTS

You must use JavaScript to implement your programming solutions. NOTE: If you plan to use an Third Party Applications (eg. Bootstrap), you must consult your instructor in advance.

You are required to use CSS/Styling to create pleasing and useful interfaces for your applications.

30% of the grade of any of these applications will be allocated to the User Interface. Your interfaces must be useful and adhere to good design.

‘Zip’ ALL of your project files and upload them to the Brightspace site under Assignments/Final Project