

Homework 3 - Functions

- Due Sep 23 by 11:59pm
- Points 15
- Submitting a text entry box or a website url
- Available Sep 3 at 5am - Dec 13 at 11:59pm

Homework 3 - Functions

What you will do:

- Create a web page with the heading Homework 3 - Functions
- The contents of this page should be, besides the heading:
 - hyperlink to Part 1 web page with a short description about what will be presented
 - hyperlink to Part 2 web page with a short description about what will be presented
 - hyperlink to Extra credit (if you will develop it) with a short description about what will be presented
 - hyperlink to go back to the main page of the course (Homework 1)

DO NOT USE ALERT BOX FOR ANY ERROR MESSAGES!!

For Part 1:

- Create a webpage that contains:
 - the heading **Student Grades**,
 - a form with input boxes for Homework Average, Mid-Term Exam Score, Final Exam Score, and Participation - all those grades will be entered as integers
 - you should have a button for submitting the data and a button to clear the form
 - you should have a textarea (or div or paragraph element) to present final results and/or error message
- Create a script that:
 - checks for valid input, i.e., that the input is between 0-100 and that, of course, the input are all numbers.
 - If all input is valid then calculate and display the student's final average according to the formula:
 $final\ average = (.5 * hwAvg) + (.2 * midExam) + (.2 * finalExam) + (.1 * participation)$
 - in the output area, you will present the student's final average (rounded to an integer) and final letter grade using the following criteria:

Points/Percentage to
Letter Grade

Percentage	Letter Grade
90-100	A
80-89	B
70-79	C
60-69	D
0-59	F

- If the student's final grade is 'D' or 'F' then display "**Student must retake the course**".
- If the input of the user is invalid, you will present an error message and, of course, no result will be presented

Note: The user should be able to reuse that same form for another student!

For Part 2:

A large company pays its salespeople on a commission basis. The salespeople receive \$250 per week, plus 9% of their gross sales for the week.

For example, a salesperson who sells \$5000 in a week, will receives \$250 **PLUS** 9% of \$5000, which will be a total of \$700. You have been supplied with a list of the items that can be sold by each salesperson. The value of these items are as follows:

Sale Price
per item #

Item Value

1	20.99
2	12.75
3	9.95
4	35.89

Create a web page that might look similar to the layout shown in the image below.

Your page does not need to contain the input box for the name of the sales person as shown in the image below.

There will be other input boxes so the user will input the quantity of items sold, for each item.

You will use that input to calculate how much the salesperson sold and how much he/she will earn, and then output the results.

TIP: Be sure to round the outputted results to 2 decimal places (money is not represented with more than 2 decimals, right?).

Use input boxes to input the number sold for each of the items 1-4.

You should also check for valid numeric input (make sure the number of items sold is never negative, < 0 , as nobody sells negative quantity of items, right?)

The results of **total sold in dollars** and the **total money that will be received by the sale person** should be right-justify (aligned to the right)

Here is an image showing a possible display that is using tables, and form elements (labels, input boxes, buttons, etc.) and some tips:

- The boxes for seller's name, and Item 1, Item 2, Item 3, Item 4 will be the boxes that the user will type data - as mentioned before, you do not need to have the input box for the seller's name.
- All the boxes below Qty Sold and Total should be readonly as you will present the values from the calculations in your code - for example, the Qty Sold box beside the Item 1 should present the number typed by the user for the quantity the seller sold of Item 1, then, below the Total, the box that is related to Item 1 will show the gross amount of that sale which, of course will be the multiplication of the quantity sold of item 1 by the price of Item 1 (20.99), etc.

- In the box beside **Total Amount Sold**, you will present the sum of all the values for each of the Total boxes (for item 1, item 2, item 3, item 4) and, again, remember to only present a number with 2 decimals.
- Calculate the 9% of that Total Amount Sold value and add 250 to that result and this is the value you will present in the box beside **Total Weekly Earnings**, remember again to present a number with 2 decimals only.
- For the input boxes besides **Item 1**, **Item 2**, **Item 3**, and **Item 4**, it will be better if you use input type="number" because if you use input type="text" you will need to test if the user typed a REAL number before doing ANY type of Math operation with that value!
- If you need to issue an error message - for example, the user did not type a number for the Item 1, or Item 2, or Item 3, or Item 4 boxes, and/or the user typed a negative number in one of these boxes - you should not present any result value, only the error message.
- If the user fixes the error to a valid input, you should present the result values and, of course, clear the error message

Homework 3 - Part 2

Seller's Name:

For each item type the quantity sold by the seller

Item 1:

Item 2:

Item 3:

Item 4:

Item#	Price	Qty Sold	Total \$
1	\$20.99	<input type="text"/>	<input type="text"/>
2	\$12.75	<input type="text"/>	<input type="text"/>
3	\$9.95	<input type="text"/>	<input type="text"/>
4	\$35.89	<input type="text"/>	<input type="text"/>
Total Amount Sold			<input type="text"/>
Total Weekly Earnings			<input type="text"/>

Extra Credit (2 points if it is working as requested):

Computers are playing an increasing role in education! There are many interesting educational games out there!

- Write a program that will help an elementary school student learn multiplication.

- Use **Math.random** to produce two one-digit integers (**NO DECIMALS**) from 0 to 9.
- With those two numbers, you should present a question such as: ***How much is X times Y?*** where ***X*** is one of the random numbers and ***Y*** is the other random number
- The student then types the answer into an input box you will provide.
- Your program checks the student's answer.
- If it is correct, display the message "Very good!" in a text area or another part of the page, and then shows a question to the user to see whether he/she wants to keep playing.
 - If the user responds **YES** (click on a button that says YES), you will then generate another question
 - If the user responds **NO** (click on a button that says NO), you will present a message such as ***"Thanks for playing, see you next time"*** and you **will NOT present another question.**
- If the answer to the multiplication question is wrong, display the message ***"No. Please try again."*** directly on the page in a textarea or another part of the page, and let the student try the same question again repeatedly until the student finally gets it right.

To submit your homework:

- Create the **main page for Homework 3**
- Create the pages for **Part 1, Part 2, Extra Credit** (if you are doing it) - **make sure to include a hyperlink to the main Homework 3 page on each of those pages**
- Update your **Homework 1** page (main course page) to include the **hyperlink to Homework 3**
- Upload all the created or updated files to the web server
- Test the URL of your **Homework 1** page in the browser and make sure that the link to **Homework 3** is working and test the links to all the parts of your Homework 3 from the Homework 3 page
- If everything is ok, then submit the URL of **Homework 1** in the space provided - from that page, I should be able to click on the hyperlink to go to the Homework 3 page and then go to each part to evaluate your work

Checklist for grading:

- your page can be seen in the browser with the URL you submitted with no errors such as NOT FOUND or FORBIDDEN and you submit the Homework 1 URL with the link to this assignment updated - **1 point**
- the script for each case is working as requested (with the tests requested and also with the output requested) - **12 points** (6 points for each part)
- you thought about the user experience/user interface (providing good navigation, using right input fields, making sure the user will not be typing anything that will return "weird" results such as NaN, etc.) - **1 point**
- your HTML and CSS are valid for all the pages - **1 point**

Note about extra credit points: The extra credit points, if it is working correctly, will be shown added to the total of the points of the assignment. **If you submit the assignment late, the extra points will be added and then deductions applied according to the syllabus**