

## CSCI 3300 Assignment 4

**Total estimated time** for this assignment: **7 hours**

When you see “Richard Ricardo” in the example screen captures, change it to **<your name>**.

When you see “Richard” in the example screen captures, change it to **<your first name>**.

If you do not put **<your name>** / **<your first name>** in the above mentioned fields, you will get **0 points** for the question(s).

**No two students** should submit webpages with exactly the same code, or same content, or same layout, or same color combination. If found, **both** students will get **0 points**.

Create a folder on your hard disk, name the folder **lastname\_firstname\_assignment4**. Save all the files from this assignment in this folder.

Create the following subfolders (in the folder lastname\_firstname\_assignment4): **q1, q2, q3**.

As a result, you should have the following folder (directory) structure for this assignment: (-2 points if wrong)

- lastname\_firstname\_assignment4\q1\
- lastname\_firstname\_assignment4\q2\
- lastname\_firstname\_assignment4\q3\

Use Firefox **Web Console** (Tools > Web Developer > Web Console) to help debugging JavaScript. All html (JavaScript) files must pass JavaScript validation at Firefox Web Console without any error, without any warning (**-2 points for each error, each warning**).

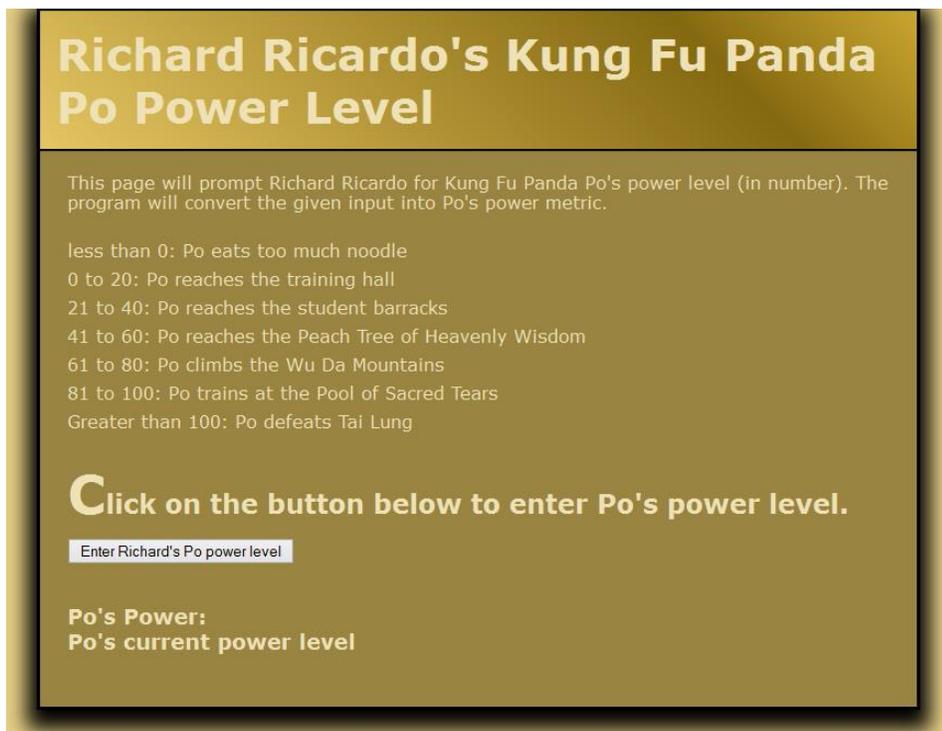
- Turn on your Firefox **Menu Bar** (Hint: right-click).
- Make sure your Firefox is up to date (Help > About Firefox).
- Turn on Web Console when you code (Tools > Web Developer > Web Console).
- Fix any error/warning immediately when you see them. Do NOT accumulate errors.

When you view page source in a web browser, **<!DOCTYPE html>** must be at the top of every page. In other words, all pages must be written in HTML5. (**-20 points** if not)

**Question 1** – JavaScript Chapter 3 (30 points)**Estimated time: 2 hours**

- Save question 1 files in subfolder “**lastname\_firstname\_assignment4\q1**”. (-25 points if no JavaScript)
- Create a web page that displays the default information (example shown). (1 point each, total 5 points)
- The initial page and related outputs should look like the examples shown below.
- Create your page using “<your name>’s Kung Fu Panda Po Power Level” as the page title. Save the page as **index.htm**. Document the html file with html and JavaScript comments. (1 point each, total 3 points)
- **Prompt** the user to **enter a number** representing Kung Fu Panda Po's power level. (3 points)
- The power level conversions are as follows: (10 points for logic, 5 points for output format)
  - less than 0: Po eats too much noodle
  - 0 to 20: Po reaches the training hall
  - 21 to 40: Po reaches the student barracks
  - 41 to 60: Po reaches the Peach Tree of Heavenly Wisdom
  - 61 to 80: Po climbs the Wu Da Mountains
  - 81 to 100: Po trains at the Pool of Sacred Tears
  - Greater than 100: Po defeats Tai Lung
- Create a css file named **style.css** to format index.htm by creating your own layout (no two students should have the same layout). Use css comments to document the css program. You can use **the same (or similar) css file(s)** to format all questions. (1 point each, total 2 points)

Example: Initial Page



Example: Input q1a



**Richard Ricardo's Kung Fu Panda Po Power Level**

This page will prompt Richard Ricardo for Kung Fu Panda Po's power level (in number). The program will convert the given input into Po's power metric.

less than 0: Po eats too much noodle  
0 to 20: Po reaches the training hall  
21 to 40: Po reaches the student barracks  
41 to 60: Po reaches the Peach Tree of Heavenly Wisdom  
61 to 80: Po climbs the Wu Da Mountains  
81 to 100: Po trains at the Pool of Sacred Tears  
Greater than 100: Po defeats Tai Lung

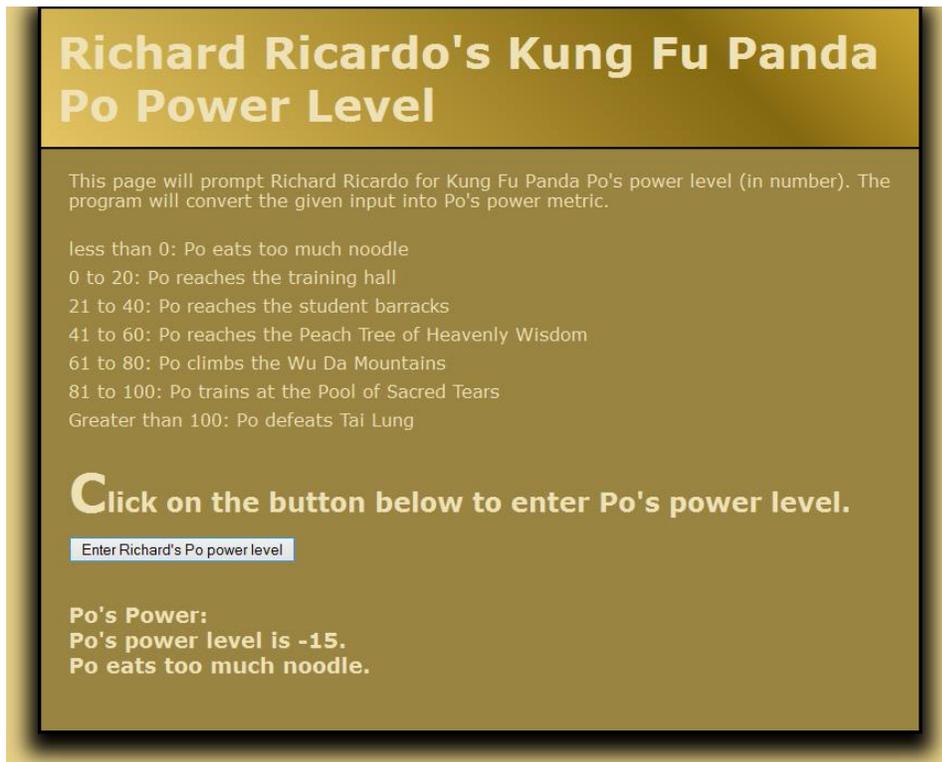
**C**lick on the button below to enter Po's power level.

Enter Richard's Po power level

Po's Power:  
Po's current power level

Enter po's power level:  
-15  
OK Cancel

Example: Output q1a



**Richard Ricardo's Kung Fu Panda Po Power Level**

This page will prompt Richard Ricardo for Kung Fu Panda Po's power level (in number). The program will convert the given input into Po's power metric.

less than 0: Po eats too much noodle  
0 to 20: Po reaches the training hall  
21 to 40: Po reaches the student barracks  
41 to 60: Po reaches the Peach Tree of Heavenly Wisdom  
61 to 80: Po climbs the Wu Da Mountains  
81 to 100: Po trains at the Pool of Sacred Tears  
Greater than 100: Po defeats Tai Lung

**C**lick on the button below to enter Po's power level.

Enter Richard's Po power level

**Po's Power:  
Po's power level is -15.  
Po eats too much noodle.**

## Example: Input q1b



**Richard Ricardo's Kung Fu Panda Po Power Level**

This page will prompt Richard Ricardo for Kung Fu Panda Po's power level (in number). The program will convert the given input into Po's power metric.

less than 0: Po eats too much  
0 to 20: Po reaches the training hall  
21 to 40: Po reaches the student barracks  
41 to 60: Po reaches the Peach Tree of Heavenly Wisdom  
61 to 80: Po climbs the Wu Da Mountains  
81 to 100: Po trains at the Pool of Sacred Tears  
Greater than 100: Po defeats Tai Lung

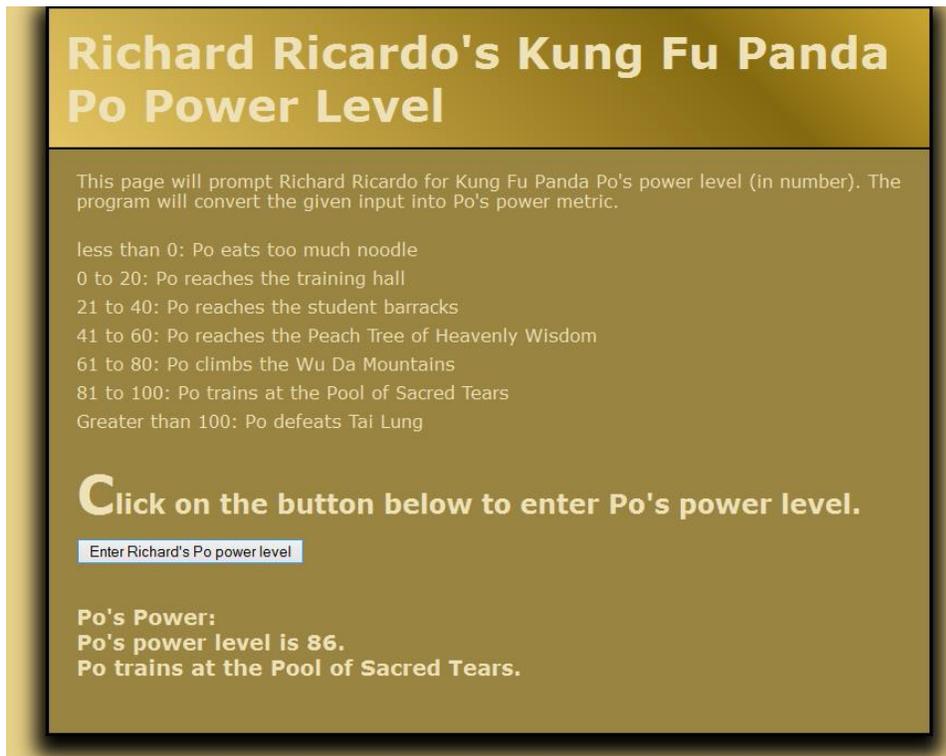
**C**lick on the button below to enter Po's power level.

Enter Richard's Po power level

Po's Power:  
Po's power level is -15.  
Po eats too much noodle.

The screenshot shows a modal dialog box titled "Enter po's power level:" with a text input field containing the number "86". Below the input field are "OK" and "Cancel" buttons. The background page is dimmed and shows the same content as the output screenshot, but with the power level set to -15.

## Example: Output q1b



**Richard Ricardo's Kung Fu Panda Po Power Level**

This page will prompt Richard Ricardo for Kung Fu Panda Po's power level (in number). The program will convert the given input into Po's power metric.

less than 0: Po eats too much noodle  
0 to 20: Po reaches the training hall  
21 to 40: Po reaches the student barracks  
41 to 60: Po reaches the Peach Tree of Heavenly Wisdom  
61 to 80: Po climbs the Wu Da Mountains  
81 to 100: Po trains at the Pool of Sacred Tears  
Greater than 100: Po defeats Tai Lung

**C**lick on the button below to enter Po's power level.

Enter Richard's Po power level

**Po's Power:  
Po's power level is 86.  
Po trains at the Pool of Sacred Tears.**

The screenshot shows the same web application as the input example, but with the power level set to 86. The output text is bolded and indicates that Po trains at the Pool of Sacred Tears.

**Question 2** – JavaScript Chapter 3 (35 points)**Estimated time: 2 hours**

- Save question 2 files in subfolder “**lastname\_firstname\_assignment4\q2**”. (-30 points if no JavaScript)
- Create a web page that displays the default information (example shown). (1 point each, total 5 points)
- The initial page and related outputs should look like the examples shown below.
- Create your page using “<your name>’s Kung Fu Panda Po Body Mass Index” as the page title. Save the page as **index.htm**. Document the html file with html & JavaScript comments. (1 point each, total 3 points)
- Write a program that calculates and displays Kung Fu Panda Po’s body mass index (BMI). Po’s BMI is calculated with the following formula: (5 points)
  - $BMI = weight \times 703 / height^2$
  - Weight is in pounds, and height is in inches
- The program should also indicate if Po has optimal weight, is underweight, or is overweight.
  - If Po's BMI is less than 18.5. Po is underweight
  - If Po's BMI is between 18.5 and 25. Po has optimal weight
  - If Po's BMI is greater than 25. Po is overweight (15 points for logic, 5 points for output format)
- Note: The BMI reading is **not for humans**. It is **modified** for Po. Please do not use it for humans.
- Hint: You may use `toFixed()` JavaScript method to display a number with the exact number of decimals.
- Create a css file named **style.css** to format index.htm by creating your own layout (no two students should have the same layout). Use css comments to document the css program. (1 point each, total 2 points)

Example: Initial Page

**Richard Ricardo's Kung Fu Panda Po Body Mass Index**

This page will prompt Richard Ricardo for Kung Fu Panda Po's weight and height. The program will convert the given input into Po's body mass index, and will inform Richard if Po is overweight or underweight.

$BMI = weight \times 703 / height^2$   
 Weight in pounds. Height in inches.  
 If Po's BMI is less than 18.5. Po is underweight  
 If Po's BMI is between 18.5 and 25. Po has optimal weight  
 If Po's BMI is greater than 25. Po is overweight

**C**lick on the button below to enter Po's weight and height.

Enter Richard's Po weight and height

**Po's BMI:**  
**Po's current Body Mass Index**

## Example: Input q2a

The image shows two sequential screenshots of a web application titled "Richard Ricardo's Kung Fu Panda Po Body Mass Index".

**Left Screenshot:** The page displays the title and a description: "This page will prompt Richard Ricardo for Kung Fu Panda Po's weight and height. The program will convert the given input into Po's body mass index, and will inform Richard is Po is overweight or underweight." Below this is the formula  $BMI = \text{weight} \times 703 / \text{height}^2$  and instructions: "Weight in pounds. Height in inches. If Po's BMI is less than 18.5, Po is underweight. If Po's BMI is between 18.5 and 25, Po has optimal weight. If Po's BMI is greater than 25, Po is overweight." A modal dialog box titled "Enter po's weight" is open, with a text input field containing "100" and "OK" and "Cancel" buttons. Below the dialog, a button labeled "Enter Richard's Po weight and height" is visible.

**Right Screenshot:** This screenshot is identical to the left one, but the modal dialog box is now titled "Enter po's height" and the text input field contains "60".

## Example: Output q2a

The image shows a single screenshot of the web application titled "Richard Ricardo's Kung Fu Panda Po Body Mass Index".

The page displays the title and the same description as the input screenshots. The formula  $BMI = \text{weight} \times 703 / \text{height}^2$  and instructions are also present. Below the instructions, a button labeled "Enter Richard's Po weight and height" is visible.

The output section shows the following results:

- Po's BMI:**
- Po's weight is 190 pounds.**
- Po's height is 60 inches.**
- Po's BMI is 37.10.**
- Po is overweight.**

## Example: Input q2b

The image shows two sequential screenshots of a web application titled "Richard Ricardo's Kung Fu Panda Po Body Mass Index".

**Left Screenshot:** The page displays the title and introductory text. A modal dialog box titled "Enter po's weight" is open, with the input field containing the value "160". Below the input field are "OK" and "Cancel" buttons. The background page is dimmed.

**Right Screenshot:** The modal dialog box is now titled "Enter po's height" and the input field contains the value "68". The "OK" and "Cancel" buttons are still present. The background page remains dimmed.

Both screenshots show the following text on the page:

This page will prompt Richard Ricardo for Kung Fu Panda Po's weight and height. The program will convert the given input into Po's body mass index, and will inform Richard is Po is overweight or underweight.

BMI = weight x 703 / height<sup>2</sup>  
 Weight in pounds. Height in inches.  
 If Po's BMI is less than 18.5, Po is underweight.  
 If Po's BMI is between 18.5 and 25, Po has optimal weight.  
 If Po's BMI is greater than 25, Po is overweight.

Click on the button below to enter Po's weight and height.

Enter Richard's Po weight and height

Po's BMI:  
 Po's current Body Mass Index

## Example: Output q2b

The image shows a single screenshot of the web application titled "Richard Ricardo's Kung Fu Panda Po Body Mass Index".

The page displays the title and introductory text. The modal dialog box is no longer present. The background page is no longer dimmed.

This page will prompt Richard Ricardo for Kung Fu Panda Po's weight and height. The program will convert the given input into Po's body mass index, and will inform Richard is Po is overweight or underweight.

BMI = weight x 703 / height<sup>2</sup>  
 Weight in pounds. Height in inches.  
 If Po's BMI is less than 18.5, Po is underweight.  
 If Po's BMI is between 18.5 and 25, Po has optimal weight.  
 If Po's BMI is greater than 25, Po is overweight.

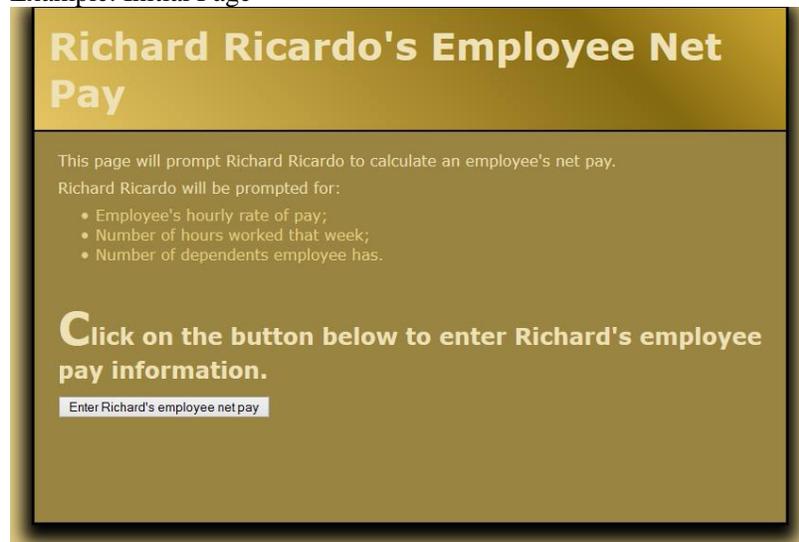
Click on the button below to enter Po's weight and height.

Enter Richard's Po weight and height

Po's BMI:  
 Po's weight is 160 pounds.  
 Po's height is 68 inches.  
 Po's BMI is 24.33.  
 Po has optimal weight.

**Question 3 – JavaScript Chapter 3 (35 points)****Estimated time: 3 hours**

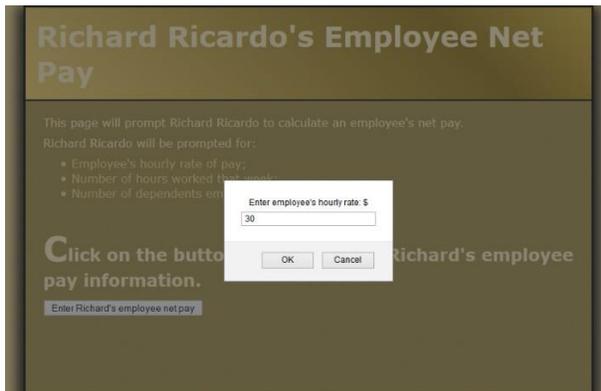
- Save question 3 files in subfolder “**lastname\_firstname\_assignment4\q3**”. (-30 points if no JavaScript)
- Create a web page that displays the default information (example shown). (1 point each, total 3 points)
- The initial page and related outputs should look like the examples shown below.
- Create your page using “<your name>’s Employee Net Pay” as the page title. Save the page as **index.htm**. Document the html file with html and JavaScript comments. (1 point each, total 3 points)
- Write a program that calculated an employee’s net pay. (1 point each, total 3 points)
  - The program should prompt for
    - Hourly pay rate
    - Number of hours worked in a week
    - Number of dependents claimed
  - If the employee works more than 40 hours in a week, overtime is calculated at 1.5 times the regular hourly rate. (10 points for logic)
  - Taxes are then deducted from the gross pay as follows: (10 points for logic)
    - No dependents: tax rate is 28% (5 points for output format)
    - 1 to 3 dependents: tax rate is 25%
    - 4 to 6 dependents: tax rate is 15%
    - More than 6 dependents: tax rate is 10%
- Create a css file named **style.css** to format index.htm by creating your own layout (no two students should have the same layout). Use css comments to document the css program. (1 point each, total 2 points)

**Example: Initial Page**

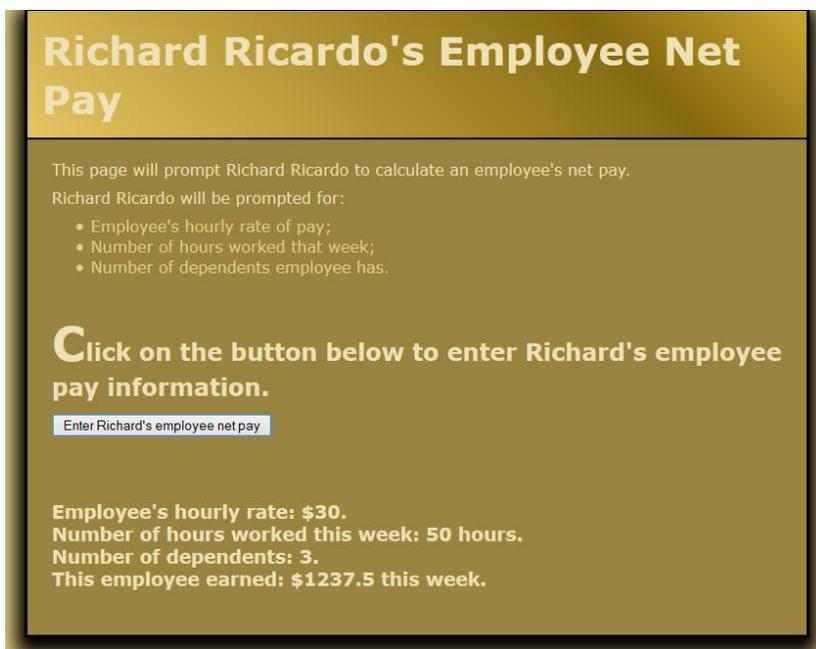
<b>Input q3a</b>	<b>Output q3a and logic</b>
Employee's hourly rate: \$30. Number of hours worked this week: 50 hours. Number of dependents: 3.	First 40 hours pay = $\$30 \times 40 \text{ hours} = \$1200$ Next 10 hours pay (overtime) = $\$30 \times 10 \text{ hours} \times 1.5 = \$450$ Gross pay = $\$1200 + \$450 = \$1650$ Tax rate for 3 dependents = 25% Tax deduction = $\$1650 \times 25\% = \$412.5$ Net pay = $\$1650 - \$412.5 = \underline{\$1237.5}$ This employee earned: <u>\$1237.5</u> this week.
<b>Input q3b</b>	<b>Output q3b and logic</b>
Employee's hourly rate: \$40. Number of hours worked this week: 30 hours. Number of dependents: 4.	First 40 hours pay = $\$40 \times 30 \text{ hours} = \$1200$ Next 10 hours pay (overtime) = $\$30 \times 0 \text{ hours} \times 1.5 = \$0$ Gross pay = $\$1200 + \$0 = \$1200$ Tax rate for 4 dependents = 15% Tax deduction = $\$1200 \times 15\% = \$180$ Net pay = $\$1200 - \$180 = \underline{\$1020}$

This employee earned: \$1020 this week.
---

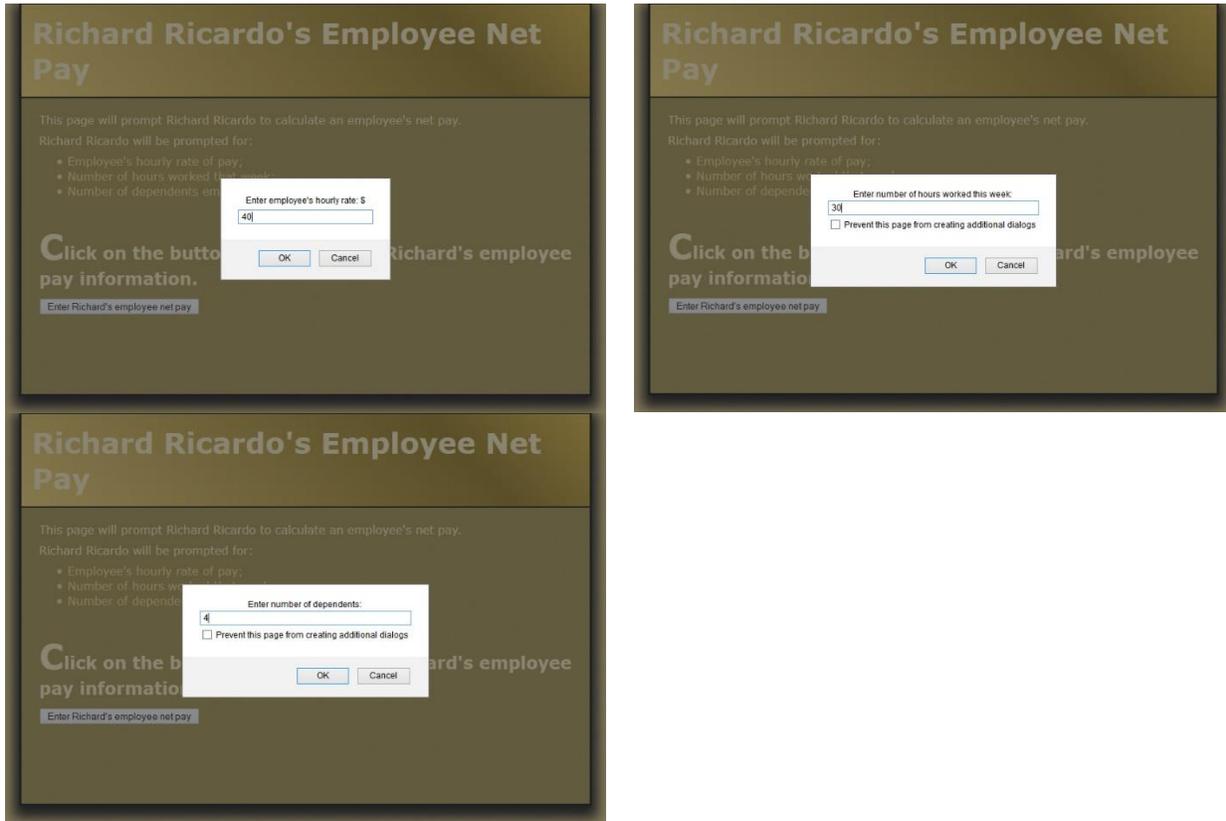
Example: Input q3a



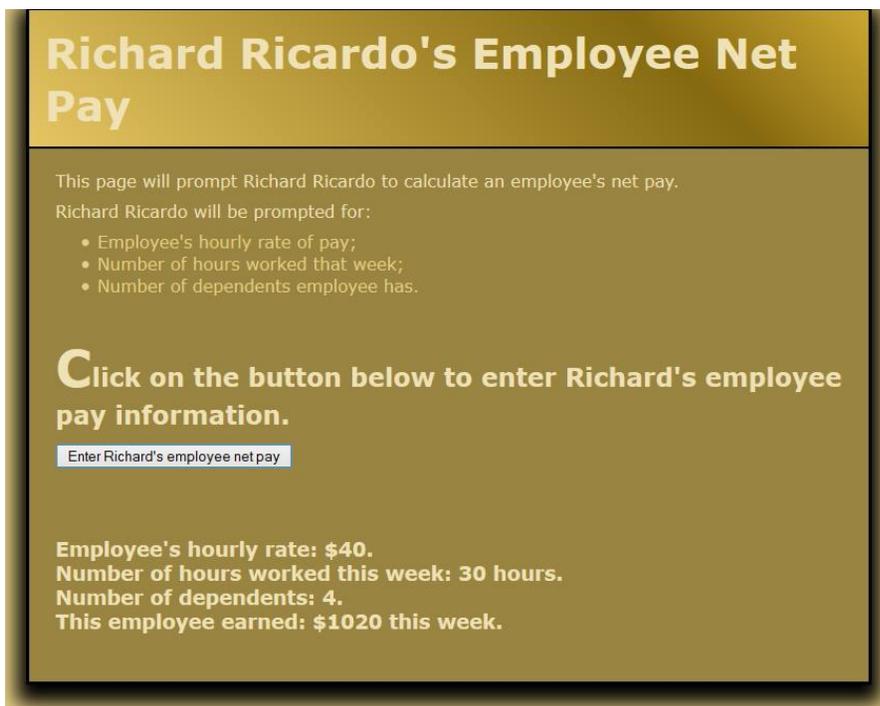
Example: Output q3a



Example: Input q3b



Example: Output q3b



**Important:**

1. If you do not put `<your name>` / `<your first name>` in the above mentioned fields (as shown in the examples), you will get **0 points** for the question(s).
2. **No two students** should submit webpages with exactly the same cod, or the content, or same layout, or same color combination. If found, both students will get **0 points**.
3. When you view page source in a web browser, `<!DOCTYPE html>` must be at the top of every page. In other words, all pages must be written in HTML5. (**-20 points** if not)
4. All html files must pass html validation at <http://validator.w3.org/> without any **error/warning** (with only 2 warnings). Use the validator's "File Upload" tab to check each file. (**-2 points** for each error/warning, **2 warnings allowed**)
5. All css files must pass css validation at <http://jigsaw.w3.org/css-validator/> without any **error/warning**. (**-2 points** for each error/warning)
6. If your html file contains any css component, your html file must pass both html validation (3 above), and css validation (4 above) without any error.
7. If your files do not pass the validations, **2 points will be deducted** for **each error** (and each JavaScript warning) found.
8. Document (comment) your html files (`<!-- -->`), css files (`/* */`), and JavaScript files (`/* */`). (**-1 point** for each file with insufficient comments)

**Submission instructions:**

- You need to test the above document(s) in your web browser.
- Do screen capture(s) of the **initial page** and the related **output(s)**. Use any graphic editing software (e.g. Microsoft Paint, Adobe Fireworks, GIMP, or Microsoft Expression Design etc) to cut out the browser output (from the screen capture), paste them into a word document.
- For this assignment, you only need to do screen capture(s) of the **initial page** and the related **output(s)**, you do not need to do screen capture(s) of the input pages. Provide **2 different test cases** for each question. In other words, for **each question**, you need to have **1 initial page** screen capture and **2 related output(s)** screen captures. (-50 points for no test cases, -5 points for only 1 test case)
- Do screen capture(s) of html validation results and css validation results, cut and paste them into the word document. You do not need to do screen capture(s) of JavaScript validation results.
- Save the word document as a pdf file.

You need to submit the following:

1. A pdf file containing the screen capture(s) of the web browser output (all html pages) and the screen capture(s) of all html validation results (from <http://validator.w3.org/>) and css validation results (from <http://jigsaw.w3.org/css-validator/>), name the file **lastname\_firstname\_assignment04.pdf**.
2. All html file(s), css file(s), and other related files (e.g. image files). Zip your file folder (lastname\_firstname\_assignment4) into a single zip file (or rar file) **lastname\_firstname\_assignment04.zip**. In the above example, the zip file should contain the following files and subfolders. If there is any image, there should be a `\images\` subfolder.

- lastname\_firstname\_assignment4\q1\index.htm
- lastname\_firstname\_assignment4\q1\style.css
- lastname\_firstname\_assignment4\q2\index.htm
- lastname\_firstname\_assignment4\q2\style.css // you may put style.css in a subfolder
- lastname\_firstname\_assignment4\q3\index.htm
- lastname\_firstname\_assignment4\q3\style.css

Please submit an electronic copy (the above mentioned **two files**: .pdf and .zip) to D2L digital dropbox.

**Grading guidelines (programming questions):**

Your programs will be judged on several criteria, which are shown below.

- Correctness (50%): Does the program compile (run) correctly? Does the program do what it's supposed to do?
- Design (20%): Are operations broken down in a reasonable way (e.g. classes and methods)?
- Style (10%): Is the program **indented** properly? Do variables have **meaningful names**?
- Robustness (10%): Does the program handle erroneous or unexpected input gracefully?
- Documentation (10%): Do all program files begin with a **comment** that identifies the author, the course code, and the program date? Are all the classes, methods and data fields clearly **documented (commented)**? Are unclear parts of code **documented (commented)**? (Some items mentioned may not apply to some languages)

A program that does not compile (run) will get at most **50% of the possible points**.