

# Exercises

1. **Compute BMI:** Write a function `computeBMI` that takes as parameters a weight in pounds and a height in inches and returns the Body Mass Index (BMI) for an individual with these characteristics. BMI is calculated from the equation:  $703 * \text{weight} / (\text{height} * \text{height})$ .
2. **BMI Calculator:** The following HTML snippet shows the skeleton of a BMI calculator. Write the `calculate()` function that takes the weight and height in the text boxes and displays the BMI in the span with the id of “score.” You may find the `computeBMI` function in the **Compute BMI** exercise helpful for this task. You should also describe the person’s body type based on their BMI score using the following criteria:
  - Underweight:  $< 18$
  - Normal:  $18 - 25$
  - Overweight:  $25 - 30$
  - Obese:  $> 30$

The person’s body type should be displayed in the span with the id of “type” and you may assume valid input into the text boxes.

```
<h1>Calculate your Body Mass Index (BMI)!</h1>

<fieldset>
  Weight (in pounds): <input type="text" id="weight" /><br />
  Height (in inches): <input type="text" id="height" /><br />
  <button onclick="calculate();">Get BMI!</button>
</fieldset>

<p>
  Your BMI is <span id="score"></span>. <br />
  Your body type is <span id="type"></span>.
</p>
```

3. **Flower Garden:** The following HTML snippet is from a webpage that allows users to design rectangular flower beds. When you click the “Plant Garden” button, the page should generate a rectangular grid of flower images in the div with the id of “garden.” The dimensions of grid are given by the values entered in the text boxes. Write the `grow()` JavaScript function to complete the functionality for the webpage. You should randomly display either “rose.jpg” or “daffodil.jpg” for each flower image, and either flower should show up with equal probability.

## 2 Exercises

```
<h1>Flower Garden</h1>

<div>
  Rows: <input type="text" id="rows" />
  Columns: <input type="text" id="columns" /> <br />
  <button onclick="grow();">Plant Garden</button>
</div>

<div id="garden"></div>
```

4. **The Amazing Weightinator:** The following HTML snippet shows the skeleton of a pounds-to-kilograms, kilograms-to-pounds conversion tool. Write the `convert()` function that takes the value in the text input and converts it from the unit selected in the left dropdown box to the unit selected on the right. The unit should be displayed in the empty span with the id of “answer.” The conversion factor from pounds to kilograms is 0.45359237, and the conversion factor from kilograms to pounds is 2.20462262. You should edit the HTML to add ids to the elements as necessary and you may assume valid input.

```
<fieldset>
  Convert <input type="text" />
  <select>
    <option>kg</option>
    <option>lb</option>
  </select>
  to
  <select>
    <option>lb</option>
    <option>kg</option>
  </select>
  <button onclick="convert();">Calculate</button>
  <span id="answer"></span>
</fieldset>
```

5. **Make Change:** The following HTML snippet contains the beginnings of a change calculator. Write the `calculate()` function that takes the value from the input field and computes a number of dollars, quarters, dimes, nickels and pennies in change such that all the change adds to the total entered. The higher coin values are preferred over lower coin values. (i.e. 1 quarter rather 2 dimes and 1 nickel). The message with the amount of change should be placed in the span with id “change.” 1. For example, if a person enters “3.19” as the input amount, the calculator should display “3 dollars, 0 quarters, 1 dimes, 1 nickels, 4 pennies” in the “change” span. You may assume valid input.

```
<h1>Calculate your change</h1>
<fieldset>
  Amount:
  <input type="text" id="amount" />
  <button onclick="calculate();">Calculate</button>
</fieldset>
<p>
  Change: <span id="change"></span>
</p>
```

6. **Coin Flip:** Write JavaScript code to randomly change the src of the img in the HTML snippet to "heads.jpg" or "tails.jpg" when the "Flip!" button is pressed. The image should change to the heads or tails picture with equal probability. Keep track of how many flips have resulted in heads and tails and report the results beneath the button in two spans with ids "num\_heads" and "num\_tails" respectively. You should write unobtrusive JavaScript code to attach the event handler to the button and you may assume the user enters valid input.

```
<h1>Heads or Tails?</h1>

<div>
   <br />
  <button id="flip">Flip!</button>
</div>

<p>
  Heads Total: <span id="num_heads">0</span><br />
  Tails Total: <span id="num_tails">0</span>
</p>
```

7. **Pizza Order Form:** The following HTML snippet represents a skeleton of a price calculator for a pizza order form. A small pizza is \$8.99, a medium pizza is \$10.99 and a large pizza is \$12.99. Tax is an additional 9% to the order. The user must also pay a tip from 10-20% of the total cost of the order after tax. Write the JavaScript code necessary to calculate and display the total cost of the order in the span with the id of "price" when the user clicks the "Calculate Order" button. You should write unobtrusive JavaScript code to attach the event handler to the button and you may assume the user enters valid input.

## 4 Exercises

```
<h1>Pizza Order Form</h1>

<fieldset>
  # of Small Pizzas <input type="text" id="small" /><br />
  # of Medium Pizzas <input type="text" id="medium" /><br />
  # of Large Pizzas <input type="text" id="large" /><br />

  Tip:
  <label><input type="radio" id="ten" name="tip" /> 10%</label>
  <label><input type="radio" id="fifteen" name="tip" checked="checked" /> 15%
  </label>
  <label><input type="radio" id="twenty" name="tip" /> 20%</label> <br />

  <button id="submit">Calculate Order</button>
</fieldset>

<p>
  Cost of order: <span id="price"></span>
</p>
```

- Bloom:** Write JavaScript code such that the size of a div element with an id of “rectangle” will increase each time a user clicks on it. Both the height and width of the div should increase by one fourth of its previous size on each click. For example, if the div initially has a width and height of 50px , its width and height increase to 62.5px after the first click, then 77.5px after the second click, and so on. Use unobtrusive JavaScript to add the event handlers to the div.
- Font Sizes:** Write JavaScript code to adjust font sizes of the text in paragraphs on a webpage. If a user clicks the element with the id of “smaller,” all paragraphs should decrease in font size by 2 pt. Similarly, if a user clicks the element with the id of “bigger,” all paragraphs should increase in font size by 2pt. Use unobtrusive JavaScript to add the appropriate event handlers. You should not have to edit the HTML in any way to complete this exercise.

```
<h1>About the Octopus</h1>

<div>
  <span id="smaller">a</span> | <span id="bigger">A</span>
</div>

<p>
  The octopus is an 8-legged sea creature. It inhabits many regions of the
  ocean.
</p>

<p>
  Contrary to popular belief, the plural of “octopus” can be octopi
  <em>or</em> “octopuses”; in fact, some argue that “octopuses” is the
  most correct terminology.
</p>
```

- Pig Latin:** Write a function called `atinLay()` to convert text from English into Pig Latin. To change text to Pig Latin, you take the first letter of each word to the end and add "ay" to it.

## Error! No text of specified style in document.

Then you move this syllable to the end of the word, preceded by a hyphen. The rule is only true for words beginning with a consonant; if the first letter of a word is a vowel (aeiou), then the word is unchanged. For example, the sentence "Kelly and Brent are neat" would be translated to "elly-Kay and rent-Bay are eato-nay" in Pig Latin.

You should also write code to attach the `atinLay` function as an `onclick` event handler to all paragraphs on a page so that when you click a paragraph, the text is converted to Pig Latin. You do not have to worry about the case if a user clicks on a paragraph multiple times.

- Essay Builder:** Write JavaScript code to append paragraphs of text to a div with the id of "essay\_body." A snippet of the HTML for this webpage is shown below:

```
<fieldset>
  Text to insert: <br />
  <textarea id="paragraph" rows="10" cols="50"></textarea>
  <button id="add">Add</button>
</fieldset>
<div id="essay_body"></div>
```

When a user clicks the "Add" button, your JavaScript code should create a new paragraph element with the text from the textarea, and the element should be appended to the "essay\_body" div. The text from the textarea should be also cleared immediately after a new paragraph is added.

- Rectangle Explosion:** Write JavaScript code to create a user-specified number of randomly placed, randomly sized, absolutely-positioned black rectangle divs when a user clicks the button on the following webpage. The rectangles' sides should be between 5px and 300px in length and the rectangle's locations should be constrained to a reasonable x and y, such as 800px in the x direction and 1200px in the y direction. You should use unobtrusive JavaScript to add this functionality to the page and you may assume valid user input.

```
<fieldset>
  Number Rectangles: <input type="text" id="number" />
  <button id="explode">Explode</button>
</fieldset>
```

- Reverse List:** Write a JavaScript function `reverseList()` that takes an id of an ordered or unordered list as a parameter and reverses the items in the list. You do not have to worry about reversing the items of nested lists.
- Mirror:** Write a JavaScript function `mirror()` that reverses the order of the characters in every heading on a webpage. (There are 6 HTML headings, h1 through h6.) Calling the function multiple times should toggle between the original text and the mirrored text.
- Chuck Norris Facts:** Create event handlers that will display Chuck Norris facts on a web page when a mouse moves over a Chuck Norris image and hide the fact when the mouse moves off the image. The Chuck Norris image has an id of "chuck" and the facts should be displayed in a paragraph with an id of "facts." You may use this array in your JavaScript for a source of facts:

## 6 Exercises

```
// taken from http://www.chucknorrisfacts.com/  
var $FACTS = ["Chuck Norris counted to infinity - twice.", "There is no  
'ctrl' button on Chuck Norris's computer. Chuck Norris is always in  
control.", "Chuck Norris destroyed the periodic table, because he only  
recognizes the element of surprise."];
```

16. **Picture Reveal:** Write JavaScript code to create an interactive picture-revealer tool. Initially the page appears black. When the user mouses over a black area, the webpage reveals the background image.



The body of the HTML page is solely a one-line div with an id of “container.”

```
<body>  
  <div id="container"></div>  
</body>
```

The CSS for the container div gives the div a background image, which is 250 x 250 px in dimensions.

```
#container {  
  background-image:url("starrynight.jpg");  
  height: 250px;  
  width: 250px;  
  position: relative;  
}
```

When the page loads, your JavaScript should generate a 10 x 10 grid of 25 x 25px divs which are children of the container div. This grid of divs should be positioned such that there are no gaps between adjacent divs and they cover the entire background image. You should add the CSS class "block" to each div, which gives the div dimensions, absolute positioning and color.

```
.block {  
  height: 50px;  
  width: 50px;  
  background: black;  
  position: absolute;  
}
```

When the user mouses over a div, the small back divs should *hide behind* the container div. This can be accomplished by changing the z index of the div.

17. **Slideshow:** Write JavaScript code to make a slideshow by rotating through different images every 30 seconds. You may assume you have five images in the same directory as your JavaScript file, each named “image1.jpg” through “image5.jpg.” An HTML snippet from this webpage is shown below. You should not have to modify the HTML to complete this exercise.

```
<h1>Slideshow</h1>  
<div id="viewer">  
    
</div>
```

- 18.