# Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ StudentID: \_\_s3836387

You can test regular expressions online: <https://regexr.com/> or <http://regex101.com/>

or print a cheat sheet: <https://www.cheatography.com/davechild/cheat-sheets/regular-expressions/>

## Activity 1: Regular Expressions

Regular expressions can be used in input pattern attributes and javascript to detect a string's content and format. Explain what the following regex looks for, in what format, and how it works:

^(\(04\)|04|\+614)[ ]?\d{4}[ ]?\d{4}$

Ans: The following regex look for a number that start with ‘04’ or ‘(04)’ or +614.

It might be Australia number with optional spaces between group of 4 digit.

Can you write an improved version? One that allows for more flexibility?

Ans: The number 0400 123 456 will fail

Improve version: ^(\(04\)|04|\+614)( ?\d){8}$

Write a regex that matches characters found in a (Western) name field. Make sure you can match these names?

* <https://en.wikipedia.org/wiki/Gerard_%27t_Hooft>
* <http://www.imdb.com/name/nm0001876>
* <http://www.imdb.com/name/nm0000375>

Ans: ^[A-Za-z \-.']{1,100}$

|  |
| --- |
| Once these tests are complete, use the working regexes in your assignment. |

## Activity 2: Javascript Code Generation

What type of HTML element is the following javascript trying to create? Draw a picture below the code box.

|  |
| --- |
| 1. <p>Birth Year: 2. <select name='dob'> 3. <script> 4. var year = new Date().getFullYear(); 5. for ( var i=year; i<=(year-50); i-- ) 6. document.write('<option value="'+i+'">'+i+'</option>'); 7. </script> 8. </select> 9. </p> |

Ans: A select input box with 50 year listed in reverse order

Which line is the "deliberate mistake" on? Fix the error so that the code will run correctly.

Ans: The condition to stop the for-loop line is wrong making it loop forever.

for ( var i=year; i >= (year-50); i-- )

Think of a practical use for the corrected code, and rewrite line 5 so that no one alive is discriminated against.

Ans: This code might be for user that are under 50. The new version:

for ( var i=year; i >= (year-100); i-- )

How might you adapt the "practical use" code so that only those that are 18 and older some time this year qualify?

Ans: Some website that restricted the people that under 18 from register or shop. Like alcohol, vape, etc.

for ( var i= (year - 18) ; i>=(year-150); i-- )

## Activity 3: Javascript Arrays: Indexed and Associative

An array is a sequential list. The simplest is a **numerically indexed array**, **indexed** by **integers** starting at **0.** To initialise a simple 1D indexed array in javascript with one statement, we can use the following syntax:

|  |
| --- |
| var numbers = [ 'zero', 'one', 'two', 'three', 'four' ]; |

What does numbers[2] return?

Ans: two

Another way of looking at this array is that each index is a key pointing to a value(eg in the above example, the 2 in numbers[2] is a key pointing to a value). To initialise a simple 1D **associative array** in javascript with one statement, we can use the following syntax:

|  |
| --- |
| var letters = { zero:'A', one:'B', two:'C', three:'D', four:'E' }; |

What does letters['two'] return? What about letters.two?

Ans: C

C

A programmer will usually want to iterate throughout the whole array. The for-in loop simplifies the process of iterating through an array. Compare the standard for loop syntax with the for-in loop syntax:

|  |
| --- |
| // Using a for loop  for ( var key=0; key<numbers.length; key++ ) {  document.write( key + ': ' + numbers[key] + "<br>" );  }  // Using a for-in loop  for ( var key in letters ) {  document.write( key + ': ' + letters[key] + "<br>" );  } |

Ans: the for loop iterate through the array by using the number index of the array

The for in loop iterate through the array by using the array key

## 

## Activity 4: Web Storage: localStorage and sessionStorage

localStorage and sessionStorage are two new HTML5 data structures (ie variables) that store strings into associative arrays on the client's machine. Data takes the form of keys and values in both structures.

**Handy Tip:** Before using localStorage and sessionStorage make sure that the browser supports it by performing a typeof check. The example below shows localStorage, but the syntax is the same for sessionStorage.

|  |
| --- |
| if (typeof(Storage) !== "undefined") {    // There are three ways to set a storage variable  localStorage.setItem('key','value'); // or ...  localStorage['key'] = 'value'; // or ..  localStorage.key = 'value';  // Many ways NOT to set a storage variable  localStorage.setItem('key', 5); // fails, 5 is not a string  localStorage.setItem('key', true); // fails, true is not a string  localStorage.setItem('key', [1,2,3]); // fails, an array is not a string  // and one way to unset a storage variable  localStorage.removeItem('key');  } |

What is the difference between localStorage and sessionStorage?

Ans: LocalStorage stay on the client machine until deleted

SessionStorage is deleted when the user close browser

Why is using web storage better than using cookies, also stored client side?

Ans: Because security issue about using cookies