**NAME:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ DATE:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

***Note:*** This assignment builds understanding of following topics in JavaScript:

1. Topic 1
2. Topic 2
3. Create a function adder that accepts a number and returns another function that accepts a number too. The inner function (one returned) adds the number passed to adder and the number passed to itself and logs the result. You should be able to invoke your code this way

var adder10 = adder( 10 );

adder( 20 ); // logs 30

adder( 30 ); // logs 40

var adder 50 = adder( 50 );

adder50( 20 ); // logs 70

adder50( 30 ); // logs 80

1. Define an object to hold the details of an employee. Choose appropriate property names and data types for the properties.
   1. Name
   2. Location
   3. Email id
   4. Phone number
   5. Project details (of multiple projects). Each project detail will have
      1. Project name
      2. Project duration
      3. Hours of effort per week
      4. Billing rate

Next perform the following tasks.

* Print the data type of the object and each of the properties.
* Print the value of email id. Now change the value of email id and print again.
* Print the duration of the first project. Now change the value of duration and print it again.
* Remove the location property and print the employee object.

1. To the above object, add a method (during object creation time) that accepts the name of the project and returns an object with the project details. If there is no matching project, it should return null. Call the method and print the returned project details.
2. Define a constructor function Employee that accepts appropriate arguments and creates object as defined in exercise 2. Employee objects should have the method to get project details when passed the project name. Create 2 Employee object and search for project details.
3. Create a function that adds the property x and y defined on its context, and returns it. Make sure to use call to change the context of the call to the function to an object like { x : 10, y : 20 }