

## 2\_OOP-1\_Introduction of Object Oriented Programming ASYNC (2 Hours)

<b>Learning Objectives</b>	<ul style="list-style-type: none"><li>• What is Object Oriented Programming</li><li>• How to think in Object Oriented Way</li><li>• What are Classes and Objects</li><li>• Principles of Object Oriented</li><li>• Encapsulation</li></ul>
----------------------------	--

Time	Activity
(3 min)	<b>OPENING</b>
(1 hours 45 min)	<b>CONTENT</b>
(10 min)	<b>DEBRIEF</b>
(2 min)	<b>CLOSING</b>

### OPENING (3 min)

<b>Activity Type</b>	Reading
<b>Framing Text</b>	<p>Welcome to this session in which you be able to work on some advance functions</p> <p>In this section you will be able to work on programming question that will help you to understand JavaScript programming well. You have to complete all the questions</p> <p>By the end of this session, you will be able to:</p> <ul style="list-style-type: none"><li>• Make classes</li><li>• Makes Object</li><li>• Use Encapsulation.</li></ul>
<b>Assets</b>	<ul style="list-style-type: none"><li>• n/a</li></ul>

## 2\_OOP-1\_Introduction of Object Oriented Programming ASYNC (2 Hours)

### CONTENT (1 hours 10 min)

Activity Type	Coding
Framing Text	<p>This will take you approximately <b>2 hours</b> to complete.</p> <p>After completing it, you will return to code files by zipping in the folders for a debrief.</p> <p><b>Complete:</b></p> <p>Happy coding!</p> <div><p><b>Note:</b> Please reach out to instructors asking for help if you get stuck with understanding of questions</p></div>
Assets	<p><b><u>Task 1:</u></b></p> <ul style="list-style-type: none"><li>• In this task we provide you with the start of a definition for a Shape class. It has three properties: name, sides, and side Length. This class only models shape for which all sides are the same length, like a square or an equilateral triangle.</li><li>• Add a constructor to this class. The constructor takes arguments for the name, sides, and sideLength properties, and initializes them.</li><li>• Add a new method calcPerimeter() method to the class, which calculates its perimeter (the length of the shape's outer edge) and logs the result to the console.</li><li>• Create a new instance of the Shape class called square. Give it a name of square and a sideLength of 5.</li><li>• Call your calcPerimeter() method on the instance, to see whether it logs the calculation result to the browser's console as expected.</li><li>• Create a new instance of Shape called triangle, with a name of triangle and a sideLength of 3.</li><li>• Call triangle.calcPerimeter() to check that it works OK.</li></ul> <p><b><u>Task 2:</u></b></p> <ul style="list-style-type: none"><li>• Implement a class Player having PlayerID, PlayerName, PlayerRuns, PlayerDateofBirth, PlayerShirtNumber.</li><li>• It should also have methods DisplayInfo(), GetRuns(), AddRuns() that prints the splayers info, total runs of the player and can add runs.</li><li>• You should also create a class variable that stores total runs of all the players and there should be a</li></ul>

## 2\_OOP-1\_Introduction of Object Oriented Programming ASYNC (2 Hours)

- method that displays the class variable

### **Task 3:**

- Write a JavaScript program to display the reading status (i.e. display book name, author name and reading status) of the following books

```
var library = [  
  {  
    author: 'Bill Gates',  
    title: 'The Road Ahead',  
    readingStatus: true  
  },  
  {  
    author: 'Steve Jobs',  
    title: 'Walter Isaacson',  
    readingStatus: true  
  },  
  {  
    author: 'Suzanne Collins',  
    title: 'Mockingjay: The Final Book of The Hunger Games',  
    readingStatus: false  
  }  
];
```

### **Task 4:**

- Make a Bank Account class with attributes accountId , accountTitle, balance, emailID. Make each variable private. Initialize all the attributes and make them private. Make setters(mutators) and getters(accessors) for all the variables. Now make a method withdrawAmount(amount) and depositAmount(amount). Each will have a parameter of some amount. For withdrawAmount() display user the amount that has been withdrawn and remaining balance . Make sure that amount that is being withdrawn is not greater than the balance. For depositAmount() add the amount and show the current balance.

-

## 2\_OOP-1\_Introduction of Object Oriented Programming ASYNC (2 Hours)

**DEBRIEF (15 min)**

**Activity Type:**

Discussion Forum