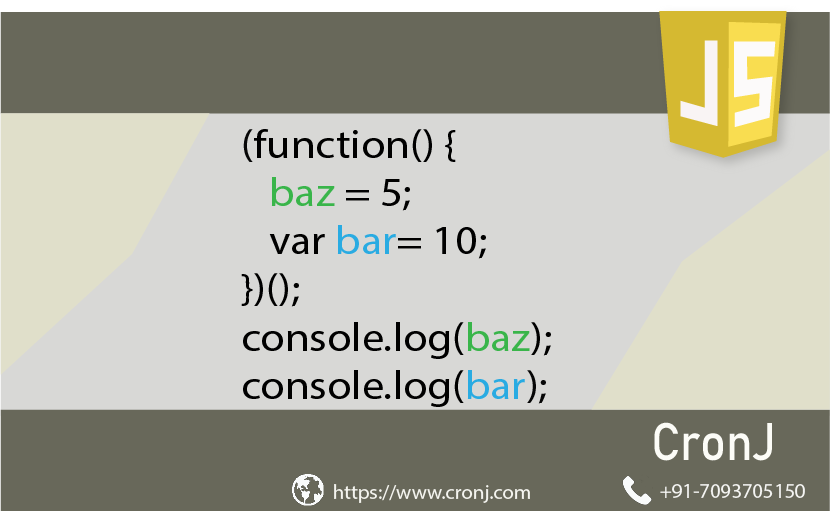
**Day 1 Lab Assignments**

1. Handle onerror event (Or any other event) on the page using Declarative function.
2. Change the function from Declarative to literal function (Function expression), without assigning the function to a variable.
3. What’s the difference?
4. Which type of functions is the best suited for handling the event? And why?
5. Make an Interval that runs each second, and executes logTest function that writes the current time each time it called in the console, and handle the function to be literal function to be passed to the first parameter to setInterval (Don’t define a function and then pass its name, define the function literally in the first parameter of setInterval()).
6. Why literal function is better than defining a declarative function, and then passes its name in that case?
7. Make a script in a body, and ask the user to enter 2 numbers, and then make a Dynamic function to take 2 numbers as parameters and alert the sum of 2 numbers and execute the function.
8. Re-declare the function (assign it to the same variable, without deleting the prev. function) to alert the subtract of the 2 numbers, and execute it.
9. Re-declare it to alert the multiply of the 2 numbers and execute it.
10. Re-declare it to alert the division of the 2 numbers and execute it.
11. Why the Dynamic function is the best type for such this case?
12. Make an IIFE function expression (Self-invoking function) that executes automatically when the page loads, and initialize some global variable in the page with zero value (Without making an name to the function, or assigning it to a variable, or calling it in onload() event).
13. Use JS constructor function to create the following custom types:
14. Create custom type “Person” that has Name and age properties.
15. Use prototype property to add PrintPerson() method to the class.
16. Create student object that inherits from Person, and its constructor have two member variables (Faculty and Total\_mark), and put default value for Total\_mark to be 0 (Using Function Constructor).
17. Use prototype property to add PrintStudent() method to the class.
18. In previous class Create private variable named (min\_degree) which value is 60, and create local (inner) function named (checkSuccess) that return “Succeed” if student degree is more than min\_degree, else it returns “Fail”.
19. Make another member function (public) named (checkResult) that alert the result of that student.
20. Make an Object of student and set it members, and test it with different values. (Try to access min\_degree variable, and checkSuccess function outside class, what did you remark?).
21. What’s the difference between adding property using prototype, and adding it to specific class object?
22. Create 2 Course objects (Using literal method) crs1, crs2 that has courseName, courseMaxDegree properties, and printCourse() method that print the course in this format "Max. Degree of JS Course is 100".
23. Add new property courseHours to crs1=30, and try to alert it from crs1 and crs2.
24. Solve this quiz!  
    

**Bonus:**

1. Implement Stack custom type using object oriented in JS (pop and push member functions, and currentSize property)
2. Redo the course custom object in previous assignment, using function factory method.
3. Write a function that takes any number of parameters and returns an array containing them (use arguments property of the function object).
4. Display the area and perimeter of an object created from using Rectangle Constructor that has width and height properties and 2 extended methods for calculating area, perimeter .
   1. Override .toString() function to display a message declaring the width , height, area and perimeter of the created object.
   2. Create Class Property that counts numbers of created objects and Class method to retrieve it.
   3. Make proper updates in your code to make Rectangle inherits from Shape Constructor, and Square inherits from Rectangle.
   4. Make sure that any created object from Rectangle is instance of both Rectangle and Shape, while any created object from Square is instance of Square, Rectangle and Shape.
5. Implement linked list using OO JS [create custom object node, and custom object linkedList, to simulate linked list].
6. Make your own mathematical constructor that add, and multiply values of an array, based upon passing required mathematical operation. Let this constructor takes 3 parameters: 1st represents array 2nd parameter represents initiation of the collecting storage (i.e.1 for multiply, 0 for adding) 3rd parameter represents the required operator. Let this function contains an inner function that uses switch..case for handling different operations. Note: This constructor may have an IIFE if needed.

**<Script>document.write(“Thank YOU”) </Script>**