Function

**Anonymous function**: Function without a name is called anonymous function. It is used when the functions are used as values.

**First class function:**

**Function statement aka function declaration:**

Function p(){

Console.log(“Hi”)

}

**Function Expression**: When we use the function as value and assign it within a variable that is called function expression.

Var b= function (){

Console.log(“Hello”)

}

**Difference b/w function statement and function expression :** main different is hoisting, if we try to call the function b from the begging before it was initialised it will give an error as because the memory space is not allocate for function k until it reaches to the function line**.**

**Named function expression:** when we assign a value to a function expression that is called named function expression.

Var b= function K(){

Console.log(“Hello”)

}

K();

Now, if we try to access k outside of the block it will give us error as value of k is not accessible outside of that block. And it works as a local variable.

**Difference b/w parameters and arguments:**

The values which we pass through a function is called parameters and the identifiers/labels that gets those passed function values are called parameters.

function add(x,y) - **parameters**

{

Return x+y;

}

add(5,6) ; – **arguments**

**First class function aka citizens**: the ability to use function as value and pass the function as argument inside another function and return a function inside a function is called first class function.

function add(x,y) - **parameters**

{

Return x+y;

}

add(function p(), function q()) ; – **arguments**

function p()

{}

Function q(){}

**Callback functions:** The function which you can pass through another function is known as callback function.

function x(y){}

x(function y(){}) ; - function y is called as callback function.

**setTimeout()** – this function use to hold the function for a duration of time in ms. The first parameter of the setTimeout function is called callback function.

Everything in JS is happened through callstack and JS has only one callstack. setTimeout function makes the Js asynchronous as Js doesn’t wait for the callback function to perform that particular time, instead of this it moves to the next line of code.

**Higher order function:**