***Java script notes***

* **Let, var,const:-**
* In java script most of time we use let for making a variables.

Eg:- let a=10;

Let b=10.125

Let c= ‘ a’

* In let we can update/change the variable by writing a=55,b=20.5,c=’r’.
* Var is not use by developer after the new update in java script in 2015.
* In var we can’t update a variable but we can create new variable of same name .

Eg:- var a=10;

Var a=50;

* Const is use if we want to make a varible which can not update and change in whole programe.
* It stay constant in whole program.
* ***Oparaters in java script:-***
* Ther is a one new oparater in java script which is exponentiation(\*\*) oparater.

Eg:- let a=5;

Let b=2;

Console.log(“ a^b/ 5^2=”,a\*\*b); // 5^2

* ***Objactive in java script:-***
* Yeha hm ek varible bnayenge jo ek objective hoga. Ish varible ke ander aleg aleg data type ke varible honge.ye aleg aleeg data type ke varible {} block ke ander honge or ye block ek varible ke name se pehchana jayege.

Eg:- let student ={

name: ”Dhruv sharma ”,

roll: 126,

phone: 9512456895,

};

* Now to access it we use 2 method

1. Console.log(student[“roll”]);
2. Console.log(student.roll);

* ***Note***

Let :- can be update.But in update it create the new variable.

Var a=10;

Var a=55;

Var:- can not update. It replace the value with old variable.

Let a=10;

a=110;

Const:- can not update, it remain constant in whole programe.

* ***For of loop and for in loop:-***
* For of loop is use for string or array.

Let str=” Dhruv”

Eg:- for(let I of str(string name)){

Console.log(I );

}

Output:- D

h

r

u

v

interators -> characters ( I -> character of strings , one by one)

* For of loop is use for objects.

Let object={

Item:” pen”,

Price: 10

};

For( let I in object(object name)){

Console.log( “ key name = ”,I ,” key value= ”,object[i]);

}

Output:- key name= item key value= pen

Key name= price key value=10

(I -> key name ).

Note:- we can not change the string but to change it we can create a new string or replace it. (string are immortal)

* ***String:-***

1. Create string:- let str=” Dhruv sharma”
2. String length:- str(string name)length
3. String index:- string index use to print/access the character of string.

Str[0],str[1],str[3]….

* ***Template literals in js:-***

A way to have embedded/join(value of variable / jo value hme print lerni hoti he ) epression in strings.

` this ls a template literals`

String interpolation

Create a string by doing a substitution of placeholder/expression

` this Is ${ expression} pen` -> the ${expression} is substitute during run time.

Example was written in programme.

* String methos:-

These are built- in function to manipulate a string.

* Str.toUpperCase();
* Str.toLowercase();
* Str.trim() ->Remove the white space frome start or end.( Eg:- “ Dhriuv “).
* Str.slice(start,end?) ->(Eg:- “stringdh” str.slice(1,4) output:- tri the last character is not included in output).
* Str1.concat(str2) -> join str2 with str1 (we can also use +)
* Str.replace(searchval,newval) -> Eg: “dhruv” str.replace(“ru” , ”M”)
* Str.replaceAll( );
* Str.charAt();
* Str.length();

Note :- here the method return new string so we can store it in new veriable. It don’t make changes in original string.

Eg:- Let new = str.toUppercase();

* ***Array:-***
* It is a collection of items .
* String array : let name=[ ”dhruv”,” rohit”,”rahul”];
* Number array: let roll=[10,20,50,10,90];
* Let info=[“rahul”,86,”delhi”]; -> Not use generally.
* Array index:- Eg- let no=[5,2,6,4,21,25];

Here, index value

1. 5
2. 2
3. 6
4. 4
5. 21
6. 25

Note:- The array are mutable. Means we can change it’s value.

Eg:- no[0]=55; here 5 -> to 55. In original array.

* Arr.length() -> it is property we use to find length of array and string.
* For of loop in array:-

Eg:- let roll=[20,10,50,30,48,02,04];

For(let el of roll){ Note:- here el only get copy of array .

Console.log(el);

}

* Array Methods :-
* Push() -> add new item/element at the end of array.(after last element/item new item will added)

It make change in original array.

* Pop() -> delete item/element from end of array(last element/item). It return the deleted item which we can store it in new variable.
* toString() -> it is use to display the array into string. It not make chages in original array.
* Concat():- to add multiple array.it return the result which we can store it in new variable. Eg:- arr1.concat[arr2];
* Unshift():- add new item from start.change original array.
* Shift():- delet item from start and return deleted item.
* Slice():- return piece of array which we want to slice.

Eg:- arr.slice( index(2),index(5));

Note:- it not include the last index when it slice the array.

* Splice(): change original array (add,remove,replace)

Arr.splice(start index, no.of item you want to delet, newitem,newitem,….);

1.delet no. of item from starting index and add new item also from starting index.

2.if we only write start index then it delet all the item from start index to end index.

* ***Function:-***

Eg:- function add(){

// code

}

* Arrow function:-

Eg:- let add=(let a,let b)=>{ -> here add is a function variable.

Return a+b;

}

//call function

add(a,b);

* Arr.forEach(callBackFunction);
* callBackFunction:It Is function which is execute for each element in arry.
* A callback is a functon is functon , which we want to execute/perform for every element in array.

Eg:- let arry=[25,2,54,86,5];

Arry.forEach( (value, index, arry)=>{

Console.log(value –index-- arry);

})

First argument is for element of array.]

Second is for index

And third for full arry.

* Hof /Hom( higher order function/method):- method/function which take function as parameter or return a function is know as higher order function/method.
* ***Map method*** :- same as a forEach method but here map return new value/array which we can store in a other array.
* It creates a new array with the result of same operation

The value it’s callback return is used to form new array.

Ler newarr.map((val,index,arr)=>{

//code

})

* ***Filter:-***

Creates a new array of element that gives true for a condition/filter.

Eg:- let newarr=arr.filter((val)=>{

Return val%2===0;

}) -> it creates arr of even number.

* ***Reduce:-***

Perform some operation on array and reduce the array into a single value .

It returns that single value.

Let newval=arr.reduce( (perivious , current)=>{

//code

})

Array=[5,2,1,4];

* Process:- previous=5,current=2 -> previous=5+2=7

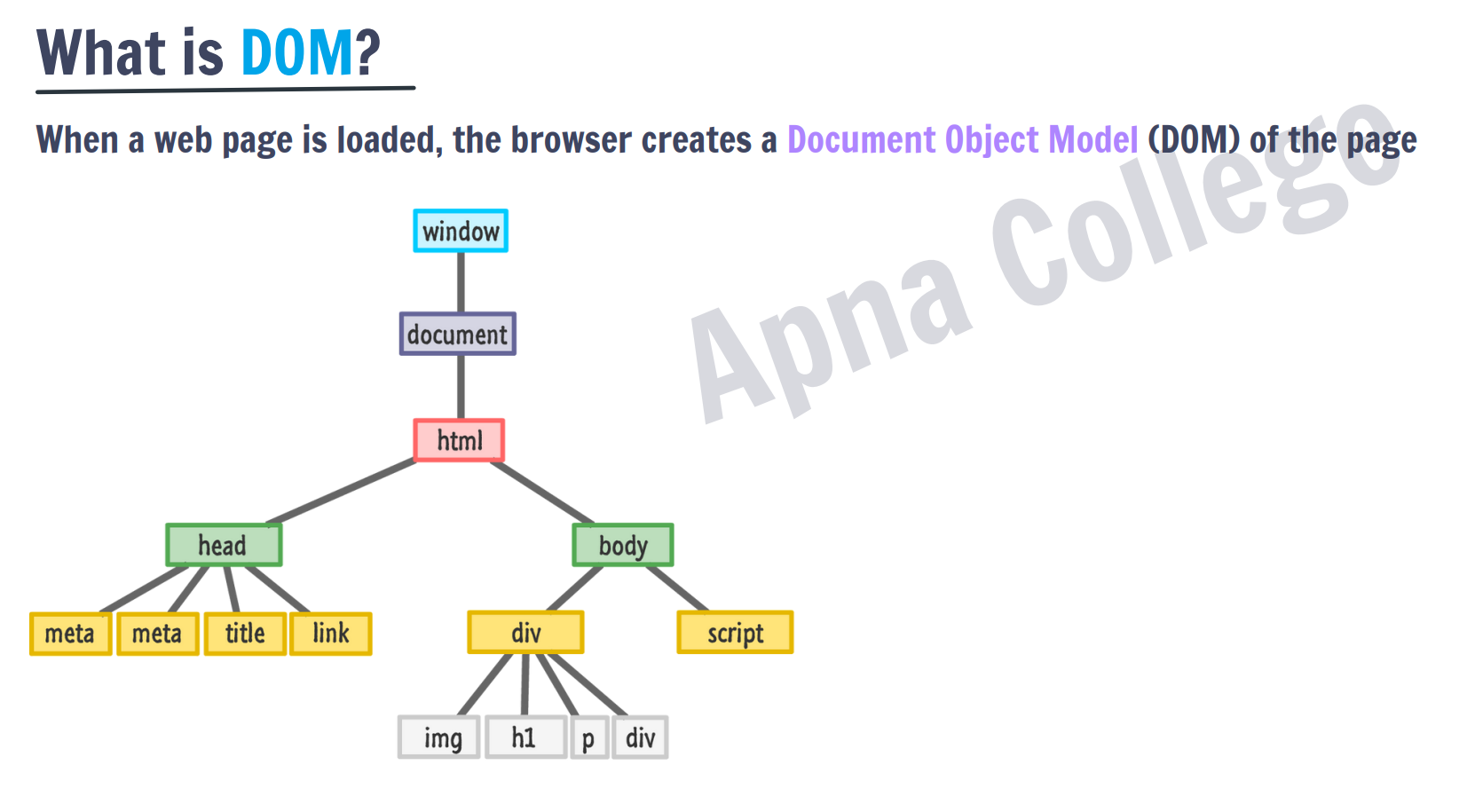
previous=7,current=1 -> previous=7+1=8

previous=8,current=4 -> previous=8+4=12

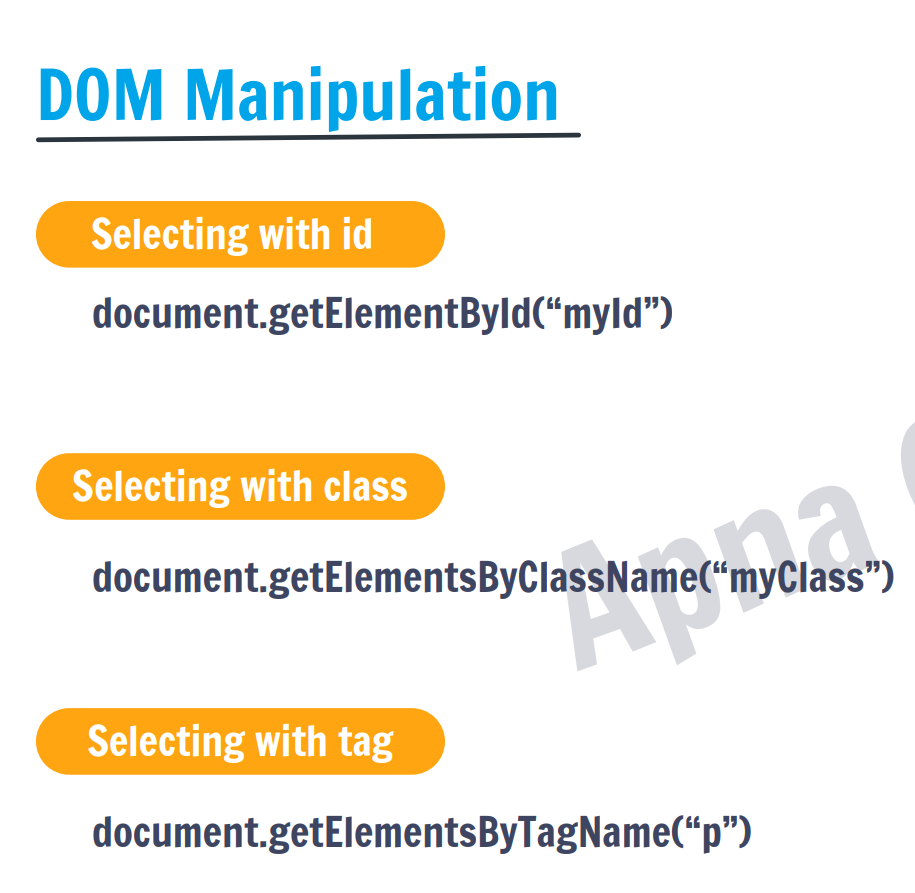
final single value is 12.

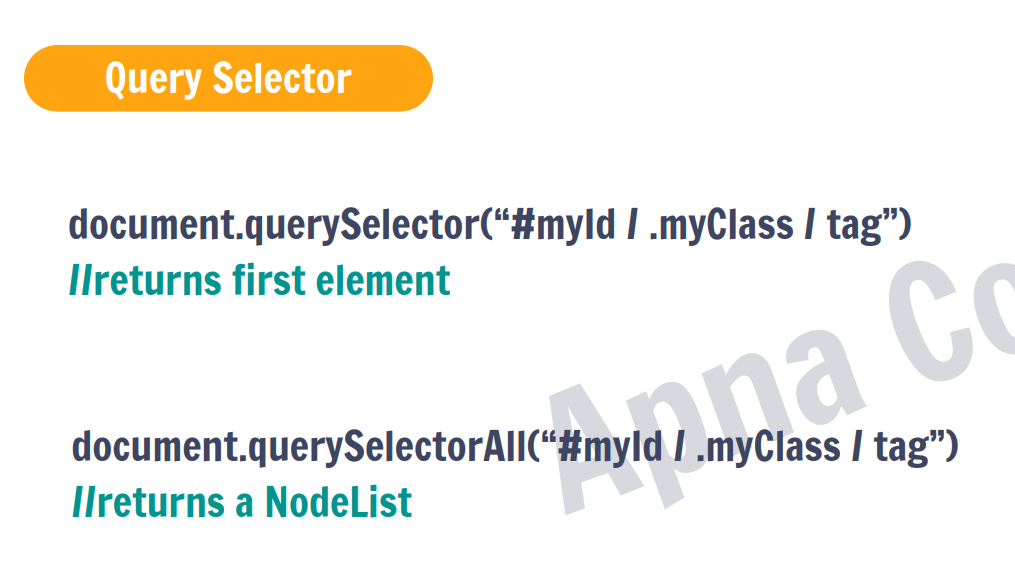
* ***DOM(Document object modal):-***
* ***Window Object:-***
* The window object represents an open window in a browser. It is browser’s object (not JavaScript’s) & is automatically created by browser.
* It is a global object with lots of properties & methods.

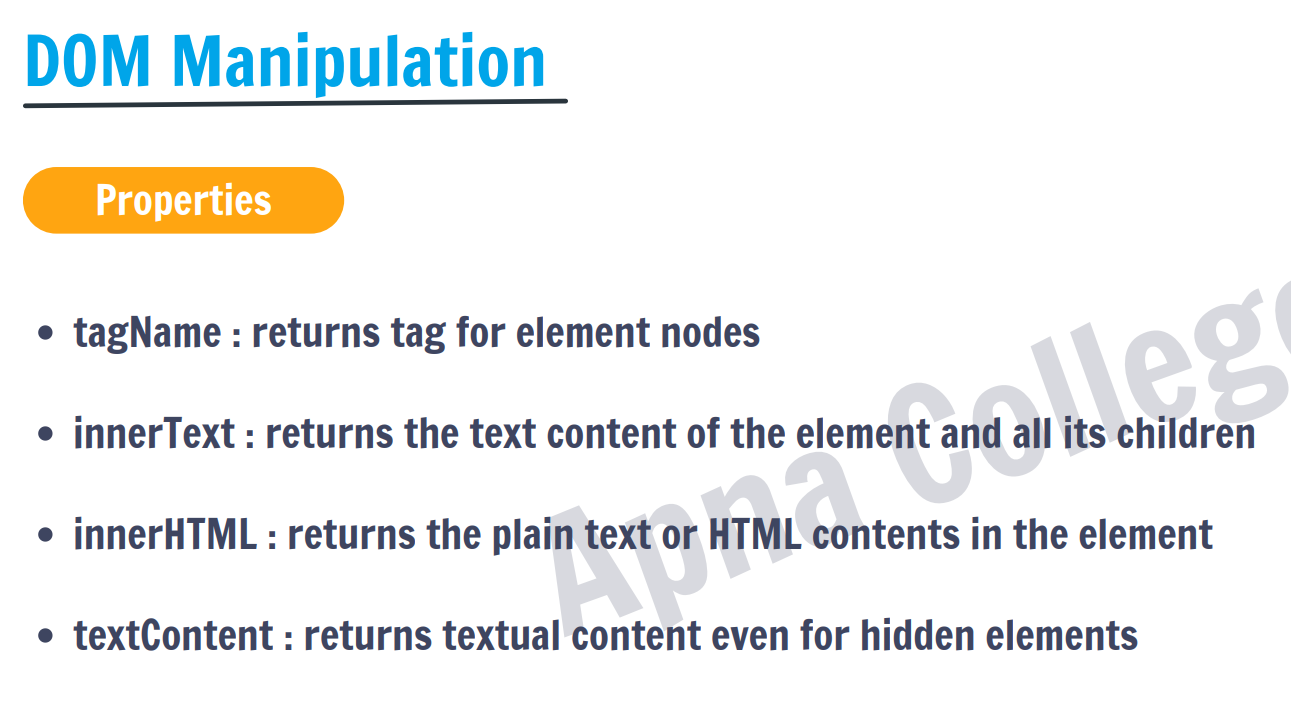
* Browser already know about it.

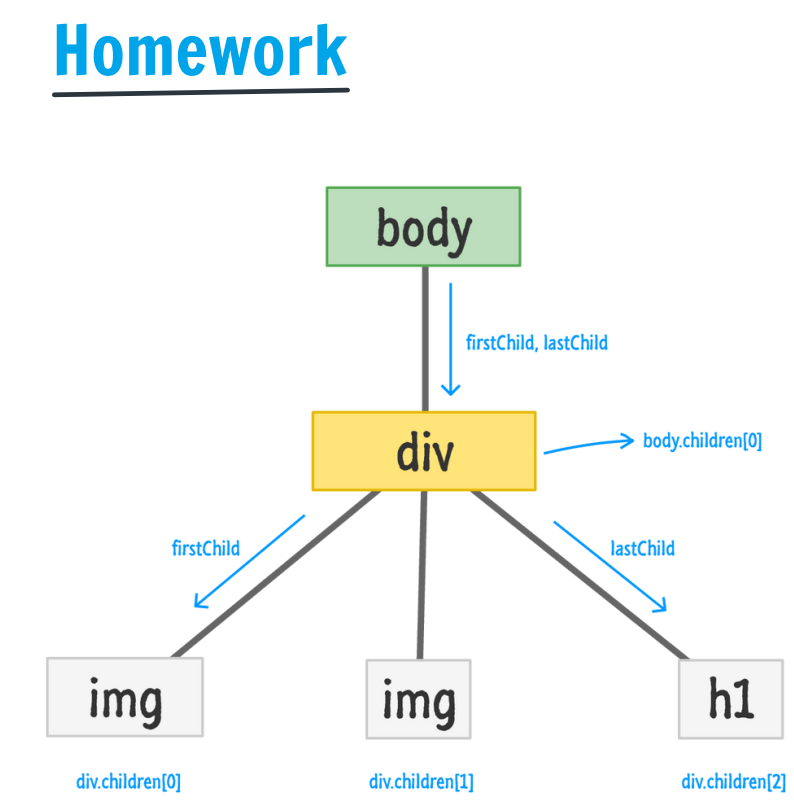


* ***DOM Manipulation:-***



******





* Access and change element by using the (getAttribute(attri) and setAttribute(attri,new value):-
* getAttribute( attr ) //to get the attribute value:-

First access the element like div , after this use elemnt . getAttribute(“element”).

Eg:- let div = document.quarrySelecter(“ div”);

Let newdiv=getquarrySelecter(“div”);

* setAttribute( attr, value ) //to set the attribute value:-

first access the element like p, after this use element and use setAttribute(“attri”, ”new value”);

EG:- let p= document.quarryselecter(“p”);

Let newp= setAttribute( “id” , ”Timepass”);

* Id value change to Timepass.
* Node.style:-

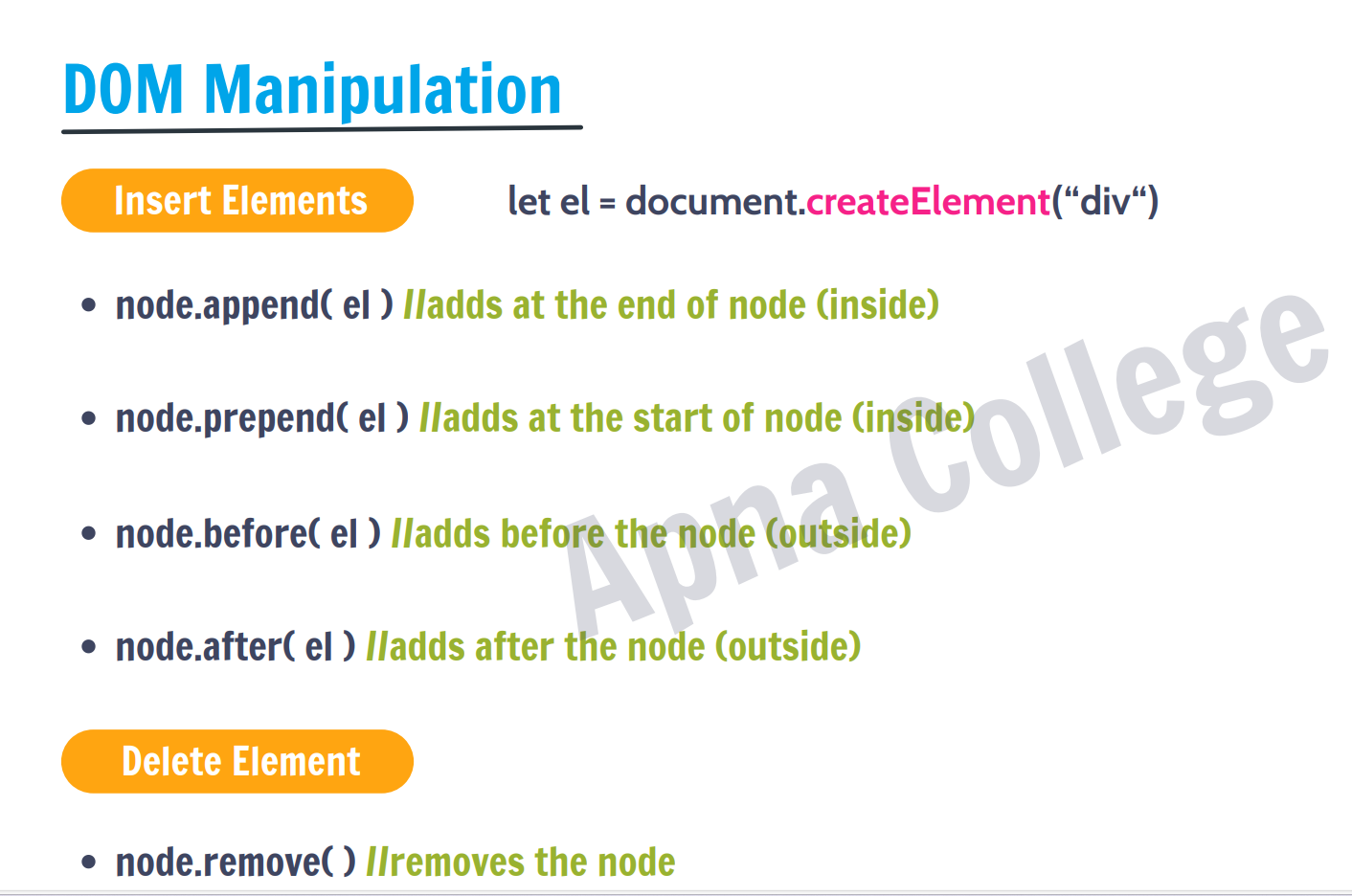
First access the element to change it ,

Eg:- element(p).style.fontSize=”20px”

We can change any attrbutes use in css by using this format.

Font-size -> fontSize.

* Methods which is use to add and remove the elements in website:-



* Here the node means the element.
* There are also other methods like appendchild()

And prependchild().

* Classlist :-

Jab ek element ko dono ki ability chahiye tab hmm classlist use kerte he.

* String method:-

Name\_of\_string.Charcodeat(index-number);

* Method to convert string

1. parseInt(“125”); => ans will come 125 in intiger.

Remember one thing you do’t get a ascii value by using it.

parseInt(“abc”)=>it give nan

1. stringname.charcodeat(index NO.);=> it give ascii value of It.

* Classes And object:-
* Object => it is a entity/structure of properties and method/function.

Direct declaring object.

Const student={ =>student is our object.

Name:”dhruv”; => object property

Roll:126, => object property

printRoll: function () { => printRoll is a function/method

console.log(“The roll is”, this.printRoll);

}

this=> yeha this ka matlab he the object name jiske ander ye bnan he), we can’t directly access the property.

* Prototype ( \_\_proto\_\_):-

Every object has a spacfial methos which a called prototype.

Const Student={

Name: name()=>{}; => method

Const employee {

Salary:

}

Employee\_\_proto\_\_ =student; => employee object get the methods of students in it’s prototype.

}

* If there is a same method in prototype and a current object then the current method will execute not a prototype method.
* Class:-

It is like a templete on the basis of that templete the object are made which has that templete property and method.

We can make a many object and give them the method of the class by declare loike this

Let object= new class();

Here class automatically create a constrater method but we can also custom that.

What ever we write in constructer() that work will done at the inisilation of object Eg:-

Class {

Constracter(){

name(hello){ this.name=hello}  
}

}

Let add=new class(“Dhruv sharma”);

Here first we have to give name at the time of inizialisation of object.

* Inheritances:-

Inheritance means passing down properties and methods from parents to child class.

Eg:-

Class student {

Name(name){ this.name=name}

}

Class employee extends students{

Salary (sal){ this.salary=sal}

}

If child and parent has a same method then the child method will used.

* Super():-

In inheritance when child class invoke it’s constracter then we get error

To solve it we have to call our parent constracter fist then the child constracter.

So, for doing this we normaly use super()function.

Eg:- class student{

Constracter(){

Console.log(“hello”);

}

}

Class clgstudent extends student{

Super(); => using super we first call parent constracter.

Constracter(){

Console.log(“hii”);

}

}

* Super(value which we want to pass to the parent constracter);
* Super. method name which is made in parent class();

We use this to invoke parent class method.

* Error handling:-

Try{

// write the code jo tumhe lege error dega

}catch(err){

Console.log(err);

}

=>by using this ,we can run the rest of the code if the code inside the try block has error.

* setTimeout() and setIntervel():-

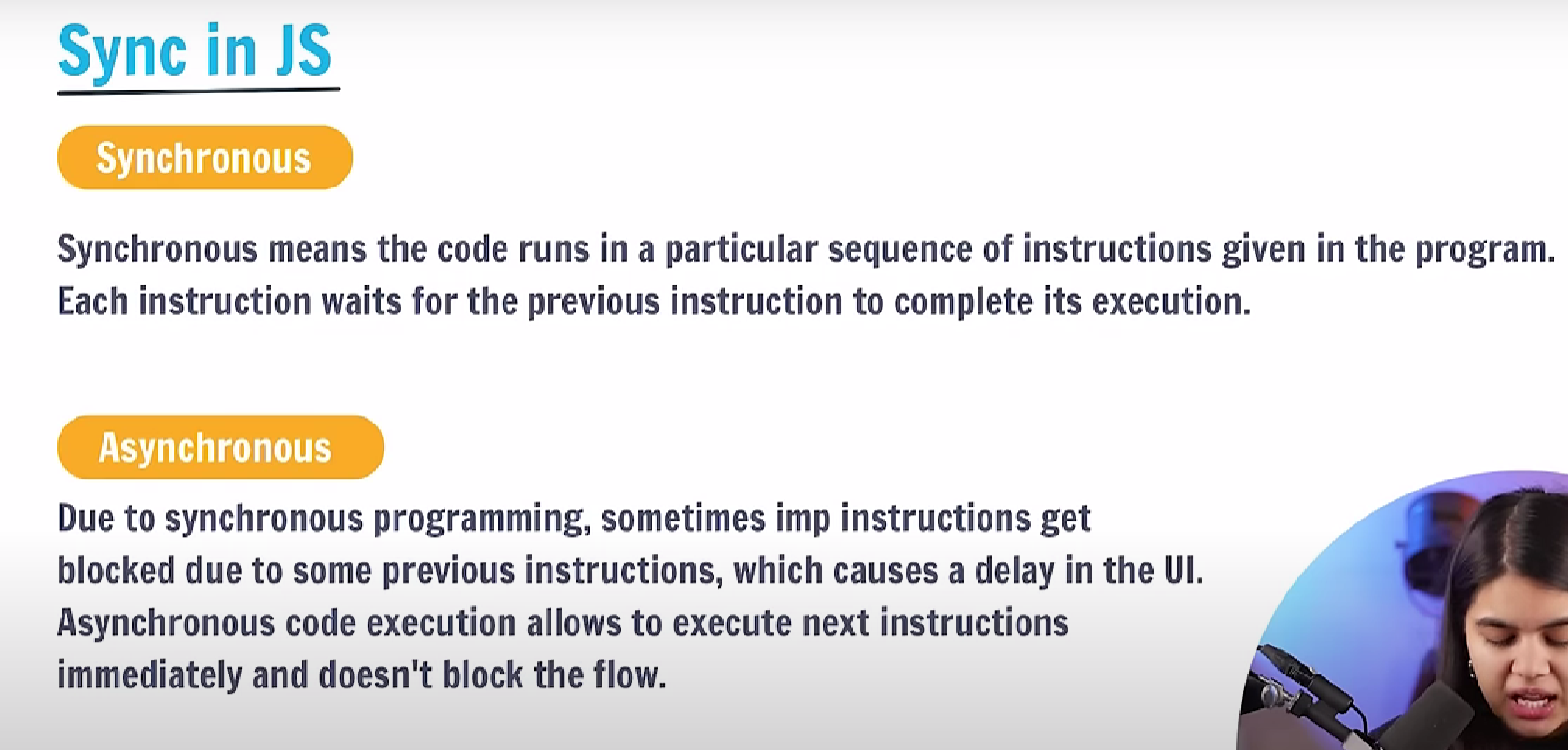
let a= setTimeout(function, time in milliseconds(kitni der me tumhe ye kam kervana he))

clearTimeout(a)=>ye ek id dega jo hmne a me store kiya or ush ki help se hm ushe clear krenge.

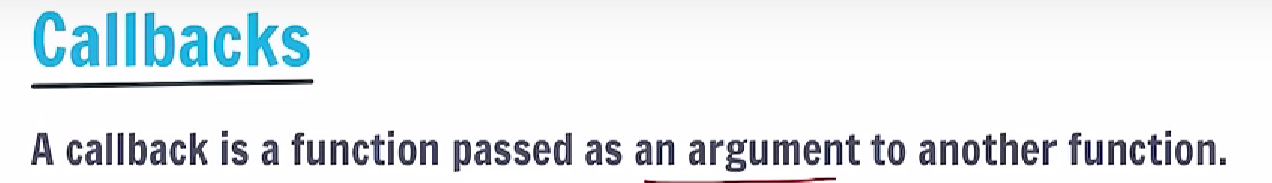
* setTimeout diye hua fuction ko ek hi bar execute krega ,diye hua time khatam hone ke baad.

let a=setInterval(function, time in milliseconds(kitni der me tumhe ye kam kervana he))

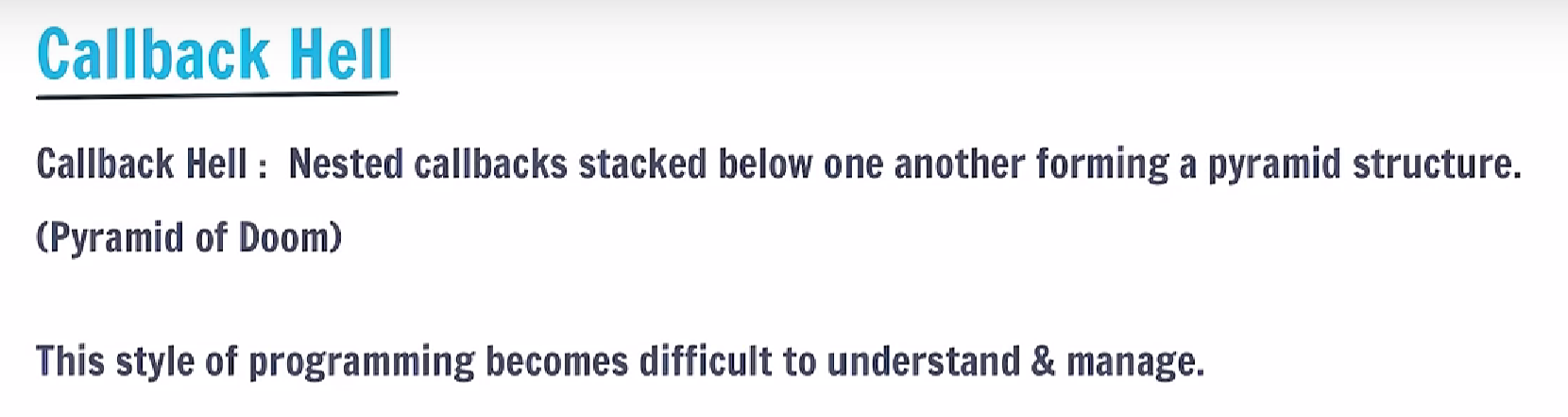
* yeha ye kaam bar bar ush diye hua time pe krega
* Eg;- ager hmne 1 second ka time diya ,to setInterval wo kaam hr 1 second ke baad ye kaam ko execute krega.
* Yeha kaam 1sec ke baad hoga baar baar or loop me 1sec ka delay nhi hota wo turen execute ker deta he baar baar.
* Callback() / callback Hell problem:-



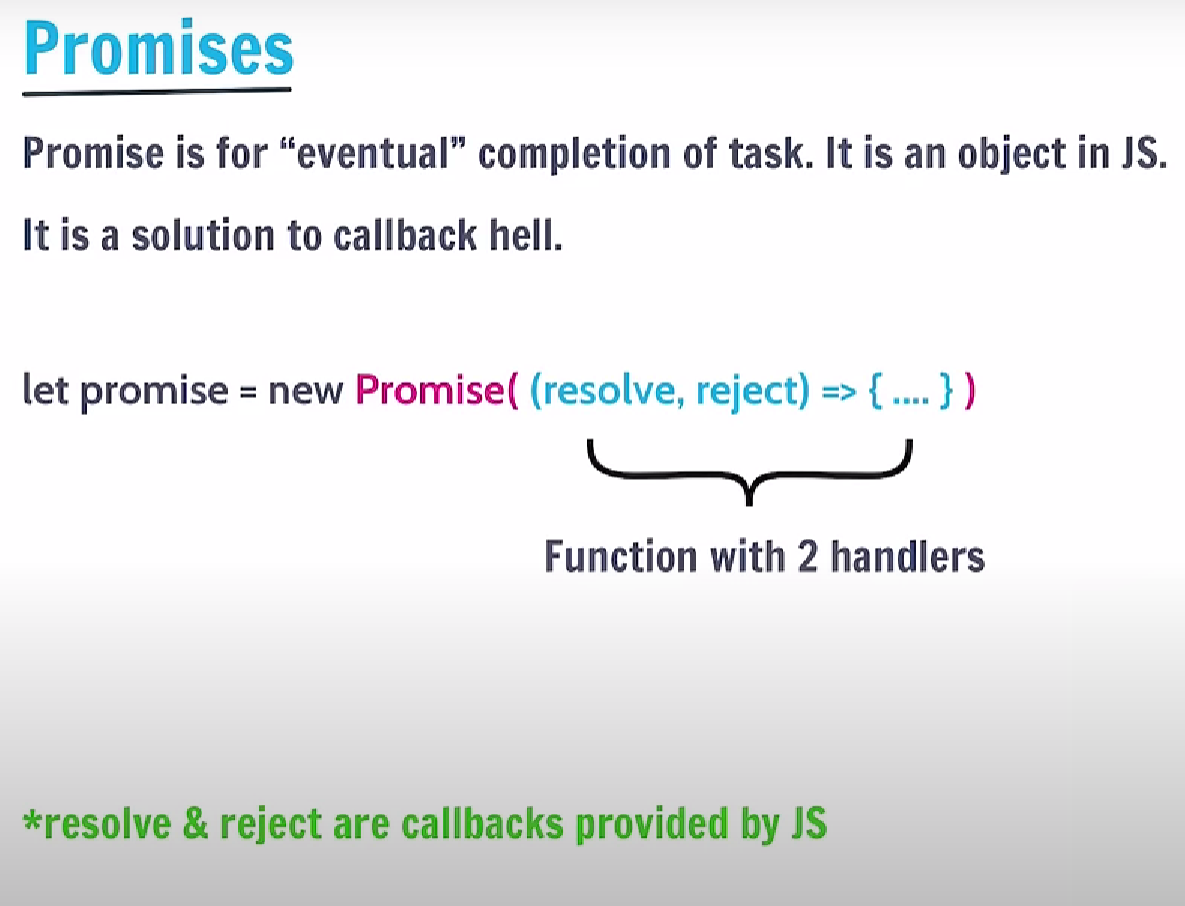
* Callback function:-

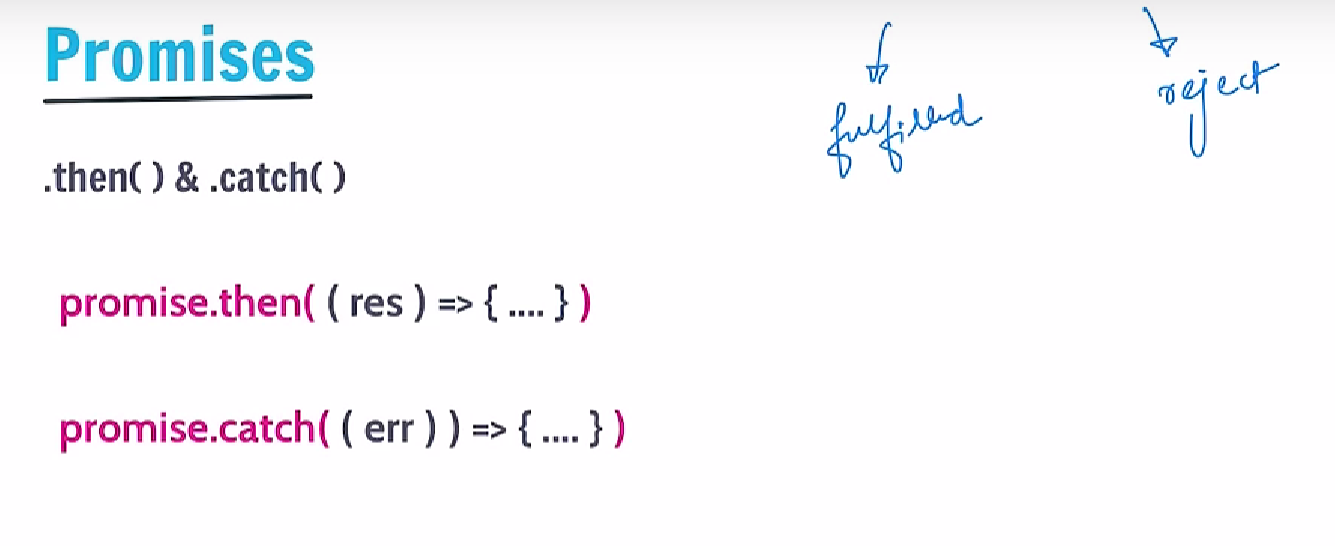


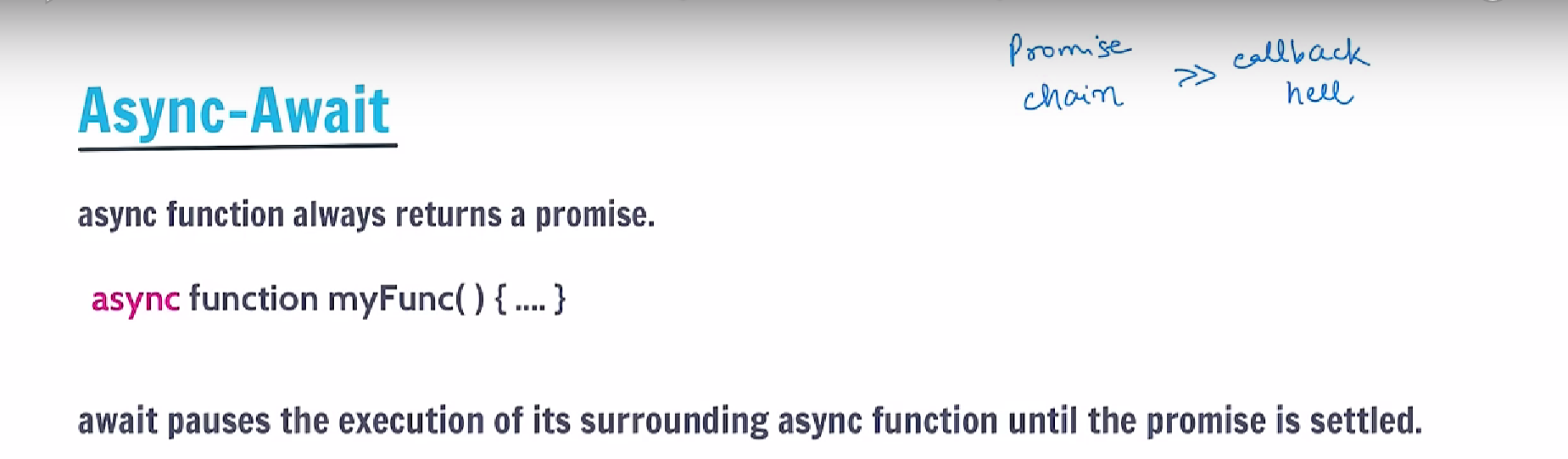
* Callback hell:-



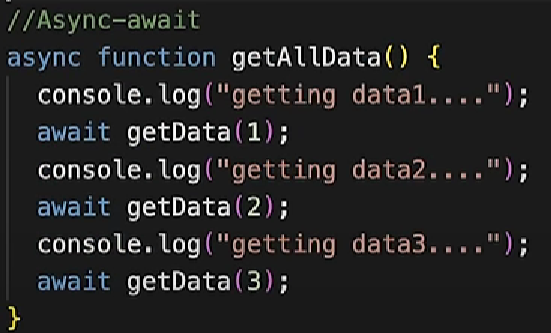
* Promise:-



* Promise.then() / promise.catch():-
* async function:-



* Await use:-



* IIFE:-

