Web development by apna college

=> we get a package form the server as a response and then from the package the browser builts the website or us.  
=> html mein jo markup hai uska mtlb hai – ki ek website ke upar jitni bhi cheese hai unka kya format hai aur kya structure hai.  
=>tags container hote hai aur unke andar jo kuch rakh ahi usko content. Aur dono ko sathme html element kehte hai  
=> <b> </b> idhr jo kuch bhi andar dalenge owh bold hojaega hota hia use   
=> ctrl r is used for refreshing the website.  
=>tags ke andar tags are also possible eg <p> This is <b>awesome</b>. Theres something called <i></i>.This italisize the paragraph. Nested tags.  
=><!DOCTYPE html> this tells ki hum konsa version use krre hai.  
 <html> ko root tag kehte hai  
 <head> ke andar meta information jaati hai – eg css link, js link. (idhr meat data mtlb data about data.). Idhr hi title tag aata hai.  
 <body> mein wih sab cheese hoti hai jo humare wesite mein actually mein show hongi. – eg - para heading list buttons.  
=> shortcuts on vscode are called emmet egs ‘!’.  
=> indentation mtlb proper spacing.  
=> <ol type = “a”> - ab jo list hai who a se shrur hogi. Could have made it start from A as well.  
=> Html attribute – attributes are usdto add more information to the tag. Eg - <html lang = “e”>. yeh toh lang e haina yeh hai tag hai. Upar <ol type = “a”> mein type = a yeh bhi ek tag attribute hai.  
=> anchor element is used to add links on to the website.  
=> href stands for hypertext reference   
=> anchor tags ke andar hum do types ke links pass karskte hai – jiase ki 1) absolute link (direct link to big sites  
2) relative link (file ke links)  
=> img tag is a single tag aur jo humlink share krter hai who hamesha relative hi hona chaiye.  
=> agar relative address nahi dena toh same folder mein hi html filea ur image ko rakho.  
=>relative online bhi use ksrskte same wasehi  
=> <br> <b> <i> <u>  
=> <! -🡪 comments  
=> mdn is noice developed by developers  
=> html is not case sensitive

8.HTML (level 2)  
=>Inline vs Block elements   
 -block elements takes up the entire row (block) -inline takes only the required width  
 -always starts from a new line -doesn’t start from a new line  
 -eg <h>,<p>, -eg <a> <img>

=>Div element – also called content division element – kudka kuch kaam nahi hota bass elements ko group krne mein help krta.  
=> ab maine agar teen anchor elements bane toh who teeno same line mien aajeange par agar main chata hu ki who alag alag hot oh main div ka use krunga  
 eg <body>

    <div>

        <a href="www.google.com">Google</a>

        <a href="www.youtube.com">Youtube</a>

    </div>

   <a href="www.facebook.com">Facebook</a>

</body>

Toh ab google aur youtube same row mein aare hoonge aur facebook alag se neeche  
=>Span element – yeh bhi group karne ke hi kaam ata hai but div se alaga aise hai ki yeh inline krdeta hai elements ko.  
=> <hr> is a divider whereas <br> is a line break (hr main line aajati hai)  
=> <h1> h <sup> 2 </sup> 0</h> aur ek sub bhi hota hai  
=> semantic markup – mtlb jo bola jaata hai wahi hota hai who tag bhi. Eg header  
 seo friendly banegi website humari. (search engine optimization) isse doa ur fayde yeh hai ki ek toh scrern reader ko help krta hia aur website ko aur readable banta hai.  
=> <header>,<footer>, <main> ( main content of the website), <nav> (navigation tag), <article>, <section>, <aside>  
=>html entities – aur browser use krta hai interpret  
  
&lt; lower than &gt; &hearts &amp; &quot;  
&nbsp; space dega beechme kiuki ek se zyad spacs allowed nahi hoti in html – non breaking space  
=> emmet – set of toolkit shortcut – mdm mein dekhskta- eg – p>div+img ab yeh yeh krdega  
<p>  
 <div></div>  
 img<>  
</p>  
ul>li\*5 // yeh bhi cool hai

#HTML (level3)  
#tables-  
=>it has 3 tags  
<tr> - row  
<td> - displays tale data  
<th> - header  
<table border=”black”> //to give border to the table  
=> Semantics in tables  
- <thead> - to wrap the table header  
- <tbody> - to wrap the table body  
- <tfoot> - to wrap the table footer // jab sum wagera cahiye hota hai

#Colspan and Rowspan Attributes – used to create span over multiple rows and columns //ek cell kitna space gher ra hai  
=> did question 7

#Forms  
=><form> content </form> // in body  
=>Action attribute – it is used to define what action needs to be performed when a form is submitted or where the form data should be sent. //submit krne ke baad kya action lu??  
 <form action=”/action.php”>  
 < form action=”/action”>  
=>Input element -used to create multiple form controls  
 - there are multiple types of inputs that can be createcd using type attributes  
=>khali agr <input /> toh ek box ajaega jisme main kuch likh skta hu  
=> famous types of input attritube  
<input> //by default yeh text hi hai  
<input type=”text”/>  
<input type=”password”/>  
<input type=”number/”> //my numbers ke alawa kuch type nahi kr skta  
<input type=”time”/>  
<input type=”color”/>  
email  
date  
and many more  
=>placeholder attribute – kuch type krne se phele jo dikhta hai (light sa dikhta hai)  
=>labels – these elements represents a caption for an item in a user interface also jab label par click krunga yeh apne aap mujhe filed mein le jaega //connection form hogya  
=>classical way for the same  
<label for=”username”>Enter your username: </label>  
<input type=”text” id=”username” placeholder = “username”>  
=>buttons - <button> submit </button> form ke andar wale jitney bhi buttons honge woh ek hi jagah le jaenge saare  
=>there are three types of button  
1)<button type=”submit”>submit</button>  
2)<button type=”button”>do something</button>  
3)<button type=”reset”>do something</button>  
=>wecan create button using input tag also (which we will not use)  
<input type=”submit”>  
<input type=”button” value=”click me” >

#Name attribute  
=> <label for="Username">Enter your username:</label>  
 <input type="text" id="Username" placeholder="Enter Name" name="username"/>  
uske bad main agar sumit krunag to next page mein username = abcd aega  
Basically name mein variable ki tarah use krskte aur url ke saath paas hoti hai

#Checkbox   
=> agr tick krke submit krunga toh age=on aega

#Radio Button (ek hi select hoga) (yes no aur amle female mein use hoga hia)  
=>

#Select  
=>selected akrdo toh pehle who selected aega  
#Range input element  
=>step  
=>value  
#Text area  
  
#CSS  
#Css  
=>Sundar ho tum  
=> ek language hi hai yeh bhi  
=>h1{  
 color : red;  
}  
here h1 is the selector, color is the property and red is the value.

#ways to include css  
index.html  
=> 1. Inline Style  
<h1 style=”color:red>Apna College </h1>  
=>2. Style tag in head tag  
<head>  
 <style>  
 h1{  
 color:red

}  
 </style>  
</head>  
=>external css  
in <head>  
 <link rel=”stylesheet” ref=”style.css”> //agar file same folder meinhai toh sirf name warna poora file path likhna padega  
 </head>  
#Color Property – foreground ka color change krne ke liye  
file-color.html  
1.color:color\_name;  
2.background-color:color\_name;  
#practice question  
file-practice.html  
#Color Systems- 1. Named Colors //ab 150 colors pehchanta brwoser  
2. RGB color:rgb(0,0,0) 0-255 //use color picker online se  
3. Hex code #ffffff   
color: #ffffff hexadecimal 0 1 2 3 to 8 9 a b c d e f  
#Text Properties  
file-textProp  
1.text-align  
 text-align: left/right dono ka mtlb same hi hota hai  
 text-align: right/end dono ka same mtlb  
 text-align: center  
 text-align: justify page mein bharke dikhe mtlb spred to fill left and right  
2.font-weight  
 font-weight: normal //400 values 100-900 hoti hai  
 font-weight: bold //700  
 font-weight: 100  
 font-weight: bolder  
 font-weight: lighter  
3.text-decoration  
 -sets the appearance of decorative lines on text like underline( underline, overline, line through)  
 -decoration ke kaafi alag cheese bhi hoti andar  
 eg text-decoration-color  
 tex-decoration-style  
 -iska kaam k aur hota hai anchor tag ki blue line ko remove krna  
 text-decoration: none;  
4.line-height  
 -controls the height of the line of the text  
 line-height: normal //  
 line-height: 2.5 //mtlb 2.5 tmes of normal  
5.letter-spacing  
 -controls the horizontal spacing behaviour between text characters  
 letter-spacing: normal  
 letter-spacing: 10px  
6.font-size  
 absolute relative  
 px %  
 pt em  
 pc rem  
 cm ch  
 mm vh  
 in vw + many more  
96px=1inch=2.54cms  
72pt=1inch  
12pt=1inch  
#pixels- not suitable for responsive websites  
font-size: 50px

#Font-family – kaunsi font style  
-agr koi aisa font family hai website par jo tumari brower nahi samjta toh bhi font dikheag bas by default wala hota hai use tab  
-backup in font family  
font-family: avant garde, didot, sans-serif;

#Practice2 done text-transform: uppercase/ lowercase.

#CSS2  
#Selectors in css  
file-css2  
=>types of selectors  
1. Universal selesctor – to select everything in the document.  
\*{  
property: value;  
2. Element selector – selects all the elements of the same type  
h1{  
Property: value;  
}  
h1, h2{  
property: value;  
3. ID selector  
#myid{  
property: value;  
}  
4. Class selector  
.myclass{  
property: value;  
}  
5. Decendent selector - check file  
6. Adjacent Sibling Combinator(+) – immediately bad wala only  
eg agar <p> ke baad kaafi types ke agar <h1> hai aur hume just next wale jo target krna hai toh apan use krte isko eg <h1></h><p></p>.  
7. Child combinatory(>) – selects all buttons which are direct children of spans (only ek level andar)  
eg <h1><p></p></h1>  
8.Attribute selector- selects elements based on the presence or value of a given attribute

#pseudo class – a keyword to a selector that specifies a special state of the selelcted elements  
1. :hover  
2. :active  
3. :checked //iska code dekhna  
4. : nth-of-type  
=>radio input lete hue agar banaya toh name dene se dono ko ek hi chek hoga

#pseudo element  
1.::first-element  
2.::first-line  
3.::selection  
  
#selector specificity – 1. Inline>id>class>element  
2. More selectors > less selector  
3. If specificity same tab cascading dekhte hai.  
 #practice ques tikon very good for revision  
#important inline sebhi zyada important  
h3{  
 color: yellow !important; // ab yeh kabhi bhi change nahi hoga  
}

#inheritance  
#color palletes dekhlo online eg canva colorspace or coolors  
#some properties are not inherited like width, height, border.

#CSS3  
#Box model  
file-css3.html  
1.=>height (content)  
div{  
height: 100px;  
}  
2.=>border  
a. border-width  
b. border-style  
c. border-color  
=>border shot hand  
div{  
border:2px solid blue;  
}  
=>border sides  
border-left/right/top/bottom.  
=> border-radius and border-top-right  
div{

Border-radius: 15px;  
}  
=>to make a circle make height width same and then border-radius:50%   
3. Padding (border aur content ke beehcme)  
padding-left  
padding-right  
padding-top  
padding-bottom

=>padding: 50px(all) / 10px 20px(tb and lr) /10px 20px 30px(t and lr and b)/10px 20px 30px 40px(t b l r)  
4. Margin (do divs ke beechka)  
margin-bottom  
margin-top/left/right

=>div ka right margin nahi control krskte.  
  
#Display  
display: inline; //jo cheez block jaisi dikhti hai use inline krdeat hai  
display: block; //jo inline hoti hai use bloack jaisa krdeat hai

#inline-block – ab hota kya hai koi bhi block mein aap styling krte fir usko and inline krneg toh cheese uspe how nahi hoti hai toh isliye use krte inline-block.

#relative units & em and rem //dekhlena yaar firse  
=>mtlb parent ka 50% hi le tu  
=>em use hota hai topographical mein  
=>//

# Selecting elements by id   
=> document.getElementbyId(mainImg); this wont work because mainImg ko variable ki tarah trat krra.  
=> document>getElementbyId(“mainImg”); this will work aur jo outpout milega console min woh object milega imge ka basically an object of image.  
=> we can store it in a variable   
let imgObj = document>getElementbyId(“mainImg”);  
=> imgObj.src = “kuch new value” - toh poora new hi hojaegi value;  
=> ab agar paragraph acces kran hai toh what we can do is   
document.getElementbyId(“description”);  
ab agr iski hume object form cahhute(same agar image ki chahiye hoti) toh hume console mien yeh print krna hai – console.dir(document.getElementbyId(“description”));  
=> null milega agr hum koi aisi id ko access krne ki koshish krnege who hai hi nahi.

# Selecting elements by class name  
=> it returns an html collection(similar to array but not array) or empty collection(if not found).  
=> document.getElementsByClassName(“oldImage”);  
=> agr do teen hot oh index deke ek particular select kkar skte  
docment.getElementsByClassName(“oldImg”)[0];

# Selecting getElementsBy TagName  
=> returns the elements as an HTML collection or empty collection(if not found);  
=> to select all the paragraph elements of the page we can do is  
 document.getElementByTagName(“p”);

#Query Selectors

#27 JavaScript (Part 9)  
=>everything in html becomes an object with javascript.  
=>document object store karti hai saari ki saari html aur css code ki so any change you do in this object will have change in html and css also.  
=>console.log(document) //yeh htmlcode ki tarah dikhega  
console.dir(document) //yeh document object ko print karaega  
=>#Selecting Elements  
1. getElementById -document object ka method hai ek  
-isse hume who element nahi balki us html element ki ek object form milegi.  
=>document.getElementById(mainImg) //null dega kiuki mainImg variable jaisa treat hora hai   
=> document.getElementById(“mainImg”) //aur isse hume image ka object milega  
2. getElementByClassName //agar collection mila to array ki tarah use karskte using[]  
3. getElementByTagName

#Query Selectors (ctrl + l to clear console)   
=> only ek dega who bhi fisrt wala not like methods  
=> same css selectors jaise hote hai  
 document.querySelector(“p”); // Selects first p elements  
 document.querySelector(‘#myiD’); // Selects first elements with name = myID  
 document.querySelector(‘.myClass’); // Selects first element with class = myclass  
=> aba gr hme sab chahiye naaki only ek use   
 document.querySelectorAll(“p”); //selects all p elements

#Setting content in objects  
=> using properties and methods  
 =>innerText - show the visible text contained in a node.  
 =>textContent – shows all the full text (hidden bhi) (html mein jaisa dikhar hoga waisa hi dikhar hoga).(code)(display none wlaa bhii)  
 =>innerHTML – shows full markup language.(tags styling sab)  
=>para.innerText="Hi I am Peter Parker." //console mein yeh likhunga toh <b> bhi ajaenge isliye InnerHTML  
=>para.innerHTML="Hi I am <b>Peter Parker</b>."  
=>h.innerHTML = `<u>${h.innerText}</u>`

#manipulating attributes (id, class, style, img ka src)  
=>1. Obj.getAttribute(attr) //getters  
2. Obj.setAttribute(attr) //setters  
=> eg  
let img = document.querySelector("img");  
img.getAttribute("id");  
img.setAttribute('id', 'spidermanImg'); //styling hatt jaegi change krne par  
=>eg2  
img.setAttribute('src' , 'assets/creation\_3.jpeg')  
=>koi attribute hai nahit oh hum use set bhi lkar skte hai (idhar img ki koi calss thi hi nanhi)  
img.setAttribute('class' , 'image') //now this particular image has a class as well.

#Manipulating style with style attributes  
=>dom mein styl eis in camel case. Eg- backgroundImage  
=>img.style is khul jaega  
=>obj.style eg. h.style  
=>h.style.color="purple"  
=>h.style.backgroundColor="yellow"  
=>let links = document.querySelectorAll(".box a") //  
for(link of links){  
    link.style.color="yellow"  
}  
=>let links = document.querySelectorAll(".box a")  
 for(let i=0; i<links.length; i++){  
    links[i].style.color="yellow"  
 }  
=>style property in dom humein inline style deti hai naaki css file wala isiliye apan log kam use krte hai style property.

#classList  
=>obj.classList – se hume uski saari classes ki list mil jaegi.  
=>obj.classList.add  
eg – h.classList.add(“.green”) //green mein color green set hai, eg use krte hi color of h wll change to green.  
=>obj.classList.remove(‘green’)  
=>why we not use set attribute over classList.add – kiuki ek time pe ek hi set sttribute mein ek hi class set kr skte aue set krte hi porani saari styling khtm ho jaati  
=>obj.classList.contains(“name”) //name of the class you want to check  
=>obj.classList.toggle() to toggle bw remove and add

#Navigation in JavaScript  
=> 1. Oj.parentElement 2.obj.children 3.obj.previousElementSibling / obj.nextElementSibling

#adding elements using dom  
=>document.createElement(‘p’)  
eg - let newP = document.createElement('p');  
abhi maine use sirf create kiya hai abi use insert bhi karn ahota hai, how?  
=> 1 .appendChild(element)  
eg - let body = document.querySelector('body')  
body.appendChild(newP)  
eg2 - let box = document.querySelector('.box');  
box.appendChild(newP); //ab kya hoga who bosy se hatt jaega aur box ke neeche lag jaega  
eg3 – creating button  
=>let button = document.createElement('button')  
box.appendChild(button);  
button.innerText = 'Click me';  
  
=> 2 . append(element) – usi element mein change. And to add other element also  
newP.append(“ this is new text”); or newP.append(button); //iise para aurb tutotn ek hi single line mein aajaenge.

=>3. .prepend(element) – shuru mein add krdega

=> 4. insertAdjacent(where, what);  
eg - p.insertAdjacentElement('beforebegin', btn); //para start hone se pehle  
 p.insertAdjacentElement('afterbegin', btn); //para ke satring mein //sathme  
p.insertAdjacentElement('beforeend', btn);  
p.insertAdjacentElement('afterend', btn);

#Removing element from page  
=>1. obj.removeChild(element); 2. Obj.remove(element) //same append jaisa  
=>body.remove() //poora page gayab

#Practice Qs

#28. JavaScript (Part 10)  
#DOM events  
=>Events – signals that something has happened  
=>eg     <button onclick="console.log('Button was Clicked')">Click me!!</button> //click is an event  
but then inline mein krna is not much efficient.

#Mouse/ Pointer Events  
=> 1.onclick (when an element is clicked)  
=>let btn= document.querySelector('button');  
console.dir(btn);  
btn.onclick = function() {  
    console.log("Button was clicked");  
}

=>let btn= document.querySelector('button'); //same upar wale ki tarah  
console.dir(btn);  
function sayHello() {  
    alert("Hello");  
}  
btn.onclick = sayHello;

=> ab agar kaafi buttons hai tab  
let btns = document.querySelectorAll("button");  
for(btn of btns){  
    btn.onclick = sayHello; //idhar sayHello() nahi krna kiuki () mtlb excute krna aur refresh hote hi hojaega execute 4 baar hum assign krrre  
}  
function sayHello() {  
    alert("Hello!");  
}

=> 2.onmouseenter (when mouse enters an element) (basically hover)   
eg btn.onmouseenter = function() {  
 console.log(‘You entered a button’);  
}

#Event Listeners  
=>if onclick and listeners do the same thing why do we need listenres, because onlick se ek hi baar hhoti hai function excecute or ek hi baar hoti hai value set. Agar hum chahte hai ki multiple chheze hot oh hum evemntListeners ka use krte hai  
=>addEventListeners  
syntax - element.addEventListener(event, callback)  
eg   
btn.addEventListener(“click”, function() {  
 console.log(“button clicked”);  
}  
=>ek dblCLick bhi hota hai //good example  
=> let btns = document.querySelectorAll("button");

for(btn of btns){  
 btn.addEventListener('click', sayHello);  
    btn.addEventListener('click', sayName); //also dbl click ko use krne ke liye click ko comment krdiya   
 btn.addEventListener('dblclick' , function() {

        console.log('You double clicked me');

    });  
}  
function sayHello() {  
    alert("Hello!");  
}  
function sayName() {  
    alert("Hemant");  
}

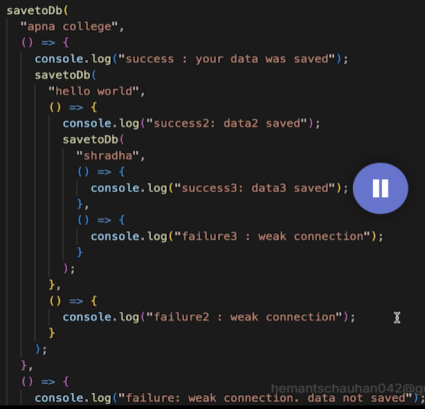
#Random color Generator Done

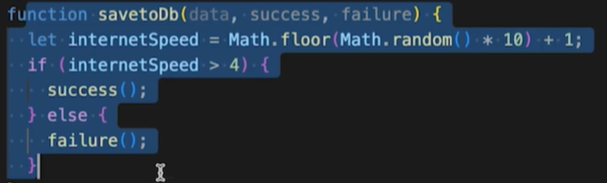
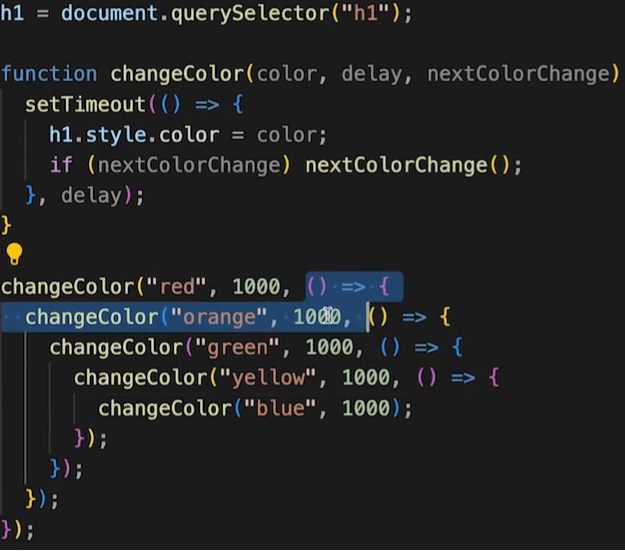
#Event listeners for Elements  
=> we can even set event triggers for different elemetns also  
let p = document.querySelector(“p”)  
p.addEventListener(“click”, function () {  
 log(“Para was clicked”);  
}

#this in Event Listeners  
=>this is used in callback of event handler of something, it refers to that something.(khudka color change krne mein use hoga).  
=>redundancy removal (mtlb code mein baar bar likha hai jiski zaroorat nahi thi)  
=>keyboard (event) mein do cheese we need to look into key and code

#Form events  
=>1. submit //next page mein kya gya  
=>2. event.preventdefault //next page mien jaane se bachane ke liye  
=>3. Inp.value //jo value submit hui hai woh access krne ke liye use hota hai  
=>form.element[0] or this.element.element[0];  
=>change event vs input event //made a text editor  
=>non character keys(enter,shift, arrow keys) don’t do change,

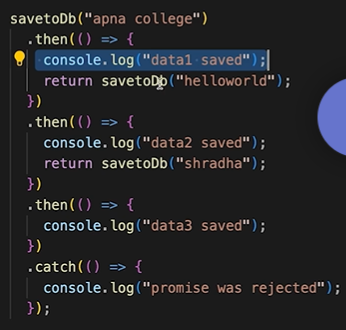
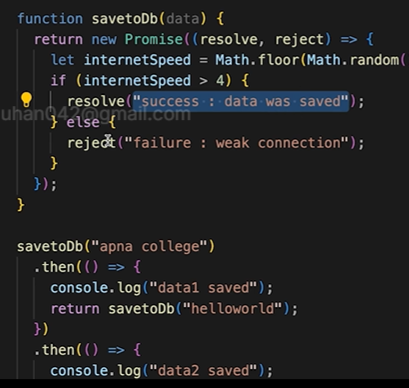
#JavaScript(Part 11)  
#Js call stack – breakpoints(jidhr se start krna hai) , sources mein jakar app.js ko slect krke.  
#Why is js a single threded language? – mtlb ek time pe code ki ek hi cheez execute hopaegi.(c++ is multi threaded language. (callback func who func jo as parameter jaate ek koi aur func mein)  
#synchronous nature mtlb ek ke baad ek cheese hori hai but jab hum js mein setInterval, setTimeout, callback jaise functions use krte hai toh hum js ko asynchronous bnra hai.  
Thus, js being a singlethreaded language still can perform asynchronous things.  
#Problems with asynchromnous nature of js  
1. Callback hell

#Setting up for promises  
   
 //problem hai yeh ek ke andar depend krna

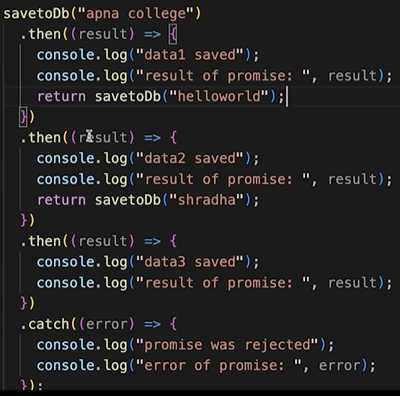


Asynchronous – ek ke baad ek cheez depend krti hai.  
ab solution is promises -  
The promise object represents the eventual completion (or failure) of an asynchronous operation and its resulting value. It has two things within it 1.resolve(success), 2.reject(fail).

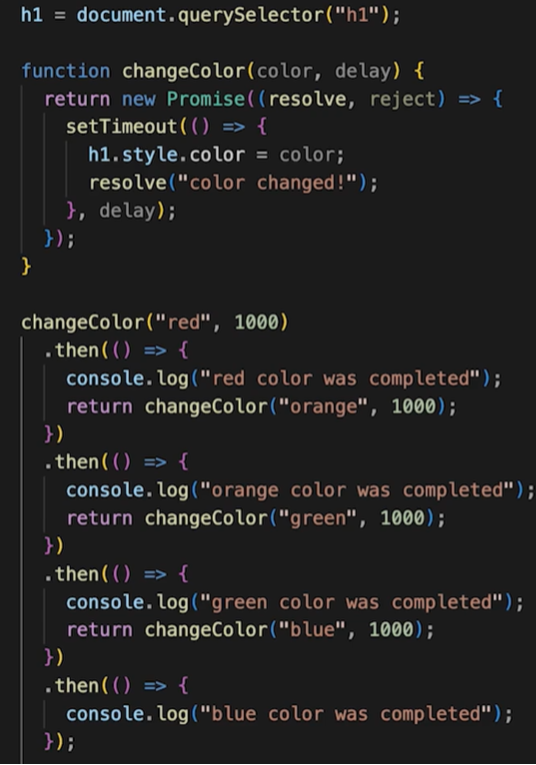
#resolved

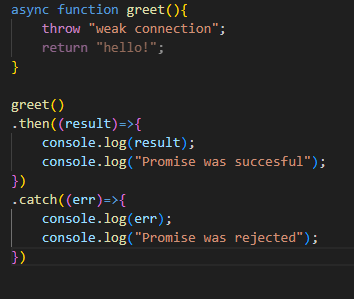
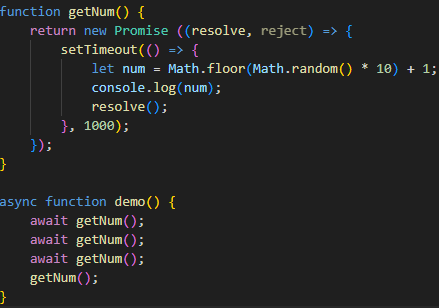
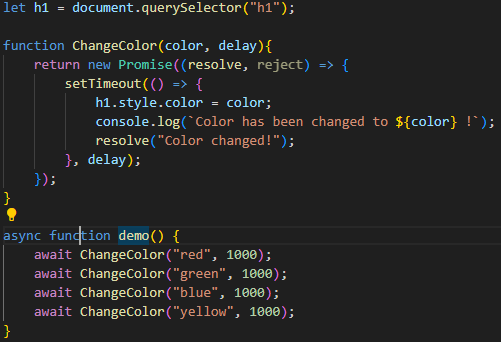
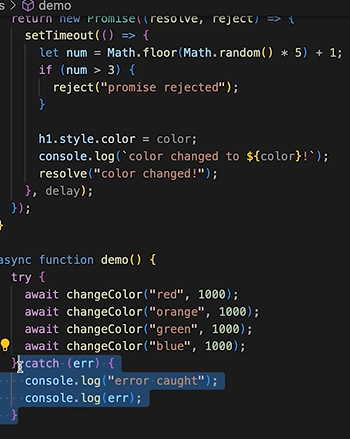


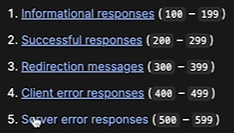
#Results and errors in romises  
=>   
  
  
#better code for change in the color

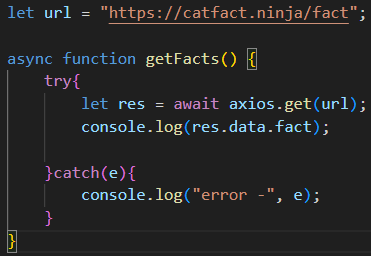


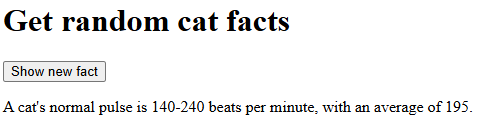
#

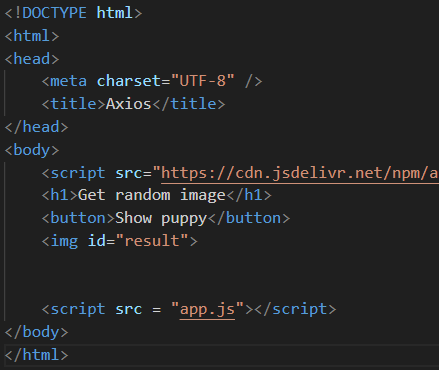
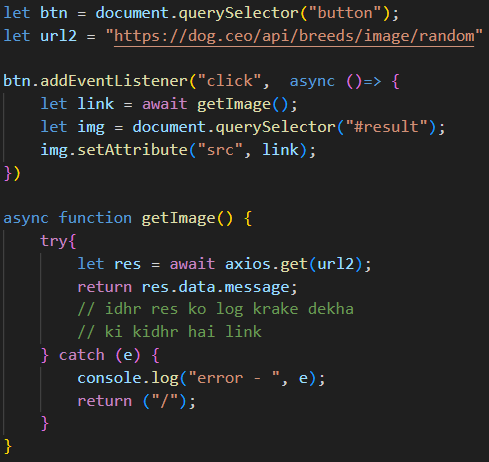


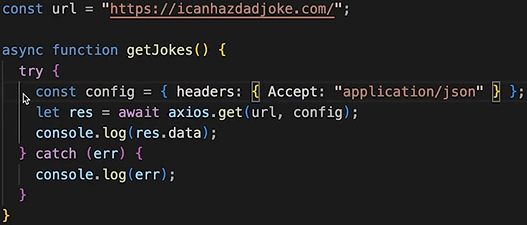
Check the savetodb file also.  
#JavaScript 35 (part 12)  
- async func greet(){};  
- by default async functions return a promise.  
- throw – throws errors  
-  
-Await – pauses executions of surrounding asyn function until proimise is settled  
-  
-jbtak pehli wale getSUm ka promise resolve nhi hota tabtak aage ki calls nhi lagengi  
-  
-handling rejections with await  
-  
-API and JSON  
-application programming interface  
-js stands for java script object notation (zarori nahi hai ki javasxpirt meion hi chla  
-har api json mein hi hota hai (iise phele api xml format mein aati thi)  
-check kr ki json valid hia oth use jsonlint.com  
-json adat is in string format  
-hum kiuki string nahi but object tko access kr pate hiatoh we use JSON.parse()  
-aur agar json object ko string mein convert krna hot oh we use JSON.stringify(json)  
-ajax - asynchronous jacascirpt and xml --- api calls ko ajax calls kehte hai - yeh aane jaane ka process poora asynchronously chlta hai  
- https code-

  
-https://www.google.com/search?q=mango&sca\_esv=6db176930c53557b&sxsrf=AE3Tif  
-yeh upar q wala is query string mtlb – humne ? ke baad kuch q value pair mien ek additional info bheji  
- jis type mein hume chahiye cheez uski type ke liye headers use hota hai  
-to get api in our js code we use fetch  
-fetch function ek promise return krta hai  
  
-res.json readale data ko json mein convert krdeta hai  
-usign await and fetch together  
- neeche agar await nhi lgaya toh js kuch bhi print krdea but then   
ab await hai toh full waiting and then sahi se print  
  
-yahi errors hoti toh  


Axios  
-its a library to make https requests also it’s a better way of fetching data  
- axios ki library mein jaake uski script copy krni hai mail html mein  
-axios ki zaroorat kiu hai kiuki fetch ke bad hume pasr krna pdta object mein but idhar yeh zaroorat nahi  
-

-get facts ek async function hai jo ki ek promise return krra hoga, toh hume await use krna pdeaga in eventlistener, getfacts() ko await bnana pdeag toh toh event listener ke callback function ko async banana pdega  
   


#dog picture api  
 

#sending api with header request  
-pehle html mein ara tha data but after setting header ab json mien aane lg gya

#33Terminal

#Basic command - ls (list of all files and folders in the place you are)  
=> ~(home directory)  
=>pwd (print working directory) - kidhar hia hum abhi  
=>clear (clear on console)  
=>cd (change directory)  
=>cd .. (peeceh chall)  
=>cd ../..( do baarr peeceh)  
=>/ wala absolute path hota hai aur withour slash relative path hota hai  
=> / (root directory)  
=>man – tell what the use of the command  
=>mkdir (make directory)  
=>flags ls –l (yeh –l flag hai) –l extra chehzo ke sath show krega  
- ls –a show krega directories jinka naam . se start hota hai.  
- touch name.ex (us exrtension ki file banjaati hai)  
- rm (remove files) permanently  
- rmdir (remove directory) only empy folders  
-rm –fr (remove with force)

#34 Git and Github  
- git ek version control system hai (tracks changes in the code)  
- README.md – md is markdown  
- git clone <some link>– clones from repositories  
- git status -   
- git add and git add . (. Mtlb sab kuch)  
git init (used to create new github repo)  
- git push –u origin main – isse mujhe baar baar git push origin main likkhne ki zaroorat nahi hpgi   
will work with only git push  
- git commit -am “added dot” - yeh dono adda ur commit ke sath krra hai  
-git checkout –b features – se new ek branch ban gyi  
git branch se current branch mien hop ta chal jaega  
- git checkout main – se main branch mwein switch krdoge  
- git checkout –d test (delete hogi bs kisi aur branch mien hona chahiye bnda)  
-feature mein hoke git diff main (this will tell about the differences bw the branches  
-ab aagr feature thodi aage chlri hai toh again features mein hoke git merge main  
-or pr (pr mtlb codes ko merge karwana)  
-git pull origin main (job hi changes hua hai hub par who idhar se pull hojaenge)  
- git reset file name – jo latest add kiya hia who kind unadd hojaega  
-git reset HEAD~1 (jo last commit hua hai use remove krna)  
-git log (saari commits ke baarein mein bta deaga)  
-fork mtlb ek rough copy banana kisi dusre ki repo ka

#Node JS

- is js ka run time environment  
- vs code btao node js par ban hai  
-Node REPL (console mien node type kar aur mast js likh)  
-REPL mein dom nahi kar skte  
- .help for commands  
-process object – process.argv  
-module.export  
module.exports = 123; (durse file mein)  
let someValue = require("./math"); (pehli fil mein) //require se milega woh  
Console.log(someVlaue); (toh 123 pribnt hojaega)  
-by default modlie.export empty object bhejta hai  
-export.function(){with definition} (this will work) but  
export =5 will not work  
-npm se packages milte hai  
-express.js aur react.js bhi packages ho hote hai send only json-packages  
-node modules tab delte krna jab kisiko ko code share krra hia(google thoda mtlb nhai aaya)  
- meta data mtlb data about data  
-apne project mein npm init se humari dependency banegi fir udhr hum koi bahr ki koi bhi package install kr skte hai)  
-module.export (only file path is enough)(also object ki tarah use) se require aur export se import(type ko moldule kar)(file path nahi poori file ko import krn ahai)  
  
#express js  
-library vs framework  
-both are pre written code but library ek chota kam kregi ex axios for api and framework application ka structure define krne ke kaam aegi(basically kais ecode likh ajaana chaiye)  
-express ek framework hai jo server side framework likne ke kaam aata hai  
- app.get aur app. Use – dono response dene ke kaam aate hai  
-app.get("/", (req,res) => {

    res.send("you contacted home path");

});

app.get("/apple", (req,res) => {

    res.send("you contacted apple path");

});

app.get("/orange", (req,res) => {

    res.send("you contacted orange path");

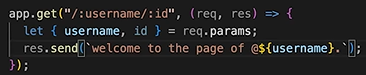
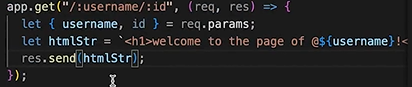
});

app.use((req,res) => {

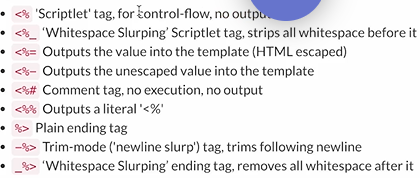
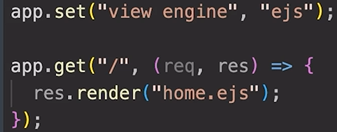
console.log("request received");

res.send("This is basic response");

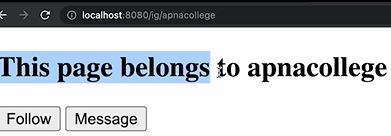
});

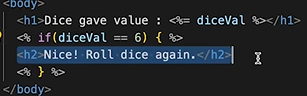
#path parameters  
- app.get("/:username", (req,res) => { //yeh /:username ek variable hai jo ki req mein store hota hai  
console.log(req.params); //yeh req mein jitne bhi parameters hai who krdo print  
res.send("Kya krra hai be");  
});  
-   
  
-hum infact poora html bhej skte  
 

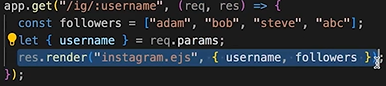
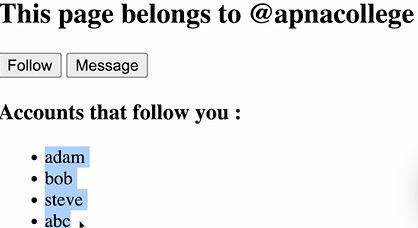
#Query string  
-  yehquery string hai who q= krke  
- hum ab query strng in req handel krenge  
-   


#EJS  
-express ko require nahi krana express apne aap krdeta  
- app.set("view engine", "ejs"); // set view engine to ejs  
-ejs is a templating language  
  
-ejs mein hu res.render krte hai  
aur usme bhi ek ejs folder ko hi bhejre honge  
-uss views folder mien ejs files bnti hai aur uss folder mien saare templates store hote mnje   
ejs files  
- aba gar hume bahar se use krna hai index jiske anadar ejsd hai toh kar skte but yeh kuch extra hai bs  
- handlebar and ejs are used for templating  
-   
-set function sets view engine to ejs -ejs ko by deafault express require krdeta haitoh require nahi krna pdta ejs ko  
-view engine mtlb templates ki baat krre  
-ejs se hum send nhi krte hum render krte hai aur uske andar ejs file ko bhejte hai  
-by default express ejs file ko views wale folder ke andar dhoondta hai.  
- toh agar hum bahar kisi folder mein toh hume path define krni padegi  
  
  
yeh do lines are needed to be writte in case we would like to run code bahar se jidghr views folder nahi ho  
-path package mein join function is used to join two paths  
-code mtlb ki set krde views ka path to jiss bhi directory mein hai usek andar ke views folder mien  
- \_\_dirname is index.js, jidhr se code run hua uski current directory //yeh hamesha krde hum

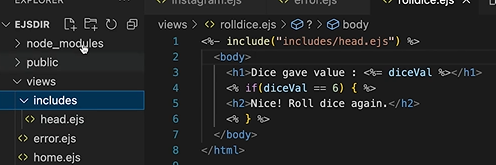
#Interpolation syntax – mtlb embedding expressions into markup text.(literals mein variable ka naam)   
- <%= %> yeh sab hote hai yeh – basically y eh html ko dynamic bna dete hai  
-[“hello’, “ bonjour”, ” namaste”][2] iska output is bonjour  
-   
 yeh aaya resultas as response  
-also ew can directly give diceVal aur ejs mein use it as num. 

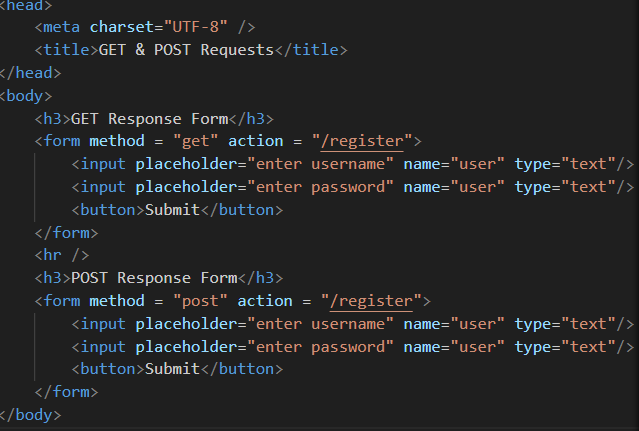
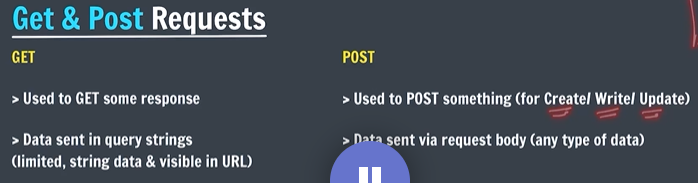
#Creating instagram page  
-  


#conditional statements in ejs  
 

#loops in ejs  
   


# Serving Static Files  
- abhi tak hum dekhre the ki kaise html ko render krre the but uske saath kuch kuch css aur js bhi aaskta hai toh tab kya krein, tab use hota hai   
- static express is a middle ware  
- static file humesha public namak folder se hi express leta hai  
- public folder saari css file serve krdega use path dene ki zaroorat nahi

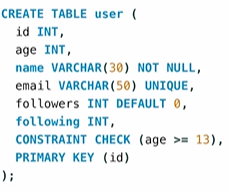
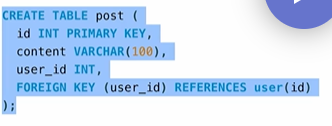
#includes/partials  
- sabhi ejs se head remove krdo aur yeh krdo  
- backend 3 ka last ke kuch are very nice

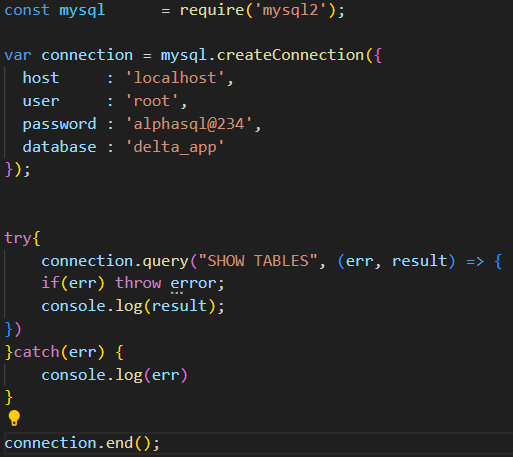
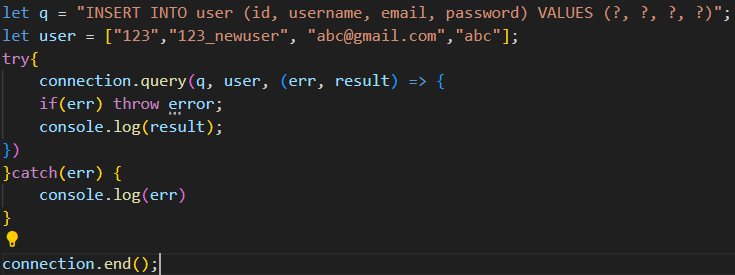
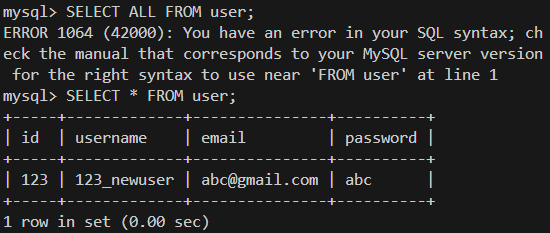
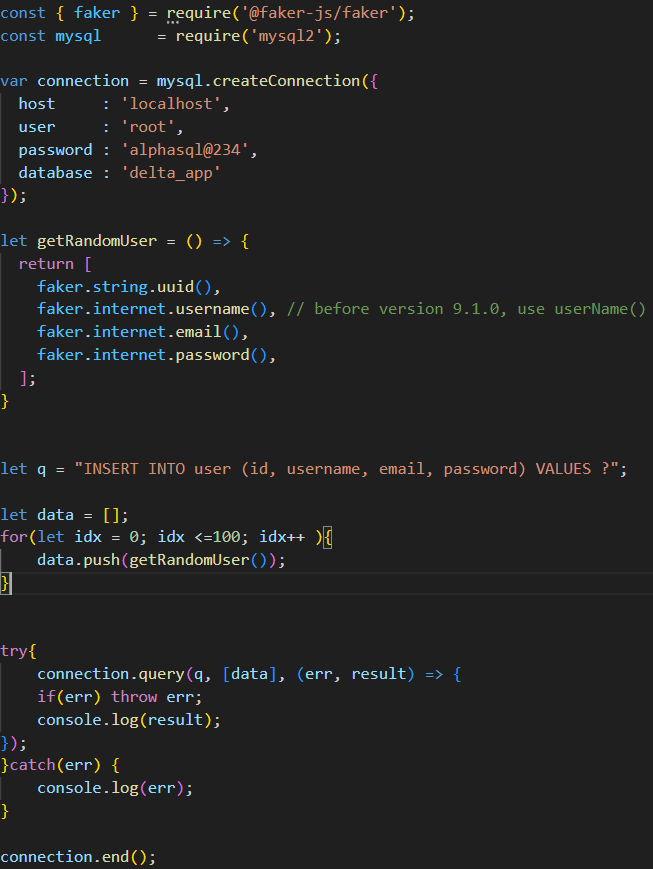
#38 Miscellaneous  
-   
  
-idhr humara form file protocol ko use krra hota hai but jab hum http:// wala   
krdenge do humare server se connect ho jaega yeh  
-get wale mien password wagera sab dikhra hai  
aur post request mein kuch nhi dikhra  
- who basically req ki body ke andar dikhta hai naaki url mien  
-

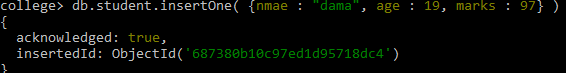
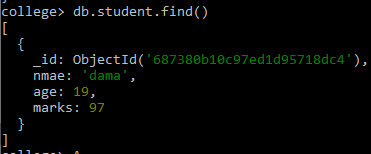
#  
   
get pe get response aegi aur post pe post response  
# ab post ke through data use krne ke liye we use   
- vvimp – req.body mien by default undefined hi hoti unless hum middle ware use krke bataye na

39. Backend 4 (REST)

#REST  
- representational state transfer.  
- it’s an architectural style that sets constraints to be used for creating webs services.  
- <https://stackoverflow.blog/2020/03/02/best-practices-for-rest-api-design/> pdhna isko ekbaar  
- crud ki cheez GET, POST, PUT, PATCH, DELETE  
- how to connect different pages – res.redirect(url)  
- status 3 mtlb redirect wale aur res.redirect gives back 302 status  
- res.redirect(“/posts”)  
-Get /posts/:id to get one post (using id)  
-

#40 Starting with SQL  
Database  
-SQL is language and Mysql is database (structured query language)  
- DBMS DB se baat krne mein use hota  
-sql is realtion database mtlb data stored in tables (tbales mtln relational) mysql,oracle,postgres  
-nosql jo hai who tables mein store nhi krte data but kisi aur treke se (mongodb, Cassandra)  
-schema mtlb kya cheez show kra table (columns)  
-tuple mtlb rows  
-   
-reverse engineer krke uyou can see how the tables are connected(un diagram ko er daigram kehte hai) (er – entity relation)

42. Backend 5 (Node with sql)  
#faker – generates faek data  
- by this method you can use sql on js(mysql package)  
-other way is to use myql –u root –p in comman line.  
-4th way to use sql is using sql file  
- in sql source schema.sql  
- dynamically aise krte add  
also to check ki add hua ya nahi -  
-bulk mein data daalna sql mein  
  
-

44. MongoDB (Part 1)  
- mongosh to start shell in cmd  
- ctrl + l to clear, show dbs (shows all dbs), exit or quit,   
- some js commands will work  
- the moment we run mongosh it creates a temporay db for us where we can run our database  
toh owh jo test hai , actually who ek temporay db hai  
- use college- agar collge hoga toh andar jaega arna new db bnaega  
- db – mujhe current db ka naam dega  
- BSON data – jis foamt mein data insert hota hai mongodb mein (binary json)  
- json is text bases and bson is binary, json is space ineeficient   
- documents – mongo stores data in the form of documents(bson data)- document is key value pair  
doxument rows same  
- mongo stores documents in collections - collection same as tables  
- show dbs  
- show collections (saare tables dikhao)  
-  to insert  
-  db.student.find() (show all documents of a collection)(also all documents are stored in the form of arrays)  
- mongo mein schema ke hisab se insert nahi knre ki zaroorat  
- db.student.insertMany([{},{}]) - multiple add hoskte  
- db.student.find({key:value}) –particular  
- db.student.findOne({key:value}) –particular par only ek  
- find(array) returns a cursor whereas findOne returns a keyvalue pair(object)  
- cursor bole toh a reference to original, ab cursor kiu dera taaki looping easy ho  
- db.student.find({city : “Delhi”, marks : 75}) – hum specific bhi ho skte hai  
#query Operators  
- db.student.find({ marks : {$gt : 60}}) //greater than //$gte (greater than equal to)  
- db.student.find({city: {$in : ["Delhi", "Mumbai"] }}) //case sensitive  
- db.student.find( { $or : [{marks : {$gt : 75}},{ city : "Delhi"} ] })  
# UPDATE in db  
- db.student.updateOne( { city: "Mumbai"}, {$set: {marks: 76}}) //$set db.student.updateOne   
- db.student.updateMany( {city : "Delhi"},{ $set : { city : "New Delhi"} })  
- db.student.replaceOne({ name : "Aarav"}, {name : "Hemant", marks : 99, state : " Uttrakhand" })  
# Nesting  
- db.student.insertOne( {name: "farah", performance: {marks: 88, grade: "A"}} )  
- db.student.find({ "performance.marks" : 88})  
- db.student.deleteOne({age : 19})  
- db.student.deleteMany({ marks : { $lt: 90}})  
- db.student.deleteMany({}) sbko delete krdega  
- db.dropDatabase() //current db ko delete krdega

# MongoDB (Part2)  
# Mongoose - library hai jo mongo db aur odejs ke beechme connection banne mein help krti hai  
- its an odm (object data modelling) library  
-