7th March Class

Let's start with what is Javascript?

Even before we go there, I hope we all are aware of the fact that our entire course is going to be based on javascript. Every other runtime or framework that we are going to use in this backend development course will be built on top of javascript right?
As we had discussed in our second class that node.js is just a runtime with javascript at the heart of it. Similarly every other framework that we are going to use are based out of javascript. So it is definitely good to understand what javascript actually is.
Okay now that we know that javascript is at the heart of the entire course, let's start with a brief introduction about javascript.

Javascript can be defined as A HIGH LEVEL, OBJECT-ORIENTED, MULTI-PARADIGM PROGRAMMING LANGUAGE.

What essentially is a programming language?
A programming language is just a tool that allows us to write code to instruct the computer to do something.
Now of course that's our main goal of using javascript right?

Now the next term is HIGH-LEVEL - now what is a high level language?? Well a high level language is one where we don't have to worry about a lot of complex stuffs such as managing the computer's memory while it runs our program. So in javascript there are a lot of abstractions over these so called abstractions over all these small details that we don't want to worry about and this makes the language a lot easier to learn and to write.

OBJECT ORIENTED - Next if you take a look at the definition you will see that javascript is object oriented meaning that the language is mostly based on the concept of objects for storing most kind of data.
And of course we will learn about object oriented programming through out this course. But to talk about an analogy, but before we go there let us remember one fact that every object has its own attributes and behaviours,
if we were to consider us human beings we have some attributes or features right??

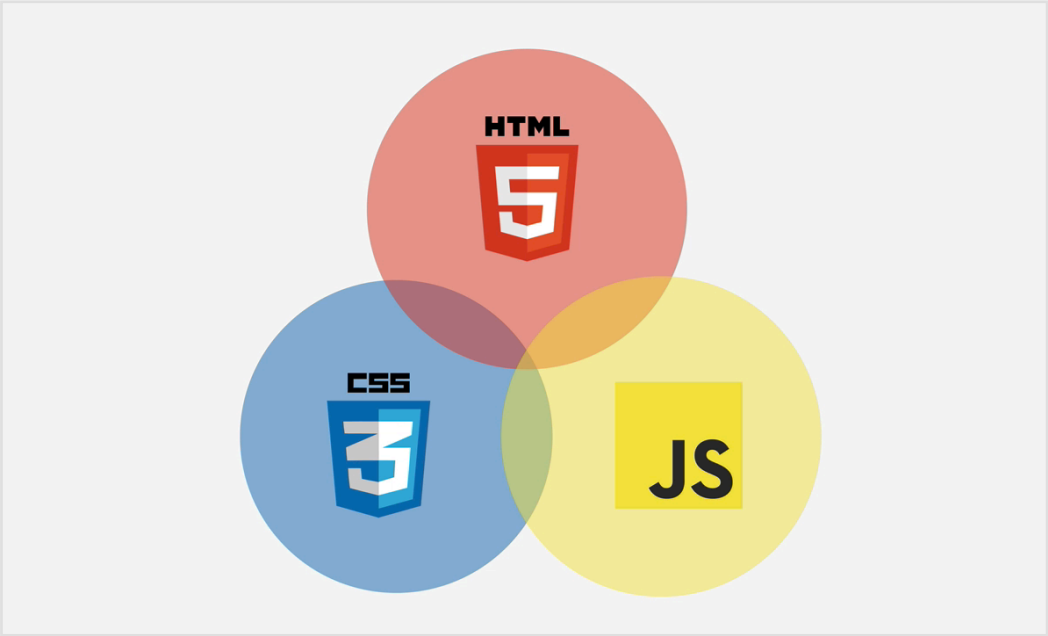
Like can you give me some examples of attributes or features of human beings?? - human beings have two eyes, two ears, two hands, two legs, 10 fingers etc. etc. If we talk about behaviours then what are the different behaviours that human beings exhibit? Human beings eat, human beings can walk, talk, analyse things, write etc. etc. right?? Similarly certain programming languages also tend to follow this practice of using objects in order to represent a specific group of data.

Finally, Javascript is a multi-paradigm language - meaning that it's so flexible and versatile, that we can use all kinds of different programming styles such as imperative and declarative programming, and these different styles are just different ways of structuring our code, again we will learn all about this through out the course.

This is just a very high level overview.

Now that we understand this much about javascript or kind of understand about javascript. Let us try to understand the role that it plays in web development.

And to answer that question - let's take a look at the three core technologies of the web.



Now these three technologies all work together to create beautiful interactive and dynamic web applications. Now the html is responsible for the content of the page.
So all the images, the text, the buttons and all the contents that we see on a web page is always written in HTML.

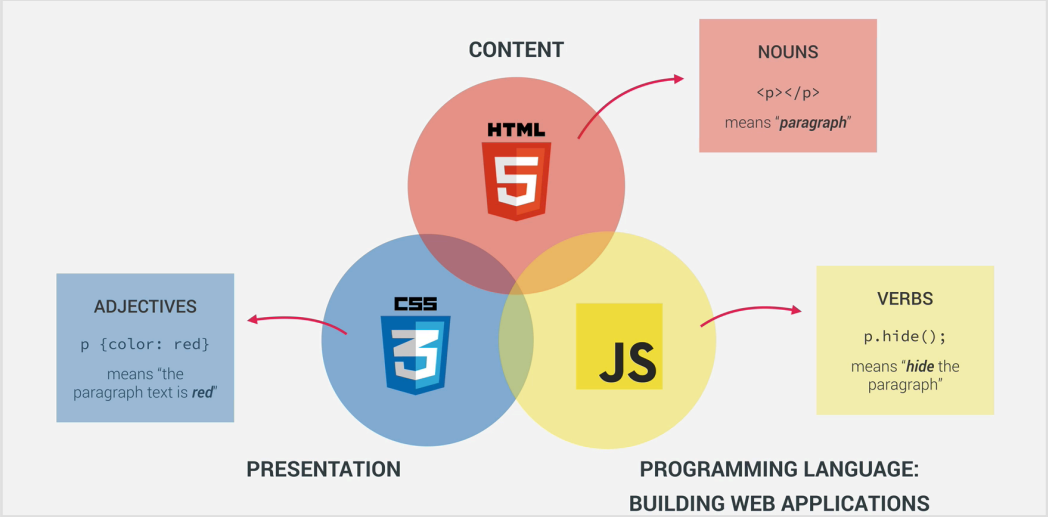
Then CSS is responsible for the presentation of the content. So basically for styling and laying out the elements of the web-page

And finally comes Javascript is the actual programming language of the internet and it allows developers to add interactive and dynamic effects to any webpage. We also use it to manipulate the content and load data from remote servers and really build entire applications in the browser which we then call web-applications.

Now we can also use the analogy of nouns, adjectives and verbs to make the

separation of these roles a bit easier to understand

HTML represents the nouns, eg. saying the P element is a paragraph and so paragraph is the noun here.
CSS is then the adjectives because it describes the noun. Like this piece of css saying the paragraph text is red and red is describing the noun
The javascript is, of course the verb like saying hide the paragraph. And so here we are doing something and so we have a verb.



Now since we are talking a lot about dynamic effects and web-applications, let's now take a real life example to make it even more clear what Javascript is capable of.
Let us open Linkedin and try to see what is happening behind the scenes.

So plain javascript helps us to add cool dynamic effects to pages. And in fact, help us build entire web-applications in the browser. Javascript is what makes modern web-development and really the whole modern web itself possible in the first place.

So as we said that there are so many cool libraries and frameworks such as react, angular, node, express that are totally built on javascript. So it is extremely important for us to be really good at javascript before learning and using any of these frameworks and not just jump into any framework by learning just 10 lines of javascript. So learning javascript properly is the single best investment that we can do in our software development career right now.

The javascript language and the web browser are two different things and it means that Javascript can also run outside web browsers for example it's possible to use JS on a web server using a very popular technology called Node.js which doesn't need any browser at all.

One last thing before we finish off our introduction on Javascript is that Javascript had a huge update to the language in 2015, which is known as ES2015 but most people also call it ES6 because it was released after ES5. And ES stands for ECMA Script.

Express

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.
It makes node powerful and developers life easier while it comes to developing the backend server of an application using node.js

NPM
You can consider as the database for all third party libraries. Now what do we mean by third party libraries.

For example we want to perform some operations for example let's say we want to connect to a database**,** now imagine if we had to write all the driver level code to be able to connect to that database, it would be what 1. Time Consuming, 2. There could be chances of error in the implementation or things that we would not have taken care of by mistake or because of lack of knowledge.

Also one more point to consider is that we are probably not the only folks in the entire world who would be trying to a mongoDB database right?? Most other folks would have already tried doing that. So what it means is that it is a pretty common operation to perform, not specific to only our application that we are building, so what happens is that such common operations that we has to be performed by every project which is trying to use that feature, has already been written and stored online somewhere so that people need not rewrite it every time they need to use it. So, these modules are directly uploaded to a repository online, such kind of a repository for node.js is the NPM also known as the Node Package Manager. Meaning, code for each such operation are uploaded online so that everyone can reference it from one place. So when someone feels that this is a feature that maybe required frequently and can be used by multiple teams who are trying to implement the same things in their own code, will publish this library in npm and the others can find it online from there.