**Javascript**

**? Difference between typescript javascript? Javascript typescript? js vs ts? ts vs js? js ts? ts js?**

Typescript is an object oriented programming language developed by Microsoft Corporation, whereas JavaScript is the programming language for the web.

For Example in Typescript we use the interfaces, enums, type keyword, strong types where as javasript is not

Javascript is a dynamic, loosely typed language .Type script a superset of javascript adds static typing for better

Development tooling and error catching. .js file is nothing but converted ts code.

**? Uses of Java script?**

Javascript is executed on the client-side as well as server-side also. There are a variety of Frontend Frameworks that you may study and utilize. However, if you want to use JavaScript on the backend, you'll need to learn NodeJS. It is currently the only JavaScript framework that may be used own the backend.

Javascript is used for frontend web development to make websites interactive . so that it runs in browsers, and enabling dynamic content, user interactions, and responsive interfaces . so that javascript creating a wide range of web related functionalities and applications.

### ? JavaScript and ECMA Script related?

ECMA Script is like rules and guidelines, while Javascript is a scripting language used for web development.

### ? What do you mean by JavaScript Design Patterns?

In JavaScript design patterns are repeatable approaches for errors that arise sometimes when building JavaScript browser applications.

We are having the 3 approaches for handle that.

1. Creational Design Pattern
2. Structural Design Pattern
3. Behavioral Design Pattern

**Creational Design Pattern:**The object generation mechanism is addressed by the JavaScript Creational Design Pattern.

**Structural Design Pattern:**The JavaScript Structural Design Pattern explains how the classes and objects we've generated

**Behavioral Design Pattern:**This design pattern highlights typical patterns of communication between objects in JavaScript.

**? What is HTML DOM? DOM?**

D OM stands for Document Object Model. DOM is a programming interface for HTML. When the browser tries to render an HTML document, it creates an object based on the HTML document called DOM HTML DOM is like a tool for web developer . it helps them organize and change thing on a web page using code like javascript. It’s how websites can do thing like updating content without reloading the whole page.

### ? What do you mean by BOM?

### BOM is nothing but a Browser Object Model is known as BOM. It allows users to interact with the browser. A browser's initial object is a window.

**For example:** The document, history, screen, navigator, location attributes are related to the browser object model

BOM in web terms stand for browser object model. It’s like a toolkit that helps computer programs like web browser.

Understand and interact with the thing on a webpage , such as windows and history. Objects deal’s with everything out side the page.

**? What is ECMAScript?**

ECMAScript is a Standard for scripting languages such as JavaScript, JScript, etc. It is a trademark scripting language specification. ECMA script is like a rule book for java script . it tells developer how java script should behave and what features it should have. So, when people talk about ECMA script, they’re basically talking about the set of rule that make sure java script work’s the same way everywhere.

**? What are new Concepts in ES6?**

ES6 and ECMA script brought some cool stuff to javascript . Those features are like

Let, const, arrow functions, default parameter, rest and spread operators, template literals, destructuring, class, import and export, promises,modules .

**? Difference between var let const? Let var const?**

**Var** is like a global scoped value which we can redeclared and reassigned the value.

**Let** is like a block scoped value which we cannot redeclared the value.

**Const :-** like let but you can’t change it after you set it . use it when the value shouldn’t be modified

**? What is the default parameters?**

Default parameters, we can use for to initialize a function with default values if any arguments are not supplied from the function call.

we assign default parameters in function itself while calling.

**? What is arrow function?**

Arrow functions are allows to create the functions in a cleaner way compared to regular functions. I mean normal function Arrow function in javascript is a faster way to write shorthand function. It’s like a quick way saying “hey here’s a simple function”. For example ,instead of a longer expression , you can use an arrow function.

**? What is Rest operator?**

Rest parameter to represent an indefinite number of arguments as an array. Actually we have to specify with 3 dots.

And always we have to pass the rest operator as last value. And rest operator contains the array of the values.

The rest operator (…) in javascript is like a collector for function arguments.

**? What is spread operator?**

Spread operator allows us to quickly copy all **or** part of an existing array

**Or** object into another array **Or** the object.

The spread operator (…)in javascript is like a magic wand for arrays or objects it helps you easily copy or combine their elements.

**? Diff between spread and rest? Spread vs rest?**

The **rest** operator (…) allows us to call a function with any number of arguments and then access those excess arguments as an array and also it’s allows the in DE structuring array or objects.

The **spread** operator which is useful for (…) allows us to expand an iterable like array into its individual elements

**? What is template literals?**

Template literals are literals delimited with back tick ( ` ) characters, allowing for multi-line strings with the embedded expressions.

Template literals in javascript are like a smart way to create string . instead of using quotes , you can use backticks ( `) and put variable directly inside with ${}`. It’s handly for making string that involve variable without lots of messy concatenation.

**? What is Destructuring?**

Destrucuring in a JavaScript that makes it possible to split the values from arrays, or properties from objects, into distinct variables.

Destructuring in javascript is like grabbing specific pieces from array or object more easily. It’s way of quickly picking out what you need from a bunch of data without writing lots of lines of code. Imagine it as getting exactly the toy you want from a toy box without going through each one.

**? What is Class?**

A class is a type of object template where we can defined the properties, methods, and constructor

**? What is constructor?**

A constructor in the class is a special method that creates and initializes an object of the class

**? What is super?**

Super is a kind of keyword where we used to add in constructor to define the parent class.

**? What is get set operators? Get and set? Getters and Setters? Getters Setters?**

Actually both are Accessor properties

**Get**: to define a getter method to get the property value.

**Set**: to define a setter method to set the property value.

? **What is Prototype**?

A prototype is a blueprint of an object. The prototype allows us to use properties and methods on an object even if the properties and methods do not exist on the current object.

**For example:** if we want to create the some method to the any data type then by using the prototype I can create that.

**Real time Example:** Yeah there is some string. In that string we have to particular key words are available or not that situation we used that.

**? What is import?**

Import is a keyword introduced in es6 as part of the javascript module which can import the variable or function or object from the exported file.

**? What is export?**

The export statement is used when creating JavaScript modules to export objects, functions, variables from the module so they can be used by other programs with the help of the import statements.

**? What is Export default?**

Default exports are useful to export only a single object, function, variable.

During the import, we can use any name to import.

**? What is import export? Import and export**

The export and import are the keywords used for exporting and importing one or more members in a module.

**? What is ES8 Features?**

Async and await. String padding, Object. Entries and Object. Value

**? What is Async and await? Async await?**

Async and await are like code markers and **Async** is makes a function to the return a Promise.

**Await** makes a function wait for a Promise.

Await :- it’s used inside an async function to pause execution until the promise is resolved .it make asynchoronous code look and behave more like synchronous code.

Async :-it’s keyword used to define a function as a asynchronous function return a promise implicitly.

async function rest(){ x = await 6+9; console.log("inside function") return x;}let res =rest()res.then(function(data){ console.log(data);})console.log(res);console.log("out of box");

**? What is pad Start?**

Pad Start method pads the current the string with another string with multiple times

Example: let text = "5"; text = text.**padStart** (4,"0"); output: 5000

Padstart is method in javascript that pads the current string with another string reaches a specified length. It is often used

To format string or align them in a certain way.

**? What is pad End?**

Pad Start method pads the current the string with another string with multiple times

Example: let text = "5"; text = text.**padEnd** (4,"0"); output: 0005

**? What is ES10 Features?**

* TrimStart() and TrimEnd()
* Symbol
* String. matchAll()
* Array. flat()

Trim Start () we can trim the value from the start position. And Trim End () we can trim the value of the last position.

**? What is symbol?**

It is a new data type introduced in the ES6 version of JavaScript. It is used to store an anonymous and unique value..

A symbol in javascript is a special kind of name that’s unique .It’s often used to prevent naming conflicts . Symbol are unique and can’t be changed

**? What are the new Concepts in ECMAScript or ECMAScript latest? New features of ECMAScript New? Latest features ECMAScript? ES 12? ES 2022?**

Yeah there are few concepts are introduced in the latest

**Replace all** for string method and **nullishequal** operator (?? ==) and new **logical assignment** operators (||,??, and &&) and added any method to the **promises**

**? Primitive data types?**

In javascript, Primitive data types can store only a single value.

Ex: null, number, undefined, string, Boolean, Object

**? Non primitive datatypes? Non primitive?**

Non primitive datatypes which is useful for to store multiple and complex values,

Ex: Array, Object

**? Difference between the Primitive data types non primitive datatypes?**

A **primitive** type has always a value, while non-primitive types can be null

And there is a difference between **primitives** and **non-primitive** is that primitives are **immutable** and non-primitive are **mutable**.

Here Mutable values are can be modified after creation. Immutable cannot be modified after creation.

**? What is Array?**

Array is nothing but a collection of the datatypes. Here datatype is like anything like number, Boolean, array like that.

An array is a data structure in programming that store a collection of elements, such as number or string .The element are

Identified by their index or position in the array,string from zero/ Array are versatile and widely used for organizing and manipulating sets of related data.

**? What is Object?**

The **object** type refers to a compound value where we can set the properties (named locations) that each holds their own values of any type.

In programming , an object collection of key-value’s pairs, where each key is a unique identifier for value . These number, string , or even other object’s . object’s are fundamental part of amany programming languages and are used to represent real-world entities and their attributes.

**? What is map in Javascript?**

A Map holds key value pairs where the map keys can be any datatype.

In javascript , a map is like a special container where you can keep pairs of thing . what’s coolis that the key-value pair’s can be anything , making it versatile for organizing and retrieving data in a structured way.

**? What is set in Javascript?**

A JavaScript Set is a collection of unique values. Each value can only occur once in a Set.

**? What is function?**

Function is nothing but a collection of statements which executes as a single thread when it called.

In javascript a function is a collection of statement’s so a function is a reusable block code that perform a specific task

.

**? What is call back function? Callback?**

A JavaScript callback is a function which is to be executed after another function has finished execution

I mean a callback function is a method that is sent as an input to another function

A callback function is like a helper function that you give to anaother function. The main action does it’s job, and when it’s done , it “callback” the helper function to finish some additional task’s.

**? IIFE (immediate invoke function)**

An IIFE (Immediately invoked Function Expression) is a way to execute functions immediately, as soon as they are created.

And method name signature is not required for the IIFE.

**Example**: **(***function(){*

*console.log(“printed”);*

*}***)**;

**? What is Closure?**

Closures are defined as inner functions that has access to variables and parameters of the outer function

**Advantages**:

They allow you to attach variables to an execution context.

Variables in closures can help you maintain a state.

A closure can be defined as a JavaScript feature in which the inner function has access to the outer function variable. In JavaScript, every time a closure is created with the creation of a function.The closure has three scope chains listed as follows:o Access to its own scope.o Access to the variables of the outer function.o Access to the global variables.

**? What is Execution context?**

Execution context is object-based. Execution context is an abstract concept that holds information about the environment where the current code is being executed.

### ? What are generator functions?

Generator functions are introduced in the ES6 version, generator functions are a special class of functions.  
  
**They can be stopped midway and then continue from where they had stopped.**

**Usually we have to declare the generator function by \* symbol**

**A generator function in programming is a special type of function in javascript that can be paused and resumed , It allows you to control the execution flow manually, yelding values one at a time. Generator function uses the yield keyword to produse a sequence of value , and they are often used for lazy evaluation or dealing with asynchronous task in amore readable manner.**

### ? What is recursion in a programming language?

Recursion is a technique to iterate over an operation by having a function call itself repeatedly until it arrives at a result.

**function** add(number) {

if (number <= 0) {

return 0;

} else {

return number + add(number - 1);

}

}

Add(3)

**? Call () Apply () Bind ()?**

In JavaScript, you can use call (), apply (), and bind () methods to couple a function with an object.

**? What is Call ()?**

Call is a kind of the **method takes arguments from the function.**

**? What is Apply ()?**

Apply is a kind of the method **that takes arguments as an array**

**? What is Bind ()?**

This method returns a new function, where the value of **“this”**keyword will be bound to the owner object, which is provided as a parameter.

const myfunction = function(){

console.log(this.name);

};

const context = { name:'lesa'};

const boundFunction = muFunction.bind(context);

boundFunction();

**? What is type of? Typeof?**

Type of in JavaScript is an operator used for type checking and returns the data type of the operand passed to it

**? What is hoisting? Hoisting?**

In JavaScript, Hoisting is the default behavior of moving all the declarations at the top of the scope before code execution.

Hositing is a behavior in javascript where varibe and ffunction declaration are moved to top of their containing means you can use a variable or function berfore it’s decalred in your code.

console.log(x);//undefined

var x=666;

console.log(x);//666

**? What is scope?**

Scope in JS determines the accessibility of variables and functions at various parts of one’s code.

Actually there are different types of scopes in JS:Global Scope, Local or Function Scope, Block Scope

In programming, scope defines the

**? What is global scope?**

Variables or functions declared in the global namespace have global scope.

For example: by using the var keyword we can declare the global scope.

**? What is Function Scope?**

Any variables or functions declared inside a function have, so i.e. function scope.

**? Block scope?**

Block scope is related to the variables declared using let and const. so for example if we declare the any reference by using the let or const i.e. block scoped

**? Why we use debugger?**

By using the debugger we can stop the execution of the code before moving on to the next line while debugging.

Debugging is the process of finding and fixing errors within a script. All modern browsers and most other environments support debugging tools – a special UI in developer tools that makes debugging much easier. It also allows us to trace the code step by step to see what exactly is going on.

**? Implicit conversation?**

Implicit conversation means we can do automatic conversation we from datatype to another datatype.

Implicit conversion is handled automatically by the compiler (the conversions of one datatype to another datatype).

**? String coercion?**

By using + operator we can convert the **integer** to **string**. So i.e. the String correction

coercion refers to the implicit type conversion that happens when JavaScript tries to perform an operation with values of different types. By using + operator we can convert the integer to string. So i.e. the String coercion.

**? Logical Operators?**

Logical operators are **used to determine the logic between variables or values.**

For example: and, or are the Logical operators

**? What is nullish coalescing operator (??)**

It is a logical operator that returns its right-hand side operand when its left-hand side operand is null or undefined

**? Is JavaScript a statically typed or a dynamically typed language?**

JavaScript is a dynamically typed language. In a dynamically typed language, the type of a variable is checked during **run time**in contrast to a statically typed language, where the type of a variable is checked during **compile time.**

**? What is strict mode? Strict mode?**

In JavaScript strict mode useful for restrict the variant like for example

Example:

1. Duplicate arguments are not allowed.
2. In strict mode, you won't be able to use the JavaScript keyword as a parameter or function name.
3. The 'use strict' keyword is used to define strict mode at the start of the script. Strict mode is supported by all browsers.
4. it will not be allowed to create global variables in 'Strict Mode’
5. semicolon should be available in every line of code

### ? What do you mean by Self Invoking Functions? Selfinvoke?

Yeah. Actually Without being requested, a self-invoking expression is automatically invoked (initiated). If a function expression is followed by parenthesis (), it will execute automatically. A function declaration cannot be invoked by itself.

Normally, we declare a function and call it, however, anonymous functions may be used to run a function automatically when it is described and will not be called again. And there is no name for these kinds of functions

**? What is Switch statement?**

The switch statement executes a block of code depending on different cases.

**? What are the differences between for...of and for...in statements? For in For of forin forof?**

Both for in and for of statements iterate over js data structures. The only difference is over what they iterate:

1. for..In iterates over all enumerable property keys of an object
2. for..of iterates over the values of an iterable object.

**? What is for in?**

Iterate (loop) over the properties of an object or array:

Here the code block inside the loop is executed once for each property.

**for** (let x in person) {

……..

}

**? What is for of?**

Iterate (loop) over the values of an array or string.

Here the code block inside the loop is executed once for each value.

**For** (let x of person) {

……..

}

**? What is break?**

The **break** statement breaks out of a switch or a loop.

In a **switch**, it breaks out of the switch block. This stops the execution of more code inside the switch.

**? What is Continue?**

The **continue** statement breaks one iteration (in the loop) if a specified condition occurs, and continues with the next iteration in the loop.

**? What is yield?**

The yield keyword pauses generator function execution which will returns an object with two properties, **“value”** which is the actual value and  **“done”**

So yield is kind of like return: you get something back. return x returns the value of x, but yield x returns a function, which gives you a method to iterate toward the next value.

**? What is regular expression? RegEx?**

A regular expression is a **pattern** of characters.

The pattern is used to do pattern-matching "search-and-replace" functions on text.

In JavaScript, a **Regular Expression** Object is a pattern with Properties and Methods.

### ? Exec () and test () methods in javascript?

test () and exec () are regular expression methods used in javascript.

We'll use exec () to search a string for a specific pattern, and if it finds it, it'll return the pattern directly; else, it'll return an 'empty' result.

We will use a test () to find a string for a specific pattern. It will return the Boolean value 'true' on finding the given text otherwise, it will return 'false'.

**let** text = "javascript"; let pattern = /e/;  
**let** result = pattern.test(text);

pattern.exec(text);

**? What is inline script?**

**? Diff between load script files in head and body?**

**Coming to the html Head:** A JavaScript function is placed inside the head section of an HTML page and the function is invoked when any action will fired.

**Coming to the html Body:** A JavaScript function is placed inside the body section of an HTML page and the function is invoked when an action will fired.

**? What is Defer keyword?**

The defer attribute in script tag refers the Boolean attribute.

Incase if specify the defer attribute in the script then thatscript is downloaded in parallel to parsing the page, and executed after the page has finished parsing.

**? What is Async in script tag?**

The Async attribute in the script tag is a Boolean attribute.

When its present that means it’s specifies that the script will be executed asynchronously as soon as it is available.

**? What is declare keyword?**

Declare is used to tell the compiler this thing (usually a variable) exists already, and therefore can be referenced by other code, also there is no need to compile this statement into any JavaScript.

**Example**: Moment is a kind of time plugin.so If I want to add the moment related methods into my component I will create the reference based on the declare keyword

### ? What are the types of errors in javascript?

**Logical errors, syntax errors and runtime errors**

**Syntax error: if we do the any Mistake or spelling problem while writing the code it appears**

**Logical Error: if the program is incorrect then Logical errors will appears.**

**Runtime Errors:** Errors that come due to misuse of the command inside the HTML language

**? What is this keyword?**

**“this” keyword refers to the object that the function is a property of.**

In case if you defined in the class it refers the object of the class.

**? What is Event bubbling?**

Event bubbling is a method of event propagation in the HTML DOM API when an event is in an element inside another element,

BY using the stopPropagation() we can stop the event bubbling.

**? What are the localstorage? Local storage**

Local storage is a property that allows JavaScript sites and apps to save key value pairs in a web browser

The local storage store data persist that ever when user close their browser tap or window.that data never disappeared even user close web browser and those data have no expiration time and browser never clear them automatically

**? What are the session storage? Sessionstroage**

Session Storage is a property that allows JavaScript sites and apps to save key value pairs in a web browser

Session storage allow to you to store the data in the browser in depends on the system memory and the data

Stored in browser until the browser is closed . in the other words closing the browser will clear the data

Storerd in session storage.

**? Difference between local storage and session storage?**

By session storage we can pass the data from one window to another window where local storage we cannot

The values related to session storage automatically delete when we close the browser window

And data in local storage don’t have any expiry time.

**? What is cookie?**

A cookie is the small amount information which we can store in the website. when user close website cookie automatically destroy.

**? What is Query string?**

A query string is the portion of a URL where data is passed to a web application. Here data would be key value pair

**? Difference between cookie and storage?**

We can read the information from cookie in the server. Whereas we can’t read the information from storage in the server.

**? Difference between === and ==?**

By using === we can compare the datatype along with value. Whereas == we cannot

**Example**: generally we use this kind of approach on undefined and null value

**? What is isNaN?**

By using isNaN we can get the Boolean value whether respective value is number or not

### ? What is NaN property in JavaScript?

NaN property represents the **“Not a Number”**value. It indicates a value that is not a legal number

**? Null Undefined? Null and Undefined? Null vs Undefined?**

**Null:** The NULL value is used to represent no value or no object. If variable is designed by the null means that there is no value inside of that.

**Undefined**: means if a variable is declared and not initialize with any value then that variable contains the undefined value.

### What is the use of Void(0)?

Void (0) is used to prevent the page from refreshing, and parameter “zero” is for passed the value while calling.

I mean Void (0) is used to call another method without refreshing the page.

The void operator evaluate given expression and return undefined.this used with void(0) means do nothing

Don’t reload and don’t navigate ,do not run any code.

### What is the use of the blur function?

Blur function is used to remove the focus from the specified object.

Blur event is used remove focus specified elelment .so example we create input field when focus

the input field it has change color and whenever you want to unfocus use this blur event it has lost the focus .

### ? Explain passed by value and passed by reference?

**In JavaScript, primitive data types are passed by value and non-primitive data types are passed by reference.**

**For example for the Passed by value: var x = 20 ; Var y = x**

**For Example for the passed by the referece by value. var x = { name: “xyz” } ; Var y = x;**

**? What is Asynchronous call and synchronous call? Async call vs sync call? Async sync? Async and sync**

Asynchronous call that means that the code allows the program to be executed parallel.

Asynchronous call non-blocking task so the execution of one task is not depends on another task

It can run simultaneously , synchronous call is blocking task so execution of each task or operation completely

Depends on another one it mean all task executive at a time.

**? What is synchronous call back? Sync?**

Synchronous call, its block the further execution of the remaining code until the code finishes the current one.

**?** **Difference between Asynchronous call and Synchronous call?**

Asynchronous code allows the program to be executed immediately

Where the synchronous code will block further execution of the remaining code until it finishes the current one.

**? What is inheritance? Inheritance?**

Inheritance is nothing but a relationship or a communication between to the two objects

Example: We can make communication between the class and multi interface by the inheritance concept or we can make the communication from one class to another class

**? What is babel?**

Babel is a JavaScript transpolar to convert ECMAScript 2015+ code into a backwards compatible version of JavaScript in current and older browsers or environments. Some of the main features are listed below,

1. Transform syntax
2. Polyfill features that are missing in our target environment (using @babel/polyfill)
3. Source code transformations (or code mods)

babel is javascript transpiler that covert edge javascript es6 into plain old es5 javascript that can run in any browser even in the old one.

**? Window.onload?**

By default, it is fired when the entire page loads, **including** its content (images, CSS, scripts, etc.).

**? Window.Unload?**

The **on unload** event occurs once a page has **unloaded** (or the browser **window** has been closed).

**? OndocumentReady?**

OnDocumentReady loads the code just after the DOM is loaded. This allows early manipulation of the code.

**? Document.onload?**

It is called when the DOM is ready which can be **prior** to images and other external content is loaded.

### ? Difference between a web garden and a web farm?

Both web-garden and web-farm are web hosting systems. The only difference is that web-garden is a setup that includes many processors in a single server. At the same time, web-farm is a larger setup that uses more than one server.

**? What are Screen objects?**

Actually Screen objects are used to read the information from the client’s screen. The properties of screen objects are –

* **Avail Height**: Gives the height of the client’s screen
* **Avail Width**: Gives the width of the client’s screen
* **Color Depth**: Gives the bit depth of images on the client’s screen
* **Height**: Gives the total height of the client’s screen, including the taskbar
* **Width**: Gives the total width of the client’s screen, including the taskbar

### ? What is unescape() and escape() functions?

The **escape** () function is responsible for coding a string to transfer the information from one computer to the other across a network.

**Example:** <script>

document.write(escape("Hello? How are you!"));

</script>

The **unescape**() function is very important as it decodes the coded string.

**Example:** <script>

document.write(unescape("Hello%3F%20How%20are%20you%21"));

</script>

### ? Decode URI() and encode URI()?

**Encode URl** () is used to convert URL into their hex coding. And **Decode URI** () is used to convert the encoded URL back to normal.

**Example:**

<script>

var uri="my test.asp?name=ståle&car=saab";

document.write(encodeURI(uri)+ "<br>");

document.write(decodeURI(uri));

</script>

**? InnerText and innerHTML?**

**Inner Text** we can insert the html tags where as we can insert the html tags in **inner html**

**Inner Text** returns text without an inner element tag. **Inner html** returns a tag with an inner element tag.

**? What is typescript?**

Typescript is the open source and super set of the JavaScript framework which primarily provides the interfaces, enums, strong types, tuples.

**? Functional typescript? Functional programming?**

Typescript is not a purely functional programming language but offers a lot of concepts that are in line with functional programming languages

**For example**: Higher order functions, Currying

**? Higher order functions in javascript?**

**Higher order functions that operates the other functions, either by taking them as arguments or by returning them, are called higher order functions**

### ? What is currying in JavaScript?

### **Currying is kind of an advanced technique to transform a function of arguments n, to n functions of one or fewer arguments.**

### **currying take a function that receives more than one parameter and breaks it into a series of unary(one parameter) functions**

**function first () {**

**return second(b){**

**return a + b;**

**}**

**}**

**first(3)(4)**

**? Typescript features? Advantages?**

* **Cross-Platform**: The Type Script compiler can be installed on any operating system such as Windows, macOS, and Linux.
* **ES6 Features**: Type Script includes most features of planned ECMAScript 2015 ([ES6](https://www.educative.io/blog/javascript-es6-tutorial-a-complete-crash-course)) such as Arrow functions.
* **Object-Oriented Language**: Type Script provides all the standard [OOP](https://www.educative.io/blog/javascript-oop-primer) features like classes, interfaces, and modules.
* **Static Type-Checking**: Type Script uses static typing and helps type checking at compile time. Thus, you can find compile-time errors while writing the code without running the script.
* **Optional Static Typing**: Type Script also allows optional static typing in case you are used to the dynamic typing of JavaScript.
* **DOM Manipulation**: You can use Type Script to manipulate the DOM for adding or removing client-side web page elements

**? Built in data types in typescript?**

Number, string, Boolean, void

**? What is strongly typed? strongtype? styronglytype?**

Strongly typed means variables and other data structures can be declared to be of a specific type

**? Why we have to use the typescript in the Angular? Uses of typescript?**

Type script supports the oops concepts like it supports interfaces. So by doing that we can make the inheritance, strong types, access modifiers, tuples, and enums. So that we use the typescript.

**? What is Access Modifiers?**

An access modifier is a keyword that we can apply to a member of a class to control its access from the outside.

* **Private** and **public** and protected are access modifiers.
* If you declare the variable by **private** you can’t access that variable outside of the function or class whereas by using the **public** access modifiers you can access entire on the like script.
* And a **protected** member cannot be accessed outside of its containing class

**? Extends and implements? Extends implements?**

Both key words are useful for build the relationship between the objects like

By using the Extends we can build the communication between one classes to another class

Whereas by implements we can build the communication between one class to another interface

Appmodule -> entire the application

Appcomponetn -> entire the

**? What are the concepts in typescript?**

Interface, strongly typed, generics, enums, tuple

**? What is Enum?**

An enum is a special "class" that represents a group of constants.

An enum is like a list of specific things you can name . It helps make code easier to understand by giving names to Differentpossibilities.

**? What is interface?**

An **interface** is a kind of a contract that a class should conform to.

By using the **interface** we can make the relationship to the class by providing the methods from the interface

And we can make communication between the class and multi interface by the inheritance concept

**? What are modules in typescript?**

Modules in Type Script are a collection of related variables, functions, classes, and interfaces.

You can think about the modules like containers that contain everything needed to execute a task. Modules can be imported to easily to share the code.

**? What is namespace?**

The namespace is a way which is used for **logical grouping** of functionalities. It encapsulates the features and objects that share common relationships. It allows us to organize our code in a much cleaner way.

We can create the namespace name by the namespace keyword.

**? What is JSX in typescript?**

JSX is an embeddable XML-like syntax that allows you to create [HTML](https://www.educative.io/blog/html-beginners-tutorial-build-from-scratch).

JSX in typescript is a way to write more readable and expressive code for building user interfaces , especially in react

It let you describe how your app should look using a syntax that combines javascrippt and xml-like tags. Typescript adds

The ability to catch certain type of errors in your code while you’re writing it.

**? What does the Omit type do?**

Omit is a kind of keyword in typescript which is a form of utility type, which easier the common type transformations

**? UnKnown?**

Unknown is kind of datatype in typescript when if you don’t know the type to expect from outside. So that situation we use the unknown type

**? Any?**

Any is a one kind of datatype where we can assign the any value.

**? Unknown any? Unknown and any?**

You can’t assign the unknown value to the known data type. It will give the error. Whereas any we can do.

**? What is .map file?**

A map file is a source map that shows how the original Type Script code was interpreted into usable JavaScript code

A map file is like a guide that help developer understand and fix issues in their web code . It connect the transformed , harder to read code that runs on the website with the original, easier to understand code that developer write. It’s like having a translation between the two to make debugging smoother.

**? What is generics?**

Generics allow creating 'type variables' which can be used to create classes, functions and type that don't need to explicitly defined.

**?** **What is tuple?**

In typescript a tuple is a typed array with a pre-defined length and types for each index.

Tuples are great because they allow each element in the array to be a known type of value.

**? What is map?**

Map is a collection of elements where each element is stored as a Key, value pair.

Map object , it can hold both the objects and primitive values in the format of either key or value and when we iterate over the map object then t it returns the key and value pair in the same order as inserted

**What is filter?**

The JavaScript Array filter () Method is used to create a new array from a given array consisting of only those elements from the given array

Certainly! In simple words, in JavaScript, "filter" is a method that helps you create a new list (array) from an existing one, including only the items that meet a certain condition. It's like a sieve that lets through only the elements you want based on some rule you specify. The original list remains unchanged, and you get a fresh list with only the items you need.

**? Diff between map and Filter?**

**Map**: returns an array of pieces of information from the original array. In the callback function, return the data you wish to be part of the new array.

**Filter**: returns a subset of the original array based on custom criteria.

**What is splice and slice?**

The slice () method can be used to create a copy of an array or return a portion of an array.

And slice () method does not alter the original array but instead creates a shallow copy. Unlike the slice () method, the splice () method will change the contents of the original array

**Angular**

**? What is angular?**

Angular is the kind of the framework for develops the responsive Single page applications.

**? Why were client side frameworks like angular has introduced? Clientside?**

Actually it was designed by Google to enable developers to create dynamic web applications

**? Difference between the angularjs and angular?**

* Angular uses typescript framework and AngularJS uses the javascript framework
* Angular follows the components for designing and AngularJS follows the MVC pattern.
* Angular does provide mobile support and AngularJS is not
* Performance wise Angular is better than AngularJS for handles the large applications
* Angular uses the Hierarchical dependency injection system where angular js is not

**? Hierarchical dependency?**

Actually Hierarchical dependency we use in services. Like if we register the service in app module section then instance of that service would available in entire the module

In Angular, a hierarchical dependency refers to the way components, directives, and services are organized and injected into each other within the application's structure. Angular uses a hierarchical injection system, where dependencies are injected based on the hierarchical relationship between components and services.

**? What is angular expressions?**

Yeah. An expression which can be written inside of the HTML i.e. the Angular.

**? What is advantage of the angular?**

An Angular framework makes it easy to develop web applications. Combining dependency injection, declarative templates, and integrated best practices, it solves almost all the challenges when creating a web app.

**? What is disadvantages of the angular?**

One of the major drawbacks to using Angular is the platform's limited SEO options and poor accessibility for search engine

And another thing is a large portion where we can’t learn quickly.

**? What is single page application? (SPA)?**

Single page application is an application that executes in a web browser and does not require page reloading during in use.

**? How to execute the angular or Load angular Project? Load angular? Angular life?**

* When we press ng serve command through terminal command prompt

Like

* All files are compiled by angular CLI
* Few JavaScript files are added into runtime which cause show the output of angular Application. Among them main.js file which is nothing but main file contains the Project main code.

**? Execution files step by step in angular project? Execution angular?**

**Angular flow? how to start angular application?**

There are 3 kind of files involved while angular project be loaded into the respective environment. Those are Main. ts, AppModule.ts and appcompnent.ts file

Coming to the

1. **Main.ts file :** which is used for register the start up module
2. **Appmodule.ts file:** which is used for register the app component as a start up component
3. **Appcomponent.ts file:** which is used for as a default component of the project

Like All three files are generated automatically with code proper structure when we create the new Project through command prompt

Actually resolver file

**? Main.ts file? :** which is used for register the bootstrap module in the angular application

Package.json : where I can see all the packages also .

Angular.json : where I can set tup the angular realted project .

**? What is AppModule.ts?**

Appmodule is the main module of the application which is used for register the app component as a bootstrap component

AppModule.ts is like the manager of your Angular app.

Decorator (@NgModule):

It uses a special tag @NgModule to tell Angular about your app's features.

Root Component:

It points to the main thing your app will show first, like the home page.

Building Blocks:

It lists all the different pieces (components, directives, pipes) your app is made of.

Dependencies (Imports):

It mentions if your app needs help from other modules. It's like saying, "I might need tools from this other room."

Shared Services:

It says if there are any helpers (services) available for everyone in your app.

So, think of AppModule.ts as the boss, telling Angular how to put together and run your whole application. It's like giving instructions to set up a shop, saying what products to sell, who's in charge, and what tools are needed.

**? What is <app root> element in index.html? app root approot**

<app root> is an element which can understand by angular and used for show the output of the app component in web application

In simple term the root element in the index.html file is like the main container for your webpage.it’s represented by the <html>tag and olds everything together, including the head(information about page) and body(content of the page).

**? What is Component? What is component? Componant?**

Component is a place where we implement a business logic for showing the output in the web application.

Component is a combination of HTML file (.html) + Typescript file (.ts) + Spec file (.spec.ts) + CSS file(.css)

**Did you create the custom component? Did you create that?**

Yeah I have created the multiple custom components in my application by executing the

ng space g space c space component name for create the component

Actually Custom Component is a place where we implement a business logic for showing the output in the web application.

Custom Component is a combination of HTML file (.html) + Typescript file (.ts) + Spec file (.spec.ts) + CSS file(.css)

**? How to create the basic Component manually? Create component, configure component**

Actually We use @Component decorator for develops the component in a typescript file

* We implements the Html Content in Template (.html)
* We implements the Business Logic under Typescript code (.ts)
* We implements the styles under the.css
* We implements the test cases under the spec files

**? How to register the component in the project? Register component**

App.module.ts is a place for declare the components.

Below are steps for declare the components

1. import the component

Declare the component name in declaration array of the @NgModule decorator

**? How to communicate the one component to another component? Component communication?**

Yeah we can make the relationship by using the @input and @output decorators. Here in @output decorator with the help of event emitter we can pass the data to another component.

And even we can communicate via services also.

**? How to communicate data between two sibling components? How we can share the data via services?**

We can create the any observable like subject. So based on the subject we can make the relationship between the 2 components

Like subject contains the two methods. Like next () and subscribe methods.

­

**? What is ngmodule? Where we use in Application?**

ngModule is a normal typescript class which is decorated by @NgModule

Module is a place we declare the components, imports the module, provide the services and also set the start-up component of the application.

Sure,think of an ngmodule in angular as a way to group and organize related pieces of web application, like component,services and more .It’s like putting similar things together in a box to keep your code neat and mangeable.

**? Importance of the ngModule?**

By default AngularCLI won’t scahostlisthern all the files in the Project. So if we register the things under in the @NgModule section then angular cli knows the register files for further process.

Here things are nothing but components, pipes, directives, services, modules.

**? What is data binding?**

Databinding is nothing but a communication i.e between template and Business Logic

Uses: dynamic content should be transfer between to the template and business Logic

**? How many types of Data binding? What are they?**

We can do Data Binding are 4 ways

1. String interpolation or string data binding
2. Property Databinding
3. Event Databinding
4. Two way Databinding

**? What is String Interpolation?**

If we want to show the string value in html through the typescript variable then we can use the string interpolation.

**? What is Property Binding? []**

If we assign the any value from.ts file through the property of the html then i.e. Property Binding.

**? What is Event Data Binding?**

If we assign the event from.html file through the function of the typescript code i.e. event databinding

**? Two way Databinding:**

Two way data binding is like a communication which deals in both directions like outputting the data and React to events then i.e. nothing but a two way databinding.

Like combination of property data binding and event databinding

Ex: ngModel.

**? Angular page lifecycle hooks? Life cycle hooks ?**

Angular component having the different page life cycles

Like ngOnchanges , ngOninit , ngDocheck, ngAfterContentInit , ngAfterContentchecked , ngAfterViewinit , ngAfterViewchecked, ngDestroy

**ngOnchanges**: lifecycle hook that is called when any data-bound property of a directive changes.

**ngOninit**: whenever you want to execute code when the component is FIRST initialized.

**ngDoCheck**():is called whenever change detection is run

**ngAfterViewInit**: A callback method that is invoked immediately after Angular has completed initialization of a component's of the view.

**ngAfterContentInit**: a kind of a call back function that is invoked immediately after Angular has completed initialization of all of the content of the templeate.

**ngDestory**: The ngDestroy is called in a component's lifecycle just before the instance of the component is finally destroyed. It is the perfect place to clean the component.

**? Differece between ngOnint and constructor?**

ngOninit is the pagecycle hook of the angular. Constructor is the special method of the class.

ngOninit we can’t pass the parameter. Constructor we can pass the parameters

**? What is decorator? And how many types of decorators we have?**

Decorators are like a design pattern that is used to separate modification or decoration of a class without modifying the original source code

**Class Decorators**. Like where we can apply to the **@component @pipe @directive**

**Property Decorators**. like @hostbinding

**Method Decorators**, like @hostlistner

**? What is event emitter?**

Event emitter which is use ful emit the events in components. I mean we can emit the data from one function to another function. By default event emitter can emit any kind of value.

**? What is Directive?**

Directives are classes that add new behavior or modify the existing behavior to the elements in the template. Basically directives are used to manipulate the DOM,

**? How many types of Directives in angular?**

1. **Structural Directives** Like ngFor, ngIf
2. **Attribute Directives** Like ngstyle, ngclass
3. **Custom Directives**

**? What is structural directives?**

Structural directives are directives which change the DOM layout by adding and removing DOM elements.

**? \*ngIf?:** A structural directive that conditionally includes a template based on the value of an expression

**? ngFor?** ngFor is a structural directive. Used to print the multiple values I mean array in the application.

**? What is attribute directives? Property directives?**

The attribute directive changes the appearance or behavior of a DOM element

**? ngStyle?** A attribute directive that conditionally includes a dynamic style based on the value of an expression

**? ngClass?** A attribute directive that conditionally includes a dynamic class on the value of an expression

**? What is User defined directives? Custom directives?**

Actually custom directive in Angular is a user-defined directive that provides users to use desired functions to extend HTML functionality.

Ng g d directive name, command useful for creates the custom directive.

**Real time Example of custom directive:**

* 1. There is some situation that needs to add only the company related email.
  2. So that situation I have created the custom directive.
  3. I mean we need to show validation the message incase if user should enters the his office email ID.

2. There is few textboxes users should not enter the special characters. So for that created the directives.

**? How to identify the angular cli about directive?**

Based on the @Directive decorator which is imported from @angular/core package

**? What is command for create the directive?**

ng g d <directiveName>

**? What is ngModel?**

The ngmodel directive which is useful for binds the value to the HTML controls (input, select, textarea) to application data.

**? What is ngmodelChange?**

NgModelChange is an Angular specific event, which we can use to listen for changes to the user input

**? How to communicate one component to another component?**

We can approach by 2 ways

1. Custom property databinding. ( @input)
2. Custom event databinding. (@output)

**? What is Custom property Data binding? (Note: Before see the definition of here please check Databinding Concepts) custom property?**

Custom Data binding is the process of passing the data from one component to another component via component property. For example @input we can do

**? What is Custom event Data binding?**

Custom Data binding is the process of passing the data from one component to another component via component function with the help of the event emitter

**? @input?**

@input is a kind of decorator where we use in the custom property databinding

Which can pass the results from one component to another component.

**? @output?**

@ouput decarotor we used in the situation of Custom Data binding.

So Custom Data binding is the process of passing the data from one component to another component via component function with the help of the event emitter

**? What is template reference variable?**

The template reference variable is a variable which contains the reference of the current element.

**? What is Local reference variable?**

The template reference variable is a variable which contains the reference of the current element.

**? What is @ViewChild?**

View Child is a Property Decorator which provides a reference to element or component or

Templates to define the view. We have to declare ViewChild with ‘@’ decorator.

**? What is @ContentChild?**

Content Child is a Property Decorator which provides a reference to element or component or templates to define content which is in component.

**? What is ngContent?**

ng content in angular is useful for inject the content into the component.

**? What is ngtemplate?**

ngTemplate is a template which can renders based on condition in DOM. If condition is true then it can renders in the DOM or it cannot

**? What is Host element?**

To turn an element from Angular component into something rendered in the DOM you have to associate an Angular component with a DOM element. We call such elements are host elements.

**? What is Host Binding?@hostbinding**

Host binding is useful for set a property value on the host element.

We declares Host binding with @HostBinding decorator in angular

**Example**: @HostBinding(‘style.backgroundcolor’) BGColor:string

**?What is Host Listener? @hostlistner**

Host Listener is useful for set an event on the host element

We declares Host binding with @HostListner decorator in angular

**Example**: @HostListner(‘’) mouseover() { // Code implementaion }

**? What is difference between Component and directive? Component directive**

* 1. A Component contains the template whereas directive does not contains the template.

I mean a Component can have the html code. Where directive don’t have the template.

1. Component can have the ngafterviewinit and ngafterconteintint page lifecycle hooks. Whereas directive do’t have the ngafterviewinit and ngafterconteintint pagelifecycle hooks

**? What is renderer2?**

Actually whenever we are working with directive it’s not good practice that directly render the host element because it might be gives errors in case directives wouldn’t load proper way in HTML DOM.

So that we use the renderer2.

**? What is Pipe? What is Custom Pipe?**

Pipes is a class which is to use in template expressions to accept an input value and return a data transformed value.

**Examples:** toUpperCase, toLowerCase, DatePipe

**What is Date Pipe?**

Date Pipe in angular handles the date format.

So , take example

**What is Async Pipe?**

The async pipe subscribes to an Observable or Promise and returns the latest value it has emitted.

When the new value is emitted, then async pipe updated the component to be checked for changes.

**? What is Pure Pipe and impure pipe?**

A pure pipe is only called when Angular detects a change in the value or the parameters passed to a pipe.

And An impure pipe is called for every change detection cycle no matter whether the value or parameter(s) changes.

**? Custom pipe real time example?**

Ng space g space p space with pipename we can generate the custom pipe.

1. So there is a big string value which we have to display in the table. So for the trim of that string we use the pipe

2. We have to filter the data based on the class status like scheduled or completed or cancelled or overdue.

so that situation also we use the pipes

**? What is Change Detection?**

Change detection is a kind of process like

Angular checks to see whether out application state has changed or not

and monitors the if any DOM needs to be updated.

**? Change detection onpush or default?**

On Push means that the change detector's mode will be set to Check Once during hydration.

Default means that the change detector's mode will be set to Check Always during hydration

**? What is async pipe?**

The async pipe subscribes to an Observable or Promise and returns the latest value it has emitted

**? What is routing?**

Routing in Angular allows the users to create a single page application with multiple views and allows navigation between them

**? What is lazy loading?**

Lazy loading is a technique in Angular that allows you to load components **asynchronously** when a specific route is activated.

Lazy loading is like loading only what you need, when you need it.

Normal Loading (Eager Loading):

Loads everything when you open the app.

Lazy Loading:

Loads only what's necessary at first.

Loads more stuff as you use the app.

Benefits:

Faster Start: App opens quicker.

Uses Less Data: Only gets what it needs.

Example in Angular:

Normal: Loads all parts at the start.

Lazy: Loads parts when you go to them.

Lazy load which loads the Angular modules and we have to use the load Children instead of component property.

So that In the load children property we have to configure the routes

**For example if you take the Advantages**:

Based on the lazy loading we can achieve the application related performance.

And based route activation we can load the particular module

**? What is eager loading?**

To apply Eager Loading strategy, at first, we need to **register all components that will be eagerly** loaded in the app routing.module.ts with forRoot strategy

**? How did you configure the Router?**

We have created the app.router.ts file and then we have divided the routing pages based on the modules. so each routing module its having the different components.

**Difference between lazy loading and early loading?**

**In Early loading** > its Load all components at a time.

**In Lazy Loading** > its based on the requirement we have to Load a specific module at a time.

**What is shared module? Sharedmodule?**

Creating shared modules **allows you to organize and streamline our code**. You can put commonly used directives, pipes, and components into one module.

**What is Core module? Core module**

The core module usually contains components that are used once in an Angular application, such as a navigation bar, loader, footer, etc.

**How to setup the Routes in application module? Setup routes**

We need to import the routing module from @angular/router as above snapshot. and then we need to configure all routes into the Router Module

**What is Router outlet? Route routet?**

Router Outlet is an Angular directive from the router library that is used to insert the component matched by routes to be displayed on the screen.

Roter-outlet in angular as a placeholder in your html where the content of different view or components will be displayed based on current route.It’s like slot on ypur website that dynamically show the content related to the chose route making your web app change and update without refreshing the whole page.

**What is Nature of anchor tag link? (<a></a>) anchortag**

It refreshes the entire the page when you are navigating to another web page. But its not a good practice.

**What is Router Link? routerlink**

Router Link is a directive for navigating to a different route declaratively.

Routerlink is like a special lik in angular that you put in your website,kind of like clicking a link to go to a different page.

**What is difference between navigating method and router Link?**

In Router Link we pass the absolute path and navigate path we pass the relative path.

**What is Activated Route:**

Activate Route Which is useful to inject the activate route in the routers section. So, for the component if u loaded this activated route is will be our router.

**? What is route?**

A Route is kind of service that is provided to each route component that contains route specific information such as route parameters, static data, resolve data, global query params and the global fragment.

**? How to handle the route params of the router?**

By doing the activate route params subscription, we can fetch the params

**? What are the Route guards? And how many types of in Route guards? Router guard**

The Angular router’s navigation guards allow to grant or remove access to the certain parts of

the navigation.

Route Guards are 4 types in angular

1. CanActivate
2. CanDeActivate
3. CanLoad
4. CanActivateChild

**So coming to the CanActivate:** Which allows the user to access the particular component

**CanDeActivate:** Which trigger whenever user leaves the component.

**CanActiveChild**: Which allows the user to access the children of the particular component

**CanLoad:** Which allows the user to a module that lazy loaded

**? What is can activate?**

It’s a kind of interface that a class can implement to be a guard deciding if a route can be activated. If the expected condition would be true then it will navigate to the required route

**? What is can deactivate?**

Actually this Angular CanDeactivate guard is called, whenever we navigate away from the route before the current component gets deactivated. I mean it would be trigger whenever user leaves the component.

**? What is can load?**

The Can Load Guard prevents the loading of the Lazy Loaded Module. We generally use this guard when we do not want to unauthorized user to navigate to any of the routes of the module and also stop then even see the source code of the module.

**? What is canActiveChild?**

Interface that a class can implement to be a guard deciding if a **child route can** be activated. If condition gives the return true value then it navigate to the required route path.

**Difference between forroot and forchild?**

**forRoot** creates a module that contains all the directives, the given routes, and the router service itself.

**forChild** creates a module that contains all the directives and the given routes, but does not include the router service. It registers the routers and uses the router service created at the root level.

**? Router resolve and router data?**

**Resolve**: in router which helps us to pass the data dynamically via router configuration.

**Data**: pass the any information statically via router configuration

**What is Wild Route path and how to configure it in the router?**

The wildcard Route which is basically handle the invalid URLS of the application. We can do by \*\*

**What is authentication?**

Authentication is a process of gives the security.

**What is authorization?**

Authorization is the process of giving permission to the user to access certain resource in the system.

**? How to handle the authentication and autherization?**

We can do by using Route Guards

Or

In order to build authentication, on the client, we need to build the login page and on the server, we should build an API Endpoint to validate the user. When the user clicks on the login button, our Angular app calls this API Endpoint and passes the username and password.

**? What is router module? routermodule?**

Routermodule is a place where we have to create all routes related the module.

**What is Service?**

A service in angular is nothing but typescript class which contains the centralized logic for business logic.

**Use of the Service:**

Service in the angular can act as a single instance in entire the application. In order that we can avoid the duplication of the code which contains in different components.

**What is @injectable?**

@injectable is a decorator which useful for create the service in angular

**Where I can provide the service?**

We can provide the service In the provider section of the appmodule

**? What is dependency injection and How to apply the dependency injection? Dependency**

Dependency is a kind of approach which a class requests dependencies from respective source like services or routers

like that rather than creating them in the component.

For example if you take constructors in the components where we can declare the routers, services, pipes.

So that we can say here the component is depends on the these routers, services

**Is it necessary that we have to declare @inject to every angular service?**

Not need. It’s useful for whenever if we need pass any dependency parameter through constructor.

**What is Hierarchical Injector and execution flow of the service?**

If we register the service in appmodule section then instance of that service would available in entire the module. So no need to create the instance of that service to child of components again.

**What is Cross-component Communication?**

A component communicate to another component via service is nothing but a cross-component communication

**Handle Webforms in 2 Ways:**

In angular we can handle the Web form in 2 ways

1. Template Driven approach
2. Reactive Approach

**? What is Template Driven Approach or TDA?**

In Template driven approach Angular takes the structure of the form object from the DOM.

**? What is Reactive Approach?**

In Reactive Approach we have to define the Form object in Typescript Code and setup the HTML code for the typescript form object. Then we have to manually sync the Typescript and HTML Code.

**? What is FormBuilder? Form builder?**

The FormBuilder is the class that is used to create both FormGroups and FormControls.

**? Diff b/w FormBuilder and formgroup?**

In Angular, a reactive form is a FormGroup that is made up of FormControls.

FormGroup is like container for form stuff and form builder is helper that makes it easier to create and mange the form stuff inside the container in angular.

**? How to create the custom validator?**

We can custom validator by 2 ways template driven approach and reactive approach.

For reactive approach we have to prepare the function and configure the control and template approach we have to prepare the directive

**? How to createt the custome validator in template driven approach?**

By directive

**? How to create the custom validator in reactive approach?**

By function

**? How to get the form control value and form object value?**

ngModel and ngForm

**? What is use of apply the validator to the form control?**

We can validate the ng valid and ng invalid css to the form control

Or

Angular form validation is an integral technical process that verifies if any input provided by a user into a web-form is correct and complete. You can manage validation in a template-driven approach or with Angular reactive forms

**? What are the considerations we have to take while doing the reactive approach?**

Form Group : Group contains the form controls

Form Control : A single Form Control

Form Array : Array of the Form Controls or Array of the Form Groups.

Validations : For showing the validation messages

**? What is Form Array and how to handle it? Formarray?**

Form Array is nothing but array of the form controls. We can handle by Form Array class.

**How to track the status changes or value changes of the form?**

By doing the subscribe of form status Changes or value Changes

**? What is form builder?**

The FormBuilder is the class that is used to create both FormGroups and FormControls

A form builder refers to a service provided by the Angular Forms module (part of the Angular framework) that helps in the creation and management of forms in Angular applications.

**? What is Form Group?**

By using FormGroup we can make the formControls and we can apply the validations with in that

• Represents a group of form controls.

• Useful for organizing related controls together.

• Tracks the values and validation status of its child controls.

**What is Form Control?**

Form control is a class that can hold both the data values and the validation information of any form element.

• Represents an individual form control, such as an input field.

• Manages the value and validation status of the form control.

**What is validator?**

A validator is a function that processes a Form Control or collection of controls and returns an error

**ng Pristine**: Class adds to the control when no update to the control

**ng dirty**: Class adds to the control when update the control

**ng touched**:Class adds to the control after touched

**ng untouched:** Class adds to the control when untouched

**ng valid:** Class adds to the control when valid

**ng invalid:** Class adds to the control when invalid

**? How to write the custom validator?**

For that we have to create the own specific function with return object. And then we have configure to the respective form element.

**? What is ngZone?**

The NgZone service provides a run() method that allows you to execute a function inside the Angular zone.

Google geocoder contains the geocode method. if you want to the access the controls into it then we have to use the ngzone.

Here Ngzone we have to take from @angular/core

• Ng Zone is an Angular service that provides a way to run code inside or outside of the Angular zone.

• The Ng Zone service exposes a method called run() that allows you to execute a function within the Angular zone.

• Angular automatically runs most of the code inside the zone, but there are scenarios where you may need to explicitly run code outside the zone to optimize performance.

**Real time Example about ngZone:**

We recently add the one textbox which we have to display world wide areas. So that we have taken one of the google api.

So we just use the google api to that textbox.

So the data whatever we are fetching from google api and attach to angular we use the ngZone class.

**? What is AOT?**

The Angular **ahead of time** (AOT) compiler converts our Angular HTML and Type Script code into efficient JavaScript code during the build phase before the browser downloads and runs that code(during build process)

**? What is JIT?**

JIT which is useful for compiles the application in the browser at runtime.

**? What is Ivy?**

Ivy is the kind of code name for Angular's compilation and rendering pipeline.

Ivy is the code name for the next-generation rendering engine in Angular, the popular front-end framework developed by Google. Angular Ivy was a significant update introduced to improve various aspects of the framework, including performance, bundle size, and developer experience.

**? What is Angular CLI?**

The Angular CLI is a command-line interface tool that you use to initialize, develop, and maintain Angular applications directly from a command prompt.

**? What ig ngCC?**

ngcc means **Angular compatibility compiler** is designed to process code coming from NPM and produce the equivalent Ivy version, as if the code was compiled with ngtsc.

**ngtSc** is a kind of wrapper of the TSC Compiler.

**Difference between AOT vs JIT?**

|  |
| --- |
| **JIT**: JIT downloads the compiler and compiles code exactly before Displaying in the browser.  Code checks at runtime  **AOT**: AOT has already complied with the code while building the application, so it doesn’t have to compile at runtime.  Code checks at compiletime |
|  |
|  |

**What type of dom angular implement?**

**In Angular apps, a real DOM is used**, which means the whole tree structure is refreshed when a single change is made.

In Angular, the virtual DOM (Document Object Model) is an integral part of its change detection mechanism, and it is known as the "Application Model" or "View Engine." Angular's change detection is responsible for keeping the actual DOM in sync with the application state.

.html

**How we can share the data between the components?**

We can share the information through like @input or @output or via services

Input and Output Properties:

• Use @Input and @Output decorators to pass data from a parent component to a child component and receive events from the child component.

View Child and Content Child:

• Use @ViewChild or @ContentChild to access child components directly and interact with their properties and methods.

**What is transpiling in angular?**

code from a high level language gets converted to another high level language

**What is Compiling in Angular?**

code from a high level language is get converted to machine level language.

The compilation process in Angular is an essential step in preparing an application for execution in a web browser. It ensures that the code is efficient, optimized, and compatible with various browsers and environments. The compilation process can be initiated manually using Angular CLI commands or integrated into a build process using tools like Webpack or other build tools.

**What is Http Service?**

The HTTP service is the component of the Application Server that provides facilities for deploying web applications and for making deployed web applications accessible by HTTP clients.

**What are important the methods in HTTP Service?**

HTTP Get, HTTP Post, HTTP Patch, HTTP Put, Http Patch

**What is HTTP Get method used for?**

The HTTP GET request method is used to request a resource from the server

The HTTP GET method is used to request data from a specified resource on a server. It is one of the standard HTTP methods used in web development and follows the request-response model.

**What is httpheaders?**

server share additional information about the HTTP request or response

**What is Http Params?**

HttpParams allows you to pass parameters with HttpClient request in angular.

Using HttpParams helps ensure that URL parameters are correctly formatted, URL-encoded, and appended to the request URL. It provides a clean and convenient API for working with query parameters in Angular HTTP requests.

**Http post vs HTTP get**

In Http Post we can pass the data through the payload whereas http get used for fetch the records from server

**Difference between http post and http get?**

Http Get: For fetch the result.

Http Post: For post the data to the server for create the records

Http Put: For post the data to the server for update the record

Http Patch: For post the partial the data to the server for update the records

Http Delete: For delete the data in the server

**? What is http interceptor?**

HTTP Interceptors is a special type of angular service that we can implement.

It's used to apply custom logic to the central point between the client side and server side outgoing/incoming HTTP request and response.

**Real time example:** There is some situation I have to add the extra headers based on the few requests. So that situation I have use that.

And needs to handle the exceptions of the each http request so that we use the http interceptor.

**?What is httpClient?**

The Http Client is a kind of a class instance acts as a session to send HTTP requests.

HTTP Client is a service in Angular that simplifies making HTTP requests to remote servers and handling the responses. It is part of the @angular/common/http module. HTTP Client provides a high-level, client-side API for communication with servers using the HTTP/HTTPS protocol. It supports various HTTP methods such as GET, POST, PUT, DELETE, etc.

**What is Viewencapsulation?**

View encapsulation defines whether the template and styles defined within the component can affect the whole application.

It’s having the 3 options

Those are none, native and emulated

**Viewencapsulation.none** means no shadow dom and no encapsulation

**Viewencapsulation.native**: means its contains the shadow dom and contains the style encapsulation

**Viewencapsulation.Emulated**: means Its wouldn’t contain the shadow dom and will contains the style encapsulation

viewEncapsulation is a concept in web development,particularly in the context an angular,popular javascript framework.it refers to the mechanism of encapsulation style within a component, preventing them from affecting other component this helps maintain modularity and avoid style conflicts in large application.

**? What is simple changes?**

SimpleChanges is an Angular Core feature that can be used to see the changes and a few more details of the declared property names in a component.

In angular , “simple change” might refer to basic modification or updates made to a component’s properties such as changing a variable value updating the content displayed, or adjusting simple attributes .It’s a general term for straightforward alteration in the data or behaviour of an angular components.

**What is ngcontainer?**

ng container allows us to create a division or section in a template without introducing a new HTML element.

In angular , ng-container is a container element used to group element together without introducing an additional element to DOM .It’s a structural directives, unlike other structural directives like ngif or ngfor , ng-container itself does not render any html.

**? What is ngtemplate?**

ngTemplate is a template which can renders based on condition in DOM. If condition is true then it can renders in the DOM or it cannot

An “NG-template” is a way to create dynamic web pages using Angular , a framework for building web applications. It’s like a special kind of Html that can change and update based on data or user action, marking your website more interactive the “ng” part is just a way angular identifier its special features within these template.

**Difference between viewchild and viewchildren**

ViewChild is used to select an element from component's template while ContentChild is used to select projected content.

**What is Template variable?**

Template variables help you use data from one part of a template in another part of the template.

Use template variables to perform tasks such as respond to user input or finely tune your application's forms.A template variable can refer to the following:a DOM element within a templatea directive or componenta TemplateRef from an ng-template

**? What is TemplateRef?**

TemplateRef is a class and the way to reference the ng-template in the component or directive class. Using the TemplateRef we can manipulate the template from component code.

A Templateref in Angular is class that represents an embedded template.. it allow you to reference and instantiate a template definrd in your Angular component. When you use ng-template in your angular component’s Html , you can obtain a reference to this template using Templateref. This reference can then be passed around and used dynamically for example, when creating dynamic component or rendering content conditionally.

**? What is ViewContainerRef?**

A ViewContainer represents a collection of ViewRef items. We can obtain a ViewContainerRef for every element in the template through dependency injection in a component

**? What is ElementRef?**

An ElementRef is backed by a render-specific element. In the browser, this is usually a DOM element.

Element one kind of a element we used import form angular core package element reroutf one kind of gobal

**What happens if you use script tag inside template?**

Angular recognizes the value as unsafe and automatically sanitizes it, which removes the script tag but keeps safe content such as the text content of the script tag.

**? What is template expression? Template expression?**

A template expression **produces a value and appears within the double curly braces, {{ }}**

**? What is template statement?**

Template statements are methods or properties that you can use in our HTML to respond to user events

**? What is exception handling?**

Exception handling is the process of handle the errors throughout page or application. We can do by try, catch, finally blocks

The **try** statement defines the code block to run (to try).

The **catch** statement defines a code block to handle any error.

The **finally** statement defines a code block to run regardless of the result.

The **throw** statement defines a custom error.

Both **catch** and **finally** are optional, but you must use one of them.

**? What is global exception?**

global error handler class which **catches all exception occurring in the App**.

By using the error handler interface we can handle.

**? How did you handle the global exception?**

So if we want to handle the global exception in the entire the project we should use the catchError operator from rxjs. and then we have to connect to the each http request.

next.handle(req).pipe(

***catchError***((error) => {

console.log('error is intercept')

console.error(error);

return throwError(error.message);

})

**? Difference between div and ngcontainer?**

Actually Div is a part of html and ngcontainer is part of angular

**? What is commonmodule?**

CommonModule is used to export all the basic Angular directives and pipes

**? What is browsermodule?**

The browser module is imported from @angular/platform browser and it is used when you want to run our application in a browser.

**? What is Rxjs?**

It’s a third party library in angular which is for reactive programming using observables that makes it easier to compose asynchronous or callback based code.

**? What is Promise?**

The Promise object represents the eventual completion (or failure) of an asynchronous operation and its resulting value.

**? What is observable?**

Observables provide support for data sharing between publishers and subscribers in an angular application.

It is referred to as a better technique for event handling, asynchronous programming, and handling multiple values as compared to techniques like promises.

**? Difference between the Promise and Observable? Promise vs Observable?**

Promises deal with one asynchronous event at a time, while observables handle a sequence of asynchronous events over a period of time.

And an Observable is allows to pass zero or more events where the callback is called for each event. By using the observable we can cancel the event by doing the unsubscribe method.

**What is subject?**

A Subject is a special type of Observable that allows values to be multi casted to many Observers.

**What is Behaviour subject?**

A Subject is a special type of Observable that allows values to be multicasted to many Observers.

 And always need an default value

Subject is having no default values

**What is swtichmap?**

The Angular SwitchMap maps each value from the source observable into an inner observable, subscribes to it, and then starts emitting the values from it.

**What is mergemap?**

The Angular mergemaps each value from the source observable into an inner observable, subscribes to it, and then starts emitting the values from it and from it replacing the original value

**What is flatmap?**

flatMap() operator in rxjs allows to process the data array which comes from the observable.

**What is forkjoin?**

Fork Join allows us to group multiple observables and execute them in parallel, then return only one observable.

**Difference between switchmap and map?**

The map operators emits value as observable.

The SwitchMap creates a inner observable, subscribes to it and emits its value as observable.

**What is Pipe in Rxjs?**

The pipe method of the Angular Observable is used to chain multiple operators together.

**What is tap?**

Tap in Rxjs which Can perform side effects with observed data but **does not modify** the stream in any way.

**What is Debounce time in RxJs?**

DebounceTime delays the values emitted by a source for the given due time.

**Examples of the observable?**

Http is the observable

**What is bootstrapping in angular?**

bootstrap is a function component in the core ng module that is used for starting up the Angular application manually

**What is Digest Cycle Process in angular?**

Digest cycle is what Angular which triggers when a value in the model or view is changed.

**How to create the custom observable?**

Observable is the class which import from the Rxjs package. By using that we can create the custom observable

**Example of the Predefined Observable in Angular?**

Http is predefined observable in angular

**What is Subscribe method?**

Subscribe is the method of the observable to take the response from the observer.

**How to unsubscribe the Observable?**

Observable having the unsubscribe () method. By doing that we can do.

**What is Synchronous Transmission?**

Synchronous data transmission is a data transfer method in which is a continuous stream of data signals accompanied by timing signals.

**W**h**at is Asynchronous Transmission?**

Asynchronous Transmission is also known as start/stop transmission, sends data from the sender to the receiver using the flow control method.

**? What is angular CLI?**

Angular cli is great tool in angular. It really focus to you work on angular code instead of workflow tasks.

Typescript should be compile before it runs into browser. So it’s very pathetic for doing manually here. Angular cli will take care of this kind of activities.

**? What is nodejs and why its need?**

NodeJS is useful for manages the dependencies of packages, as well as we can host the application into server.

**? What is Schematic?**

A schematic is **a template-based code generator that supports complex logic**. It is a set of instructions for transforming a software project by generating or modifying code.

**? What is the purpose of schematic?**

The purpose of schematic design is **to translate the project program into physical drawings of space**

**? What is multicasting in angular?**

Multicasting is **the practice of broadcasting to a list of multiple subscribers in a single execution**. With a multicasting observable, you don't register multiple listeners on the document, but instead re-use the first listener and send values out to each subscriber.

**What are the different type of compilers?**

You can compile our angular application in two ways: **JIT and AOT**. Both are suitable for a different scenario like you can use JIT for development mode and AOT is better in production mode

**What is angular universal?**

Angular Universal **executes on the server, generating static application pages that later get bootstrapped on the client**. This means that the application generally renders more quickly, giving users a chance to view the application layout before it becomes fully interactive.

**What are macros?**

[https://intellipaat.com/blog/interview question/angular interview questions/](https://intellipaat.com/blog/interview%20question/angular%20interview%20questions/)

**What is our project architecture?**

Begin by giving a high-level overview of the project architecture, explaining the key components and how they interact with each other. Use diagrams: Use diagrams, flowcharts, or other visual aids to help explain the architecture**.**

**What is settimeinterval?**

setInterval allows us to run a function repeatedly, starting after the interval of time, then repeating continuously at that interval

**What is setTimeout?**

setTimeout allows us to run a function once after the interval of time

Angular, typescript, javascript, html, bootstrap, UIKit, css,.net,.net core, core java, restful services,.net webapis, asp.net, csharp, reactive, sql, oracle, dynamodb, aws, azure

**What is cache busting in angular?**

This means that once cached locally on the user's browser, the browser won't check the origin server again for 1 whole year to see if any updates have been made to the file

**Cache busting purpose:**

Cache busting is a way to prevent browsers from caching our ad content

**What is Shadow DOM?**

Shadow DOM is like a parallel DOM tree hosted inside a component (an HTML element, not to be confused with Angular components), hidden away from the main DOM tree.

**What is transpiler and transpiling?**

Transpiring is the one of the process to convert the typescript into java script.

**More:**

Typescript does transpile into Javascript. In the tsconfig configuration file, we can specify which the target language we want it transpired to is.

**What is sass vs css?**

CSS is a style language that is used to style and create web pages. While SCSS is a particular type of file for SASS, it used the Ruby language, which assembles the browser's CSS style sheets. SCSS contains advanced and modified features. SCSS is more expressive than the CSS.

**What is CSS boxModel?**

In CSS, the term "box model" is used when talking about design and layout. The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content.

**Diff between hidden and visible? Visible vs hidden?**

**display:none** means that the tag in question will not appear on the page at all (although you can still interact with it through the dom). There will be no space allocated for it between the other tags.

**visibility:hidden** means that unlike **display:none**, the tag is not visible, but space is allocated for it on the page. The tag is rendered, it just isn't seen on the page.

**How many type of selectors we have in css?**

The simple selector is the most direct form of selector that targets HTML elements with identifiers such as id, class, and others that can be attached to an HTML tag on the page