Date _____ Page _____

11. Difference between Typescript & Javascript.

1. Type System: Typescript is a staticallly typed superset of Javascript, meaning it introduces static typing to Javascript.

Javascript on the otherhand is dynamically typed.

2. Compilation: Typescript code needs to be compiled into Javascript before it can run in a browser or any JavaScript runtime.

Javascript on the otherhand is an interpreted language & doesn't require explicit compilation.

3. Language Features: Typescript includes additional language features are not natively present in Javascript. These features include static typing.

Javascript lacks these features, although some of them can be simulated using various patterns or external libraries.

4. Compatibility: Typescript is a superset of Javascript which means all valid Javascript code is valid for Typescript code.

Date _____ Page _____

Typescript introduces new features new features and syntax but Javascript code can be gradually migrated to Typescript without any major issues.

(2) The need for web Framework:

A web framework is a Software framework that provides a structure set of tools, and libraries for developing web applications

i) Efficiency: They offer built-in functions, modules and utilities that reduce the amount of code developers need to write from scratch.

ii) Organization: Frameworks provide a Standardized way to Organize code, files, and directories.

iii) Security: Many web frameworks include Security measures by default. like cross-site scripting, cross-site request forgery and SQL injection.

iv) Ecosystem & Community: Web frameworks often have a vibrant ecosystem and community around them.

Date _____ Page _____

Distinguish between Client-side and Server-side frameworks:

Client-side frameworks are primarily focused on running in the web browser. They are responsible for rendering the user interface (UI) managing UI state, handling user interfaces, and communicating with servers or API's.

1) UI Rendering: Client-side frameworks excel in rendering dynamic and interactive UI's. They often use a Virtual DOM or other efficient rendering techniques to update.

2. Javascript Execution: Client-side frameworks rely heavily on Javascript for their execution.

3. Single-page-Application: Client-side frameworks are commonly used for building SPAs where the entire application runs in the browsers and communicates with the server through APIs. They provide a smooth and responsive user experience by avoiding full page reloads.

Date Page

15. In Angular we can specify the type of stylesheet for our application by using "style" property in the 'Component decorator.

There are two main types of stylesheet,

1. Open the component file (e.g component.ts') where we want to specify the stylesheet type.
2. Import the 'Component' decorator from '@angular/core' if it is not already imported.
3. Add the 'styleUrls' property to the component decorator. This property accepts an array of strings representing the URLS of the stylesheet files.

Example:

```
import { Component } from '@angular/core';

@Component ({
    selector: 'app-example';
    templateUrl: './example component.html';
    styleUrls: ['./example.component.css']
})
export class ExampleComponent {
    // logic here
}
```

Date Page

Example for SCSS (Sass),

```
import { Component } from '@angular/core';

@Component({
    selector: 'app-example',
    templateURL: './example.component.html',
    styleUrls: ['./example.component.ccss']
})
export class ExampleComponents {
    // logic here.
}
```

By specifying, the appropriate file extension in the 'styleURLs' property we can indicate the type of stylesheet we want to use in Angular Application.

14. There are several reasons why Typescript is prioritized over Javascript in Angular development.

1 Strong Typing: Typescript introduces Static typing to JavaScript, which enables the compiler to catch errors during development. It also provides better tooling support such as auto completion, refactoring.

2. Enhanced IDE Support: Typescript is supported by popular IDEs like Visual Studio Code, which provides advanced features like IntelliSense, code navigation, and error checking.

3. Object Oriented Programming (OOP) Features:

Type script is an object-oriented superset of Javascript. It includes features like classes, interface, inheritance and methods, which facilities the development of complex applications and promote code.

4. Improved Developer Experience: Typescript offers a more pleasant development experience with features like optional parameters and property types, default values, and type inference. It also provides a more structured approach to coding with its static typing.

5. Angular Specific Features: Angular is built using Typescript, and it provides first-class support for their functionality

15. In Angular, templates and directives are two fundamental concepts used to build dynamic and interactive web applications. Here's an

1. Template in Angular:

Templates in Angular are the user interface views of an application. They define the structure layout of the application's components, determining how the components should be rendered on the screen. Templates use HTML syntax enhanced with Angular specific features and directives to display data handle user input and manipulate the DOM.

Angular templates can include binding expressions that allow the interpolation of components data into the HTML markup, enabling dynamic rendering. They also support event binding, which allows the execution of components methods in response to user actions, such as button clicks or form submissions.

Additionally Angular templates can utilize structural directives like *ngIf and *ngFor to conditionally render or repeat elements based on certain conditions.

Date _____ Page _____

1 Directives in Angular:

Directives in Angular are markers on DOM elements that instruct Angular to perform certain actions or apply specific behavior to those elements. Directives Components directives Structural directives, and attributes directives.

- Component Directive: Component directives represent custom reusable components in Angular.

- Structural Directives: Structural directives modify the structure of the DOM by adding, removing, or manipulating elements based on conditions.

- Attribute Directive: Attribute directives modify the behaviour or apperance of DOM element by applying custom behaviour or style. Example of attribute directive in Angular include ngClass, ngStyle, and ngModel.