	SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGG.	**LABORATORY MANUAL**
	PRACTICAL EXPERIMENT INSTRUCTION SHEET	
	EXPERIMENT TITLE: Install Angular CLI and Create First Angular Application.	
EXPERIMENT NO. : **SSGMCE/WI/IT/01/5IT09/01**	ISSUE NO. : 00	ISSUE DATE : 01.07.2025
REV. DATE :	REV. NO. :	DEPTT. : INFORMATION TECHNOLOGY
LABORATORY : COMPUTER SKILL LAB–III (5IT09)	SEMESTER : V	PAGE: 1 OF 4

1.0) AIM: To install Angular CLI and create a basic Angular 20 application using command-line interface and Visual Studio Code and create your first angular app.

2.0) SCOPE: This experiment enables students to understand the Angular CLI tool and use it to initialize, build, and serve a basic Angular application. The scope includes environment setup, project creation, and launching the Angular app in a browser using the Angular development server.

3.0) FACILITIES/ APPARATUS:

i) Hardware: Intel i3/i5 Processor or higher, minimum 8GB RAM, HD Monitor, Internet Connectivity.

ii) Software: Windows 10/11 Operating System, Node.js (LTS version), Angular CLI (latest version), Visual Studio Code Editor, Google Chrome Browser.

4.0) THEORY:

4.1) Angular

Angular is a web framework that empowers developers to build fast, reliable applications. Maintained by a dedicated team at Google, Angular provides a broad suite of tools, APIs, and libraries to simplify and streamline your development workflow. Angular gives you a solid platform on which to build fast, reliable applications that scale with both the size of our team and the size of our code base.

Angular is a popular open-source JavaScript framework used for building dynamic web applications. a comprehensive set of tools and features for developing efficient and robust Single Page Applications (SPAs). Here's an overview of key concepts and features in Angular:

1. **Components:** Angular applications are built using components. A component is a self-contained unit that encapsulates HTML, CSS, and TypeScript logic for a specific part of the user interface. Components are modular and can be reused throughout the application.
2. **Templates:** Templates define the structure and layout of a component's view using HTML. Angular provides features like data binding and directives to manipulate the DOM and display dynamic content.

3. **Data Binding:** Angular offers various types of data binding to establish a connection between the component's logic and its view. This includes one-way binding (from component to view or vice versa) and two-way binding, where changes in the view are reflected in the component and vice versa.
4. **Directives:** Directives are instructions in the DOM that manipulate the behavior of elements. Angular provides built-in directives like ngIf, ngFor, and ngSwitch for conditional rendering and looping.
5. **Services:** Services are reusable components that provide specific functionality, such as data retrieval, authentication, or other business logic. They are designed to be injectable into components and can be shared across different parts of the application.
6. **Dependency Injection:** Angular's dependency injection system manages the creation and distribution of components and services throughout the application. This promotes modularity and reusability by allowing components to request dependencies rather than creating them themselves.
7. **Modules:** Angular applications are organized into modules, which group related components, services, and other pieces of functionality together. Modules help with modularity and maintainability of the application.
8. **Routing:** Angular's router allows you to define navigation paths and views in a single-page application. It enables users to navigate between different views without the need for full page reloads.
9. **Forms:** Angular provides tools for creating and handling forms, including template-driven forms and reactive forms. Reactive forms offer a more flexible and programmatic approach to handling form validation and user input.
10. **HTTP Client:** Angular's HTTP client module allows you to make requests to external APIs and services. It provides features for handling HTTP requests, responses, and error handling.
11. **Observables:** Angular leverages RxJS observables to handle asynchronous operations, including HTTP requests, event handling, and more. Observables provide a way to work with asynchronous data streams.
12. **Pipes:** Pipes are used for data transformation and formatting in templates. They allow you to display data in a desired format without modifying the underlying data.
13. **Testing:** Angular provides tools and utilities for writing unit tests and end-to-end (e2e) tests to ensure the quality and reliability of your application.

Angular is continuously evolving, and the official documentation, tutorials, and community resources provide comprehensive guidance for learning and mastering the framework. It's recommended to explore these resources to gain a deeper understanding of Angular's features and capabilities.

Regenerate.

Angular Environment Setup Procedure:

Setting up the environment for Angular development involves installing the necessary software and tools to create, build, and run Angular applications. Here's a step-by-step procedure to set up your Angular development environment:

1. **Node.js and npm:** Angular requires Node.js, a JavaScript runtime, and npm (Node Package Manager) for managing packages and dependencies.
   - Download and install Node.js and npm from the official website: https://nodejs.org/
2. **Angular CLI:** Angular CLI (Command Line Interface) is a powerful tool for creating, managing, and building Angular projects.
   - Open a terminal or command prompt.
   - Install Angular CLI globally by running:

   Command: **npm install -global @angular/cli@latest**
3. **Code Editor:** Choose a code editor for writing your Angular code. Popular choices include Visual Studio Code, Sublime Text, or WebStorm.
   - Download and install your preferred code editor.
4. **Create a New Angular Project:** Once Angular CLI is installed, you can create a new Angular project using the CLI.

   Command: **ng new project_name**

   - Navigate to the directory where you want to create your project using the terminal.
   - Follow the prompts to select options like whether to include Angular routing and which CSS preprocessor to use.
5. **Navigate to the Project Directory:** Change your current working directory to the newly created project.

   cd project-name

6. **Serve the Application:** Angular CLI provides a development server that compiles your code and serves the application locally for testing.
   - Run the following command to start the development server and open the application in your default browser:

   Command: **ng serve**

7. **Access the Application:** After running the development server, you can access the application by opening a web browser and navigating to **http://localhost:4200**.
8. **Start Coding:** Now you're ready to start coding your Angular application. You can edit the files in the **src** folder to modify your application's components, templates, and styles.

This is a basic setup procedure. Depending on your project's requirements, might need to configure additional tools, libraries, or services such as version control (e.g., Git), IDE extensions, testing frameworks.

5.0) Conclusion:

Angular CLI and TypeScript are essential tools that significantly contribute to the Single Page Application development process when working with the Angular framework.

Output: