|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Student:** Atharva Mohite | | | |
| **Roll Number:** 34 | | **LAB Assignment Number:** 9 | |
| **Title of LAB Assignment:** Introduction to Angular,Setup for local Development environment,Angular Architecture Create an application to demonstrate directives and pipes | | | |
| **DOP:** 23-10-2023 | | **DOS:** 23-10-2023 | |
| **CO Mapped:** CO5 | **PO Mapped:** PO3, PO5, PSO1, PSO2 | **Faculty Signature:** | **Marks:** |

# PRACTICAL 9

## Aim: Introduction to Angular,Setup for local Development environment,Angular Architecture Create an application to demonstrate directives and pipes

## Theory

1. **Introduction to Angular**

Angular is a popular open-source web application framework developed and maintained by Google. It is used for building dynamic, single-page web applications (SPAs) and offers a comprehensive set of tools and features for front-end development. Angular is built using TypeScript, a superset of JavaScript, which adds static typing to the language. It follows the component-based architecture, making it easy to create reusable UI components and manage the application's state.

1. **Setup for Local Development Environment**

To set up a local development environment for Angular, follow these steps:

1. **Node.js and npm:** Ensure you have Node.js and npm (Node Package Manager) installed on your computer. You can download and install them from the [official website](<https://nodejs.org/>).
2. **Angular CLI:** Install the Angular Command Line Interface (CLI) globally by running the following command in your terminal (please use cmd, avoid using powershell):  
   npm install -g @angular/cli
3. **Create an Angular Application:** Use the Angular CLI to create a new Angular application. Navigate to the desired directory in the terminal and run the following command (please use cmd, avoid using powershell):  
   ng new your-app-name
4. Follow the prompts to set up the project and dependencies.
5. **Start the Development Server:** Once the project is created, navigate to its directory using the terminal and run the following command (please use cmd, avoid using powershell):  
   cd your-app-name  
   ng serve

This command will start a development server, and your application will be available at <http://localhost:4200/> by default.

1. **Angular Architecture**

Angular follows a component-based architecture. Here are some key components of Angular's architecture:

**Modules:** Modules help organize an Angular application. They group related components, directives, services, and other code. Angular applications are organized into modules. A module is a container for components, services, and other code. The root module is typically named AppModule and is defined in the app.module.ts file.

**Components:** Angular applications are built around components, which are self-contained and reusable pieces of the user interface. Components are the building blocks of an Angular application. They consist of TypeScript code, HTML templates, and CSS styles. Each component is responsible for a specific part of the UI and the associated logic.

**Templates:** Templates define the structure and layout of the UI. Angular templates use HTML enhanced with Angular-specific syntax. These templates are associated with components and are rendered in the browser.

**Directives:** Directives are used to add behavior to the DOM elements. Directives are instructions in the DOM that tell Angular to do something with an element. Common built-in directives include ngFor for looping through lists and ngIf for conditional rendering.

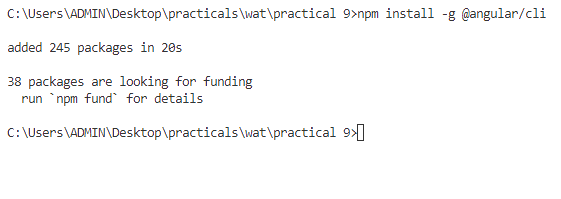
**Router:** Angular's router allows you to create Single-Page Application (SPA) with multiple views and navigation. It handles navigation between different components.

## Aim: Create an application to demonstrate directives and pipes

### Steps to execute the practical:

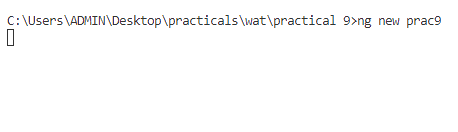
1. Create a folder “practical 9”
2. Open the terminal (please use cmd, avoid using powershell) and navigate to “practical 9” folder by using the “cd” command
3. Install Angular CLI globally by running the following command

npm install -g @angular/cli

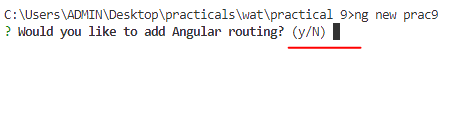


1. Create a new angular project by running the following command

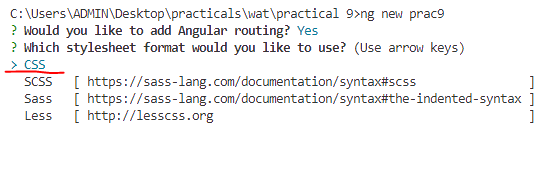
ng new prac9



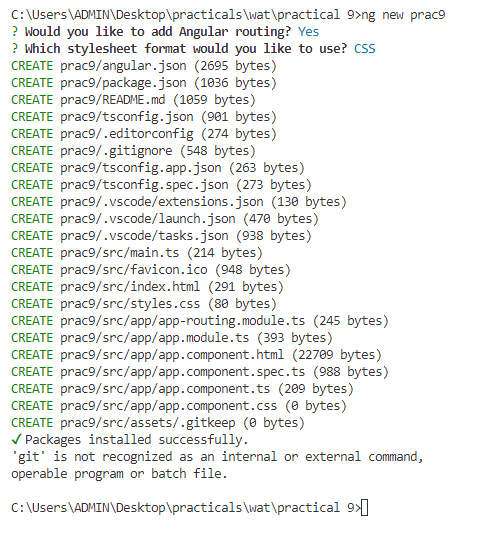
1. When prompted for Angular Routing, Enter “y”



1. When prompted for CSS, use arrow keys to select “CSS” (it is the first option and is selected by default) and then press Enter

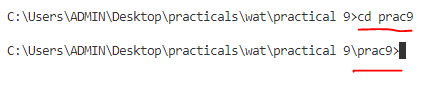


1. The new Angular project is created



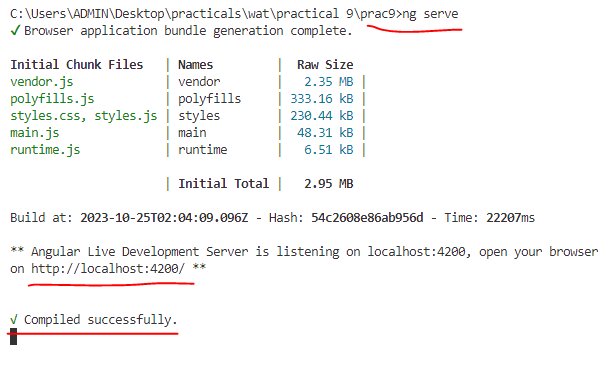
1. Navigate to the prac9 folder by using the cd command

cd prac9

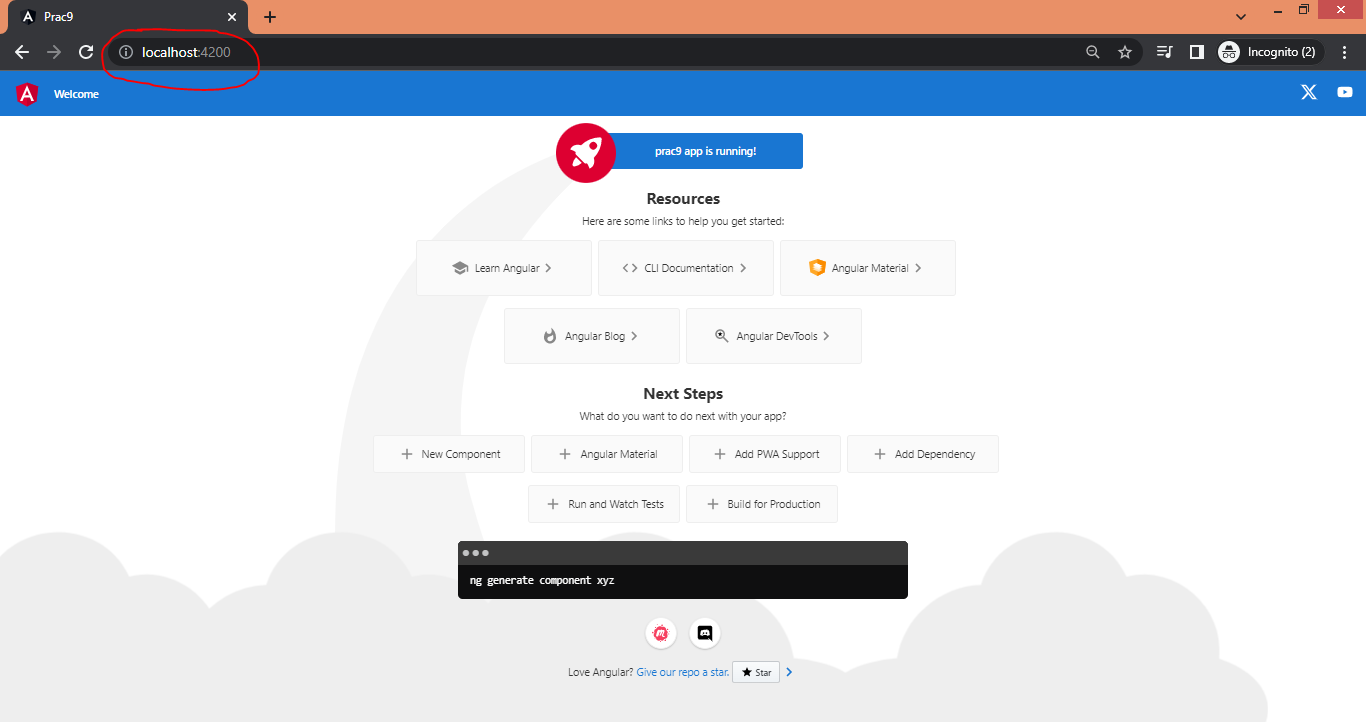


1. Run the following command to serve the angular project on a server

ng serve



1. The server is running and will automatically re-compile the project when we make any changes
2. Open any browser and go to the following link to see the output (this is the default output whenever any new Angular project is made)  
   <http://localhost:4200/>

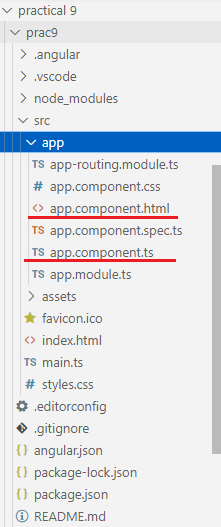


1. We will remove the default code that’s generated and add our practical’s code.

We will make changes to the following 2 files in the “app” folder (prac9/scr/app/)

app.component.html

app.component.ts



1. Open the app.component.html file and delete all the code
2. Write the following code in app.component.html

<!-- directive -->

<!-- ngIf directive -->

<!-- reference:- https://www.javatpoint.com/angular-8-ngif-directive -->

<h1>Directives</h1>

<h2>ngIf</h2>

n is = {{n}}

<div \*ngIf="n > 2; else elseBlock">

  n is greater than 2

</div>

<ng-template #elseBlock>

  n is not greater than 2

</ng-template>

<!-- ngFor directive -->

<!-- reference:- https://www.javatpoint.com/angular-8-ngfor-directive -->

<h2>ngFor</h2>

<li \*ngFor="let item of items">

  {{item}}

</li>

<!-- piping -->

<!-- reference:- https://www.tutorialspoint.com/angular8/angular8\_pipes.htm -->

<h1>Piping</h1>

<h2>Adding parameter</h2>

<!-- adding parameter -->

<div>

  Today's date :- {{presentDate}}

</div>

<div>

  Date with uppercase :- {{presentDate | date:'fullDate' | uppercase}}

  <br />

  Date with lowercase :- {{presentDate | date:'medium' | lowercase}}

  <br />

</div>

<!-- currency pipe -->

<h2>Currency pipe</h2>

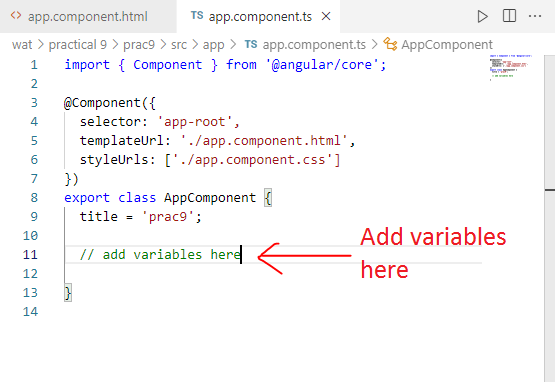
<div>

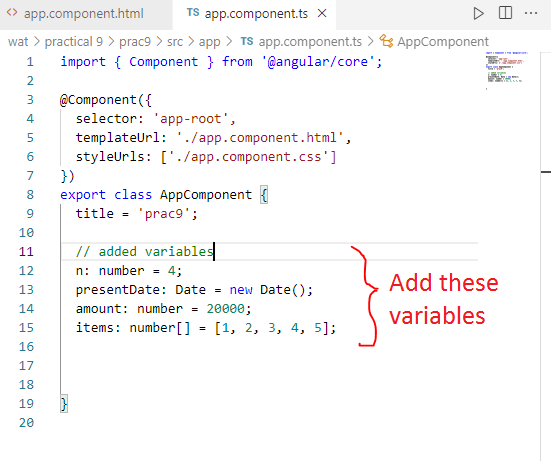
  <p>{{ amount | currency:'EUR'}}</p>

  <p>{{ amount | currency:'INR' }}</p>

</div>

1. Save the app.component.html file
2. Open the app.component.ts and make the following changes in it





The app.component.ts code should look like this

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

@Component({

  selector: 'app-root',

  templateUrl: './app.component.html',

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'prac9';

  // added variables

  n: number = 4;

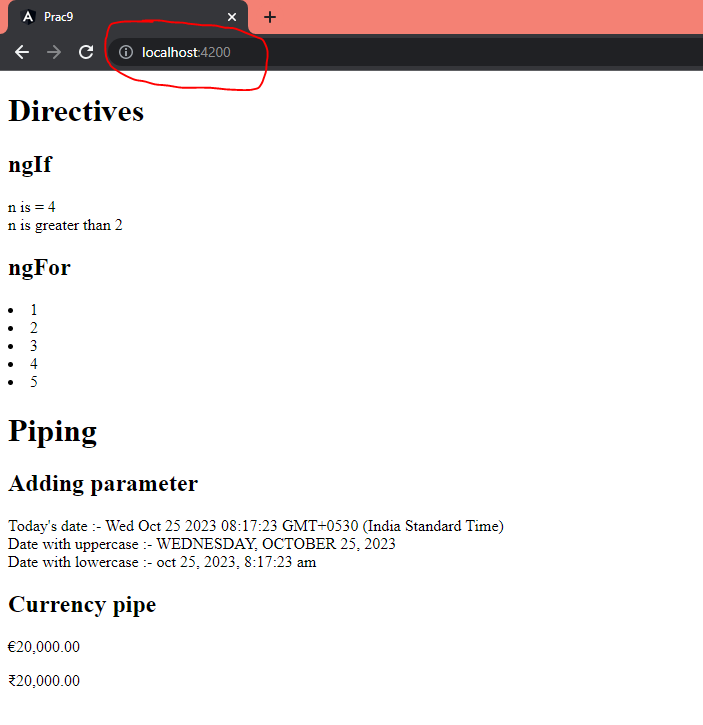
  presentDate: Date = new Date();

  amount: number = 20000;

  items: number[] = [1, 2, 3, 4, 5];

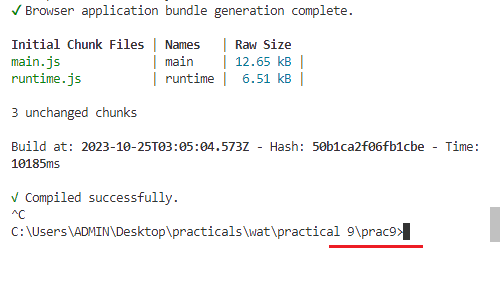
}

1. Save the app.component.ts file
2. Go to the browser to see the output at <http://localhost:4200/>



1. Go to the terminal and stop the server by pressing Ctrl + C

When we see the prompt as “....\prac9>” then it means that the server has stopped



### Code

#### app.component.html

<!-- directive -->

<!-- ngIf directive -->

<!-- reference:- https://www.javatpoint.com/angular-8-ngif-directive -->

<h1>Directives</h1>

<h2>ngIf</h2>

n is = {{n}}

<div \*ngIf="n > 2; else elseBlock">

  n is greater than 2

</div>

<ng-template #elseBlock>

  n is not greater than 2

</ng-template>

<!-- ngFor directive -->

<!-- reference:- https://www.javatpoint.com/angular-8-ngfor-directive -->

<h2>ngFor</h2>

<li \*ngFor="let item of items">

  {{item}}

</li>

<!-- piping -->

<!-- reference:- https://www.tutorialspoint.com/angular8/angular8\_pipes.htm -->

<h1>Piping</h1>

<h2>Adding parameter</h2>

<!-- adding parameter -->

<div>

  Today's date :- {{presentDate}}

</div>

<div>

  Date with uppercase :- {{presentDate | date:'fullDate' | uppercase}}

  <br />

  Date with lowercase :- {{presentDate | date:'medium' | lowercase}}

  <br />

</div>

<!-- currency pipe -->

<h2>Currency pipe</h2>

<div>

  <p>{{ amount | currency:'EUR'}}</p>

  <p>{{ amount | currency:'INR' }}</p>

</div>

<!-- Explanation

    \* ngIf Directive: https://www.javatpoint.com/angular-8-ngif-directive

    \* ngFor Directive: https://www.javatpoint.com/angular-8-ngfor-directive

    \* Piping: https://www.tutorialspoint.com/angular8/angular8\_pipes.htm

 -->

#### app.component.ts

// Import the 'Component' decorator from '@angular/core'

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

@Component({

  // Selector is used to define the HTML element where the component will be displayed.

  selector: 'app-root',

  // 'templateUrl' points to the HTML template file for this component.

  templateUrl: './app.component.html',

  // 'styleUrls' is an array of CSS style files for styling the component.

  styleUrls: ['./app.component.css']

})

// This is the component class definition.

export class AppComponent {

  // the following are variables of the class, these variables will be used in the html code

  title = 'prac9';

  // added variables

  n: number = 4;

  presentDate: Date = new Date();

  amount: number = 20000;

  items: number[] = [1, 2, 3, 4, 5];

}

/\*\* Explanation:

This code defines an Angular component using the @Component decorator. Angular components are building blocks of an Angular application, responsible for controlling a part of the user interface.

The selector property defines the HTML element where the component will be displayed. In this case, it's using the selector app-root.

The templateUrl property points to an HTML template file that represents the component's view. The view defines the structure and layout of the component.

The styleUrls property is an array of CSS style files that provide styling for the component.

Inside the AppComponent class, there are several properties:

  \* title is a string property with the value 'prac9'.

  \* n is a number property with an initial value of 4.

  \* presentDate is a Date property initialized with the current date and time.

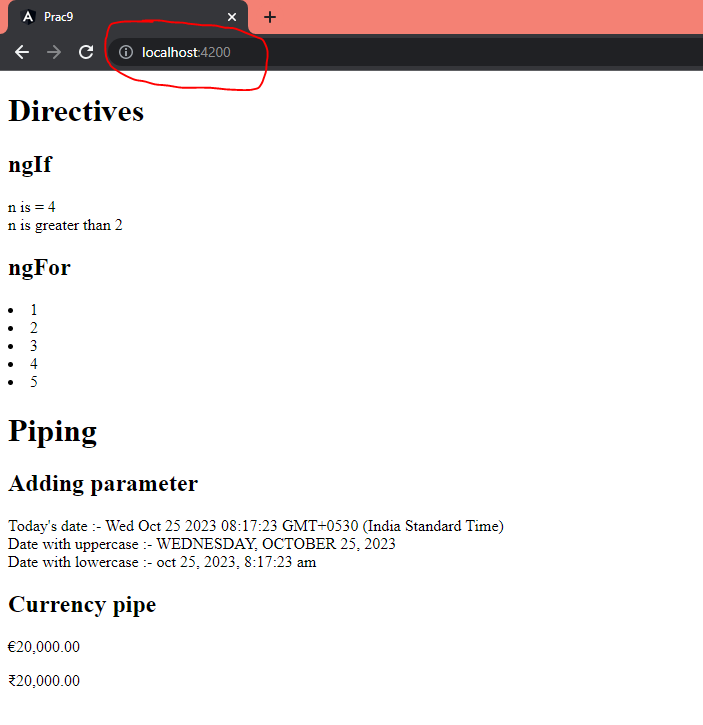
  \* amount is a number property with an initial value of 20000.

  \* items is an array property containing numbers 1 through 5.

These properties can be used in the component's template (HTML file) to display data dynamically in the user interface.

This component is a part of an Angular application and can be included in the application's module for rendering and functionality. The properties can be bound to elements in the HTML template to display their values dynamically. \*/

### Output



## Conclusion: We’ve explored an introduction to Node.JS, Advantages and Disadvantages, Node.js Process Model, Traditional Web Server Model, Installation and some programs with node.js.