MALAD KANDIVALI EDUCATION SOCIETY'S

NAGINDAS KHANDWALA COLLEGE OF COMMERCE, ARTS & MANAGEMENT STUDIES & SHANTABEN NAGINDAS KHANDWALA COLLEGE OF SCIENCE

MALAD [W], MUMBAI – 64

AUTONOMOUS INSTITUTION

(Affiliated To University Of Mumbai)

Reaccredited 'A' Grade by NAAC | ISO 9001:2015 Certified

CERTIFICATE

Name: Ms. SACHI VIRESH SHAH

Roll No: 381 Programme: BSc IT Semester: III

This is certified to be a bonafide record of practical works done by the above student in the College laboratory for the course **Hybrid Application Development (CourseCode: 2037UCSMD)** for the partial fulfillment of Third Semester of BSc IT/CS during the academic year 2020-21.

The journal work is the original study work that has been duly approved in the year 2020-21 by the undersigned.

_____ External Examiner

_____ Mr. Gangashankar Singh (Subject-In-Charge)

Date of Examination: (College Stamp)

Class: S.Y. B.Sc. IT Sem- III **Roll No: 381**

Subject: Hybrid Application Development

INDEX

Sr. No	Date	Topic	Sign
1	27/07/2020	AngularJS Data Binding	
2	7/08/2020	AngularJS Directives	
3	14/08/2020	AngularJS Controllers	
4	25/08/2020	AngularJS Events	
5	08/09/2020	Ionic Create and Build First Project	
6	12/09/2020	Ionic Adding Cordova Android Platform	
7	19/09/2020	Ionic Create, Generate and Add Pages	
8	29/09/2020	Ionic Use Tabs Starter Template	

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Hybrid App Development

Practical 1

Aim: AngularJS Data Binding

Theory:

Data binding in AngularJS is the synchronization between the model and the view. AngularJS applications usually have a data model. The data model is a collection of data available for the application.

The HTML container where the AngularJS application is displayed is called the view. The view has access to the model, and there are several ways of displaying model data in the view. You can use the **ng-bind** directive, which will bind the innerHTML of the element to the specified model property. You can also use double braces **{{ }}** to display content from the model. Or you can use the **ng-model** directive on HTML controls to bind the model to the view.

The ng-model Directive

Use the **ng-model** directive to bind data from the model to the view on HTML controls (input, select, textarea). The **ng-model** directive provides a two-way binding between the model and the view.

Two-way Binding

Data binding in AngularJS is the synchronization between the model and the view.

When data in the model changes, the *view* reflects the change, and when data in the view changes, the model is updated as well. This happens immediately and automatically, which makes sure that the model and the view is updated at all times.

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

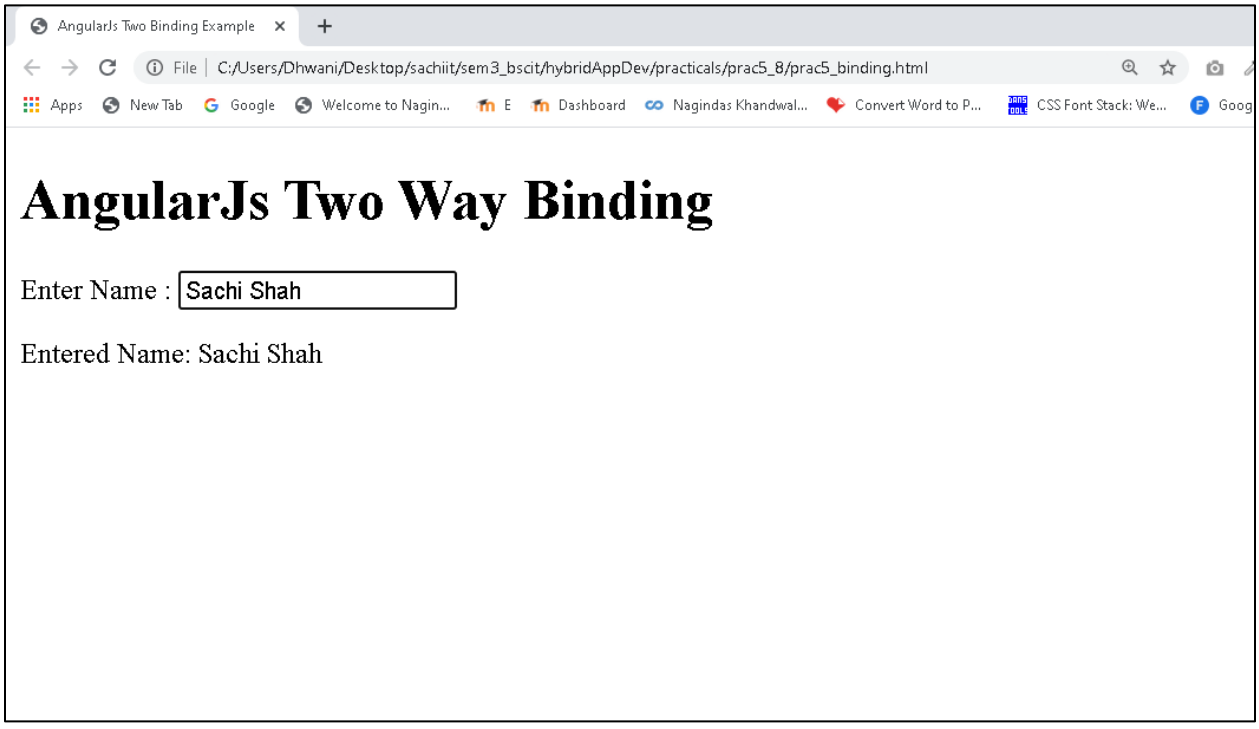
CLASS: SYIT **OLD ROLL NO: 3066**

Code: Two-way Data Binding

prac5_binding.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>AngularJs Two Binding Example</title>
        <script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
        <script type="text/javascript">

            var app = angular.module('myapp', []);
            app.controller('mycontroller', function ($scope) {
            $scope.name = '';
            });
        </script>
    </head>
    <body ng-app="myapp">

        <h1>AngularJs Two Way Binding</h1>
        <div ng-controller="mycontroller">
            <!-- whenever we change input control value automatically the appearance value also will
            get changed -->
            Enter Name : <input type="text" ng-model="name" />
            <p>
                Entered Name:   {{ name }}
            </p>
        </div>
    </body>
</html>
```

Output:



NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 2

Aim: AngularJS Directives

Theory:

AngularJS lets you extend HTML with new attributes called **Directives**. AngularJS has a set of built-in directives which offers functionality to your applications. **AngularJS also lets you define your own directives.** AngularJS directives are extended HTML attributes with the prefix **ng-**.

1. The **ng-app** directive initializes an AngularJS application. The **ng-app** directive also tells AngularJS that the element is the "owner" of the AngularJS application.
2. The **ng-init** directive initializes application data.
3. The **ng-model** directive binds the value of HTML controls (input, select, textarea) to application data.
4. The **ng-repeat** directive repeats an HTML element. The ng-repeat directive actually clones HTML elements once for each item in a collection.

The ng-app Directive

The **ng-app** directive defines the root element of an AngularJS application. The **ng-app** directive will auto-bootstrap (automatically initialize) the application when a web page is loaded.

The ng-init Directive

The **ng-init** directive defines initial values for an AngularJS application. Normally, you will not use ng-init. You will use a controller or module instead.

The ng-model Directive

The **ng-model** directive binds the value of HTML controls (input, select, textarea) to application data. The ng-model directive can also:

- Provide type validation for application data (number, email, required).
- Provide status for application data (invalid, dirty, touched, error).
- Provide CSS classes for HTML elements.
- Bind HTML elements to HTML forms.

Create New Directives

In addition to all the built-in AngularJS directives, you can create your own directives.New directives are created by using the **.directive** function.To invoke the new directive, make an HTML element with the same tag name as the new directive.

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

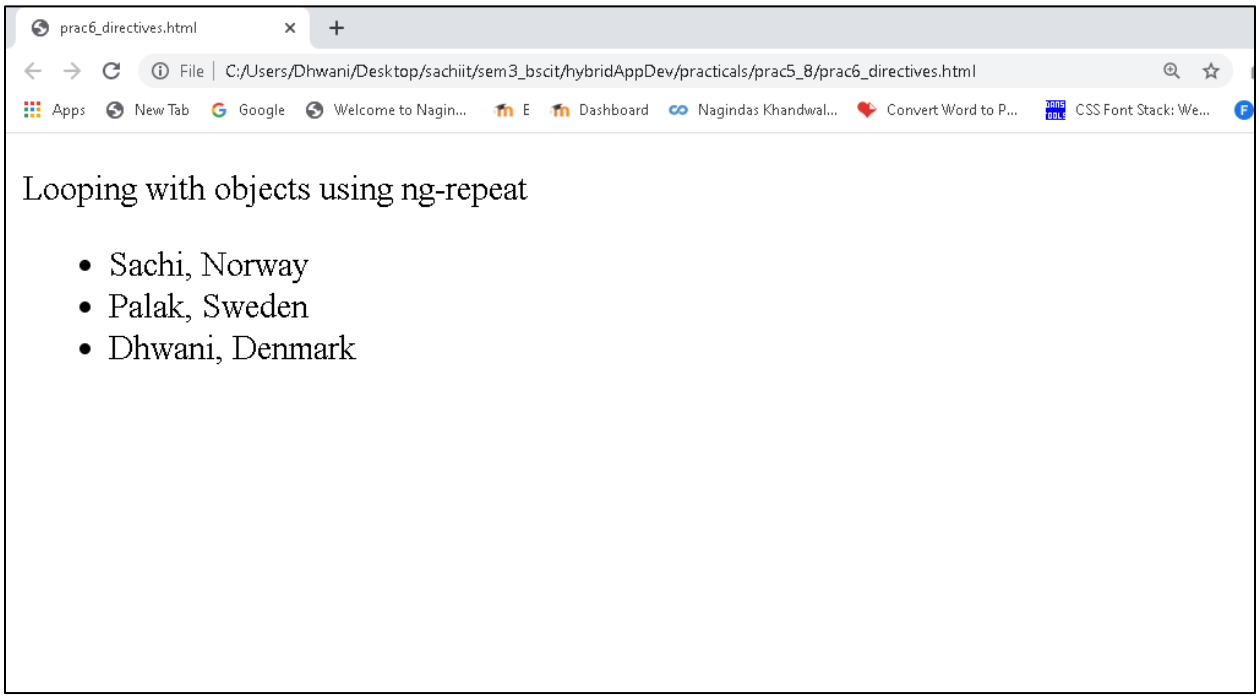
CLASS: SYIT **OLD ROLL NO: 3066**

Code:

prac6_directives.html

```
<!DOCTYPE html>
<html>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></script>
    <body>

        <!-- The ng-app directive initializes an AngularJS application.
        The ng-init directive initializes application data. -->
        <div ng-app="" ng-init="names=[
            {name:'Sachi',country:'Norway'},
            {name:'Palak',country:'Sweden'},
            {name:'Dhwani',country:'Denmark'}]">

            <p>Looping with objects using ng-repeat</p>
                <ul>
                    <!-- The ng-repeat directive used on an array of objects: -->

                    <li ng-repeat="x in names">
                        {{ x.name + ', ' + x.country }}
                    </li>
                </ul>
        </div>
    </body>
</html>

```

Output:



NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 3

Aim: AngularJS Controllers

Theory:

AngularJS controllers **control the data** of AngularJS applications. AngularJS controllers are regular JavaScript Objects. AngularJS applications are controlled by controllers. The **ng-controller** directive defines the application controller. A controller is a **JavaScript Object**, created by a standard JavaScript **object constructor**. A controller can also have methods (variables as functions). In larger applications, it is common to store controllers in external files.

Use controllers to:

- Set up the initial state of the **$scope** object.
- Add behavior to the $scope object.

Do not use controllers to:

- Manipulate DOM — Controllers should contain only business logic. Putting any presentation logic into Controllers significantly affects its testability. AngularJS has data binding for most cases and directives to encapsulate manual DOM manipulation.
- Format input — Use AngularJS form controls instead.
- Filter output — Use AngularJS filters instead.
- Share code or state across controllers — Use AngularJS services instead.
- Manage the life-cycle of other components (for example, to create service instances).

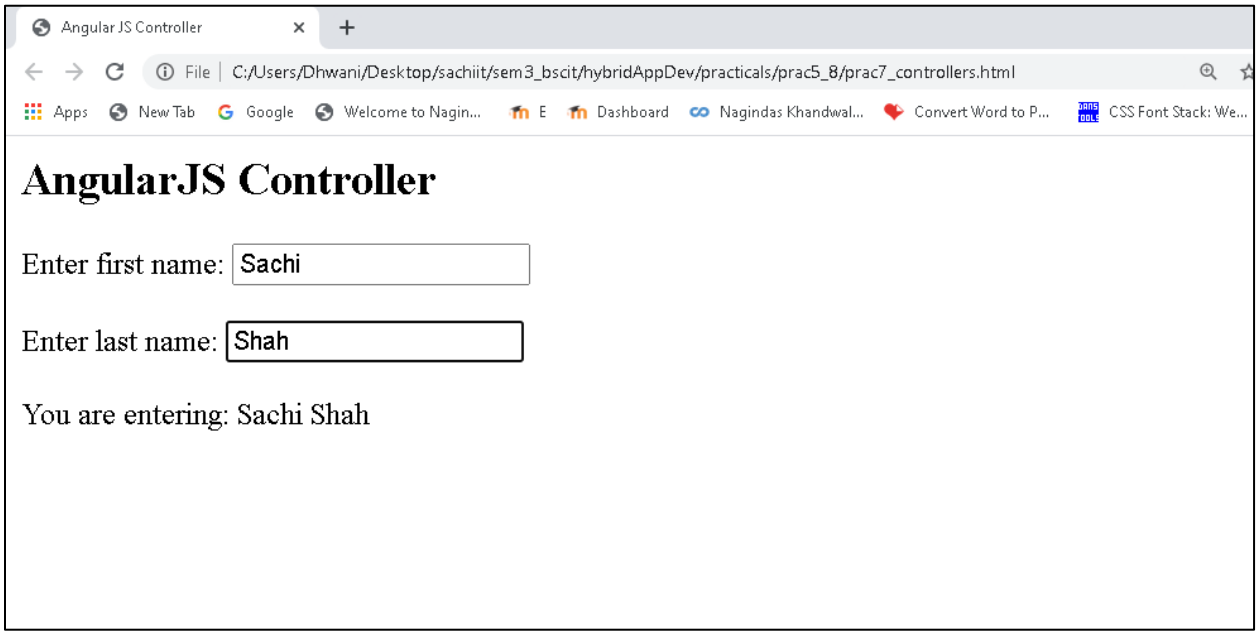
In general, a Controller shouldn't try to do too much. It should contain only the business logic needed for a single view. The most common way to keep Controllers slim is by encapsulating work that doesn't belong to controllers into services and then using these services in Controllers via dependency injection.

Code:

prac7_controllers.html

```
<html>
   <head>
      <title>Angular JS Controller</title>
      <script src = "https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js">
      </script>
   </head>
   <body>
      <h2>AngularJS Controller</h2>
      <!-- we declare a controller named studentController, using the ng-controller directive. -->
      <div ng-app = "mainApp" ng-controller = "studentController">
         Enter first name: <input type = "text" ng-model = "student.firstName"><br><br>
         Enter last name: <input type = "text" ng-model = "student.lastName"><br><br>
         You are entering: {{student.fullName()}}
      </div>

      <script>
         var mainApp = angular.module("mainApp", []);
         mainApp.controller('studentController', function($scope) {
            $scope.student = {
               firstName: "",
               lastName: "",

               fullName: function() {
                  var studentObject;
                  studentObject = $scope.student;
                  return studentObject.firstName + " " + studentObject.lastName;}
            };
         });
      </script>
   </body>
</html>
```

Output:


Angular JS Controller
File | C:/Users/Dhwani/Desktop/sachiit/sem3_bscit/hybridAppDev/practicals/prac5_8/prac7_controllers.html
Apps
New Tab
Google
Welcome to Nagin...
E
Dashboard
Nagindas Khandwal...
Convert Word to P...
CSS Font Stack: We...
AngularJS Controller
Enter first name: Sachi
Enter last name: Shah
You are entering: Sachi Shah

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 4

Aim: AngularJS Events

Theory:

AngularJS has its own HTML events directives. The event directives allow us to run AngularJS functions at certain user events. An AngularJS event will not overwrite an HTML event, both events will be executed.

Mouse Events

Mouse events occur when the cursor moves over an element, in this order:

- ng-mouseover
- ng-mouseenter
- ng-mousemove
- ng-mouseleave
- ng-mousedown
- ng-mouseup
- ng-click

The ng-click Directive

The **ng-click** directive defines AngularJS code that will be executed when the element is being clicked.

Toggle, True/False

If you want to show a section of HTML code when a button is clicked, and hide when the button is clicked again, like a dropdown menu, make the button behave like a toggle switch

$event Object

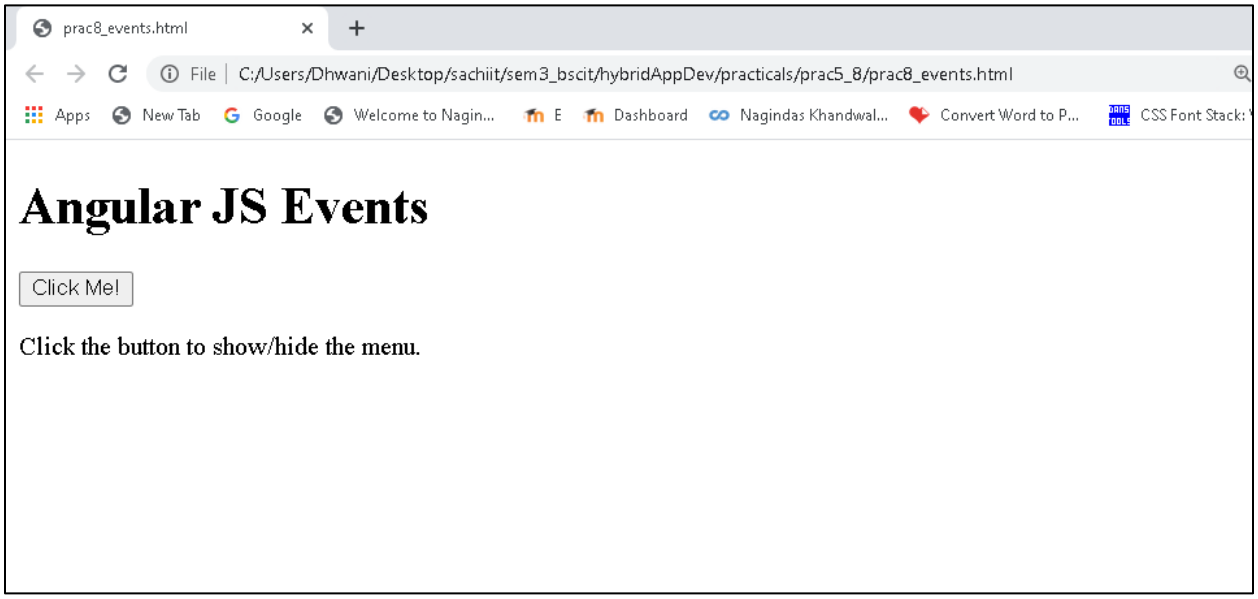
You can pass the **$event** object as an argument when calling the function.

Code:

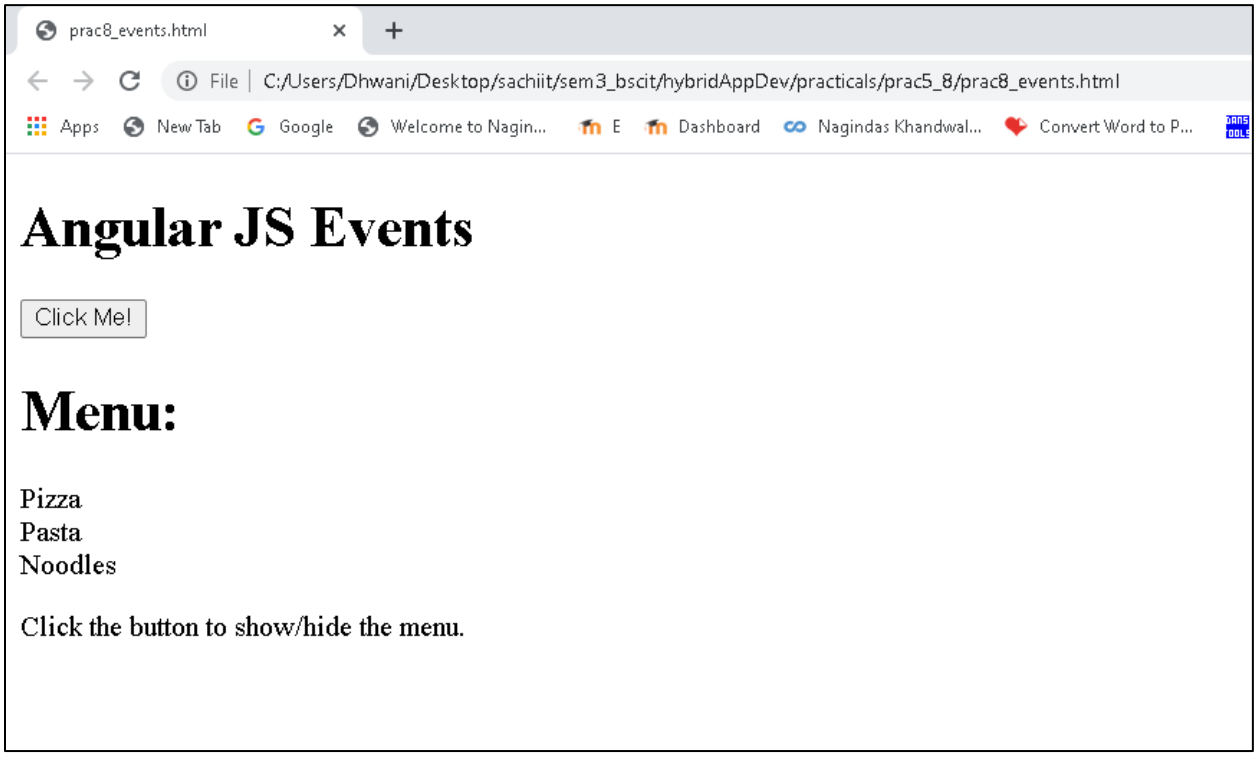
prac8_events.html

```
<!DOCTYPE html>
<html>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></script>
    <body>
        <h1>Angular JS Events</h1>
        <div ng-app="myApp" ng-controller="myCtrl">
            <button ng-click="myFunc()">Click Me!</button>
            <div ng-show="showMe">
                <h1>Menu:</h1>
                <div>Pizza</div>
                <div>Pasta</div>
                <div>Noodles</div>
            </div>
        </div>
        <script>
            var app = angular.module('myApp', []);
            app.controller('myCtrl', function($scope) {
                // The showMe variable starts out as the Boolean value false.
                // The myFunc function sets the showMe variable to the opposite of what it is,
                // by using the ! (not) operator.
                $scope.showMe = false;
                $scope.myFunc = function() {
                    $scope.showMe = !$scope.showMe; }
            });
        </script>
        <p>Click the button to show/hide the menu.</p>
    </body>
</html>
```

Output: Before clicking click Me !



Output: After clicking click Me!



NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 5

Aim: Ionic Create and build first project or application (Android)

Theory:

Hybrid App Development helps to deploy the code on different platforms using one source code.

Ionic is open source framework used for developing mobile applications. It provides tools and services for building Mobile UI with native look.

Commands to setup Ionic project:

1. Npm install –g ionic Cordova
2. Ionic start hello_world blank
   (blank is the template, hello_world folder will be created)
3. cd hello_world
4. Ionic serve

Steps:

```
C:\Users\Dhwani>npm install -g ionic cordova
npm WARN deprecated ionic@5.4.16: The Ionic CLI now uses ▯  @i
npm WARN deprecated request@2.88.2: request has been deprecate
npm WARN deprecated har-validator@5.1.5: this library is no lo
[                .....] | extract:underscore: sill extract unders
[                .....] / refresh-package-json:ms: sill refresh-p
es\cordova\bin\cordova
C:\Program Files\nodejs\ionic -> C:\Program Files\nodejs\node_
+ cordova@10.0.0
+ ionic@5.4.16
added 634 packages from 321 contributors in 337.66s

C:\Users\Dhwani>ionic start hello_world blank

Pick a framework!
```

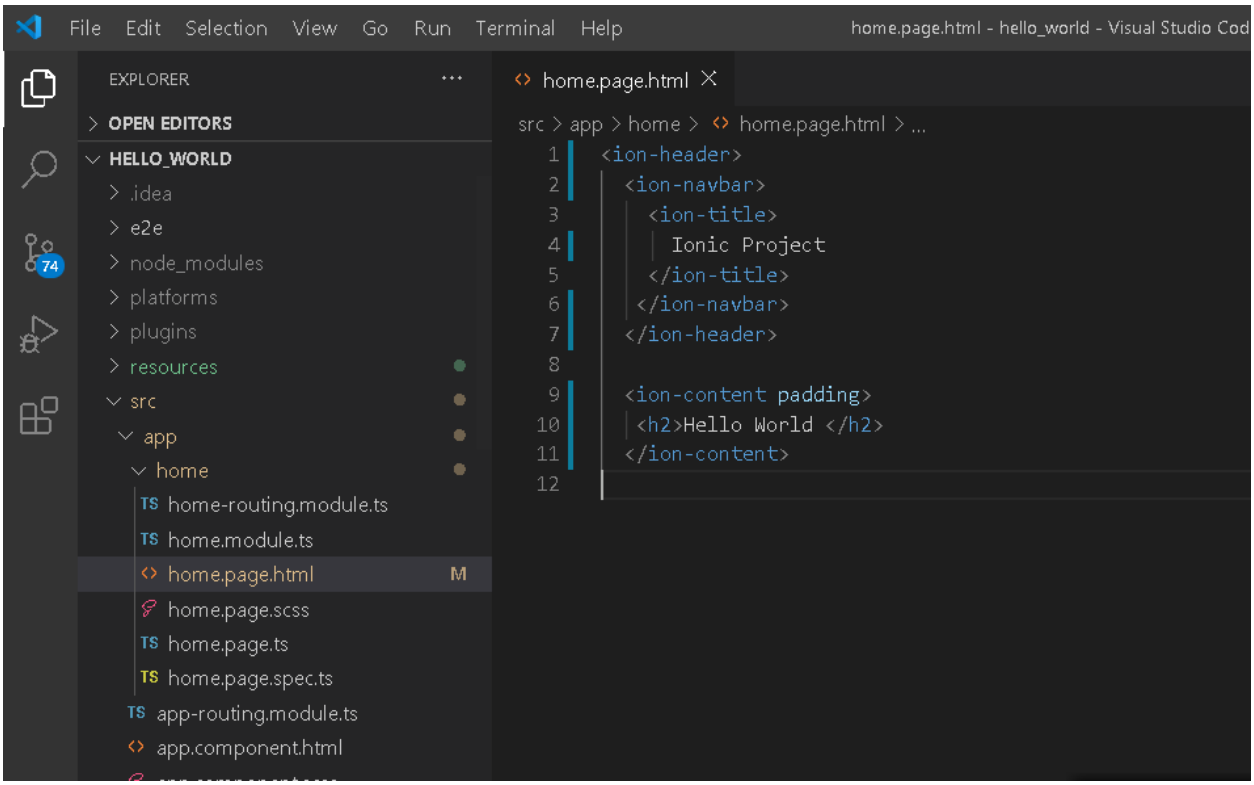
```
C:\Users\Dhwani>cd hello_world
```

```
C:\Users\Dhwani\hello_world>ionic serve
> ng.cmd run app:serve --host=localhost --port=8100
[ng] Compiling @angular/compiler/testing : es2015 as esm2015
[ng] Compiling @angular/core : es2015 as esm2015
[ng] Compiling @ionic-native/core : module as esm5
[ng] Compiling @angular/common : es2015 as esm2015
[ng] Compiling @ionic-native/splash-screen : module as esm5
[ng] Compiling @angular/core/testing : es2015 as esm2015
[ng] Compiling @ionic-native/status-bar : module as esm5
[ng] Compiling @angular/platform-browser : es2015 as esm2015
[ng] Compiling @angular/router : es2015 as esm2015
```

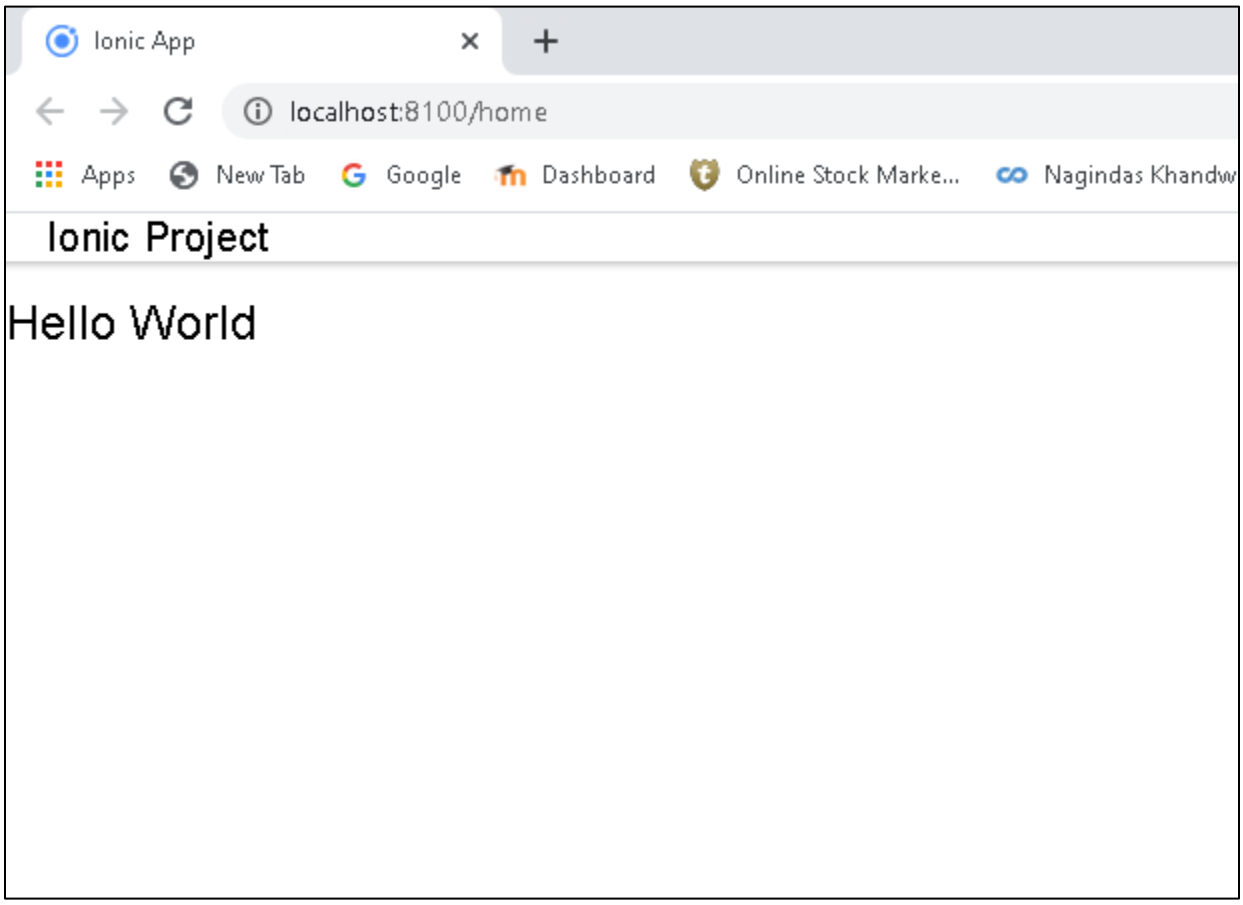
NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Code:



Output:


Ionic App
localhost:8100/home
Apps
New Tab
Google
Dashboard
Online Stock Marke...
Nagindas Khandw
Ionic Project
Hello World

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 6

Aim: Ionic Adding Cordova Android Platform

Theory:

Commands to setup Ionic project:

1. Npm install –g ionic Cordova
2. Ionic start hello_world blank
   (blank is the template, hello_world folder will be created)
3. cd hello_world
4. Ionic serve
5. Ionic Cordova build android –
6. Now to run the ionic Cordova app on android, we type
   Ionic Cordova run android
   If we have our phone connected to the machine, then we need to turn on the USB debugging under the Developer settings. This will export the app and will be viewable in your phone.
   Another way is to run by setting up an emulator in the Android Studio.

Steps:

```
C:\Users\Dhwani>npm install -g ionic cordova
npm WARN deprecated ionic@5.4.16: The Ionic CLI now uses 0  @i
npm WARN deprecated request@2.88.2: request has been deprecate
npm WARN deprecated har-validator@5.1.5: this library is no lo
[                ......] | extract:underscore: sill extract unders
[                ......] / refresh-package-json:ms: sill refresh-p
es\cordova\bin\cordova
C:\Program Files\nodejs\ionic -> C:\Program Files\nodejs\node_
+ cordova@10.0.0
+ ionic@5.4.16
added 634 packages from 321 contributors in 337.66s

C:\Users\Dhwani>ionic start hello_world blank

Pick a framework!
```

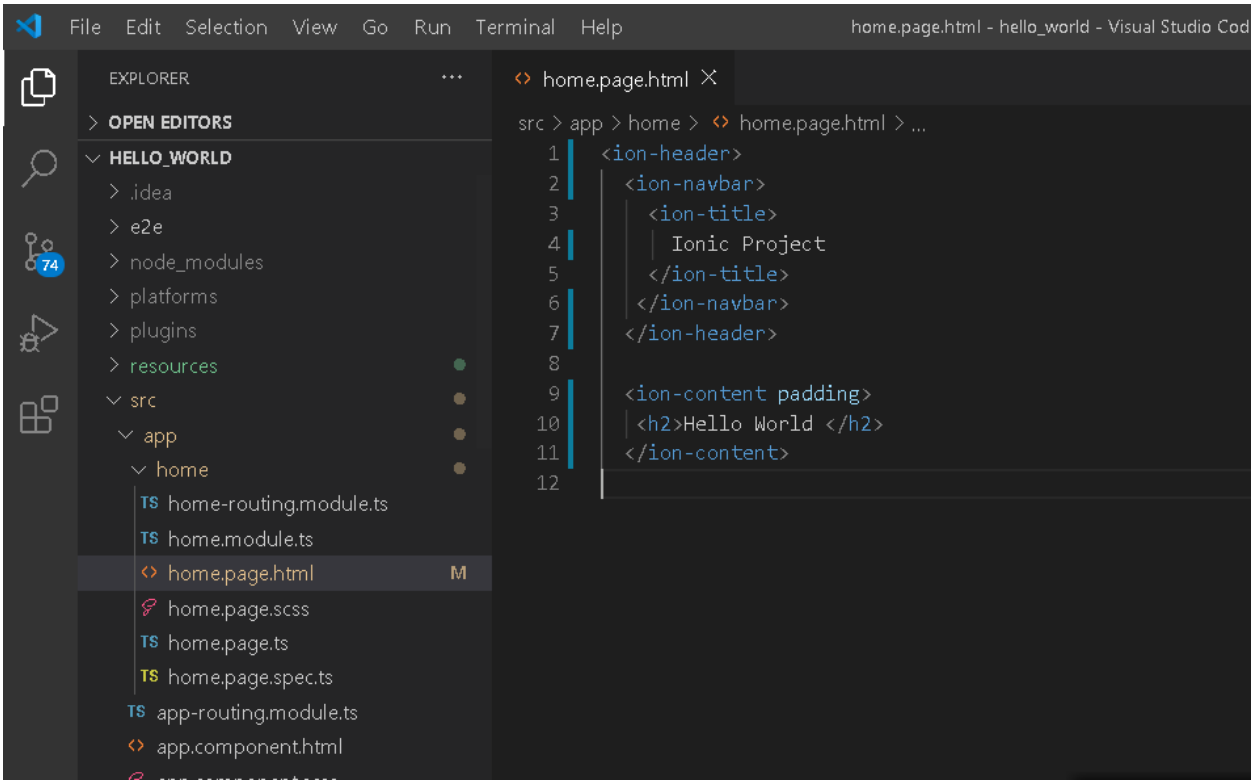
```
C:\Users\Dhwani>cd hello_world
```

```
C:\Users\Dhwani\hello_world>ionic serve
> ng.cmd run app:serve --host=localhost --port=8100
[ng] Compiling @angular/compiler/testing : es2015 as esm2015
[ng] Compiling @angular/core : es2015 as esm2015
[ng] Compiling @ionic-native/core : module as esm5
[ng] Compiling @angular/common : es2015 as esm2015
[ng] Compiling @ionic-native/splash-screen : module as esm5
[ng] Compiling @angular/core/testing : es2015 as esm2015
[ng] Compiling @ionic-native/status-bar : module as esm5
[ng] Compiling @angular/platform-browser : es2015 as esm2015
[ng] Compiling @angular/router : es2015 as esm2015
```

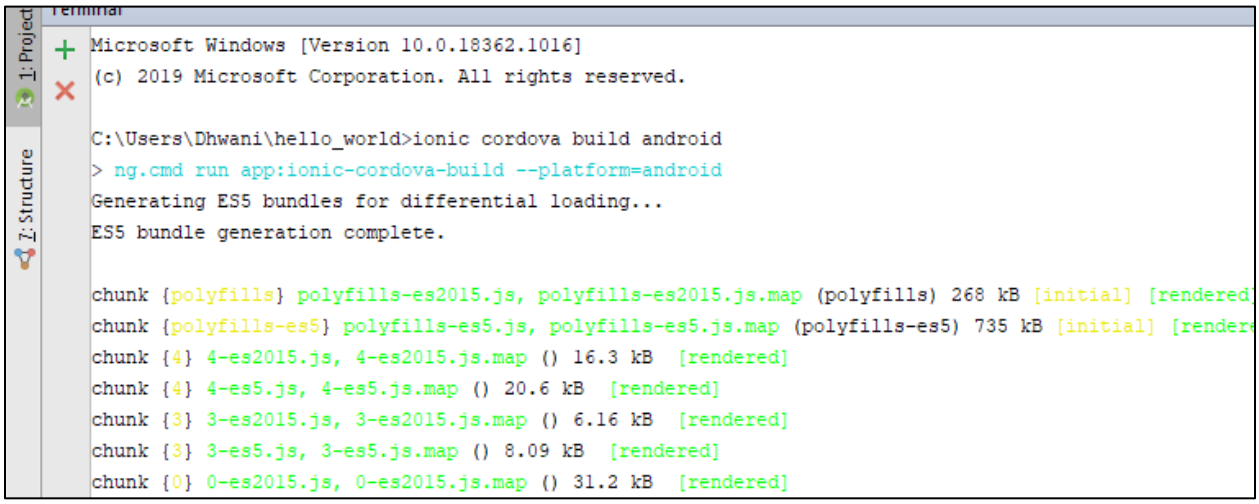
NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

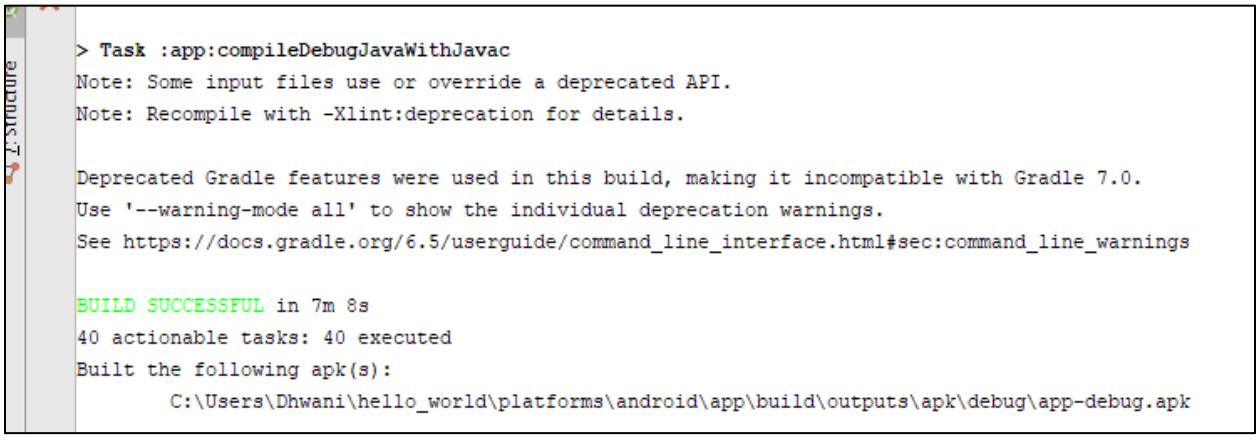
CLASS: SYIT **OLD ROLL NO: 3066**

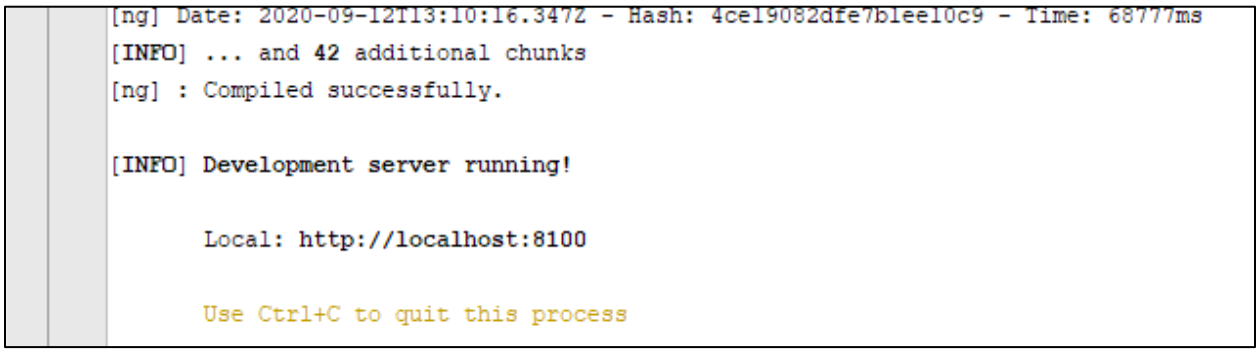
Code:



Steps to add Cordova Android Platform



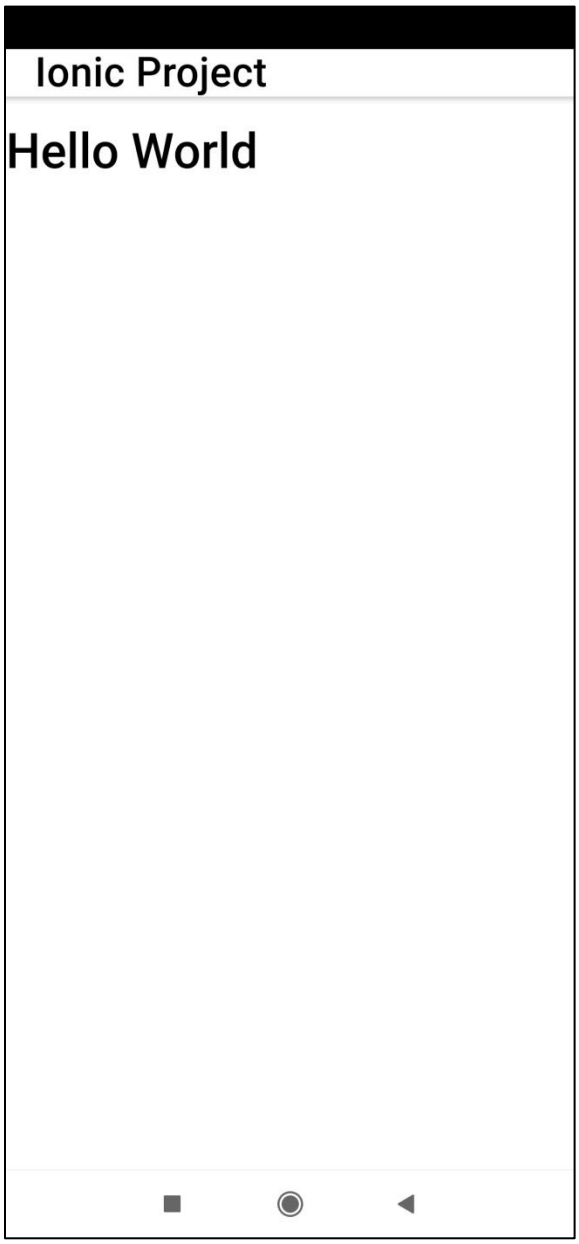




```
C:\Users\Dhwani\hello_world>ionic cordova run android
[INFO] Hardware device(s) found for android. Using --device.
> ng.cmd run app:ionic-cordova-build --platform=android
Generating ES5 bundles for differential loading...
ES5 bundle generation complete.
```

```
> cordova.cmd build android --device
Checking Java JDK and Android SDK versions
ANDROID_SDK_ROOT=undefined (recommended setting)
ANDROID_HOME=undefined (DEPRECATED)
Using Android SDK: C:\Users\Dhwani\AppData\Local\Android\sdk
Subproject Path: CordovaLib
Subproject Path: app


Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.5/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 8s
40 actionable tasks: 40 up-to-date
Built the following apk(s):
        C:\Users\Dhwani\hello_world\platforms\android\app\build\outputs\apk\debug\app-debug.apk
> native-run.cmd android --app platforms\android\app\build\outputs\apk\debug\app-debug.apk --device
[native-run] Selected hardware device d9cec42f
[native-run] Installing platforms\android\app\build\outputs\apk\debug\app-debug.apk...
[native-run] Starting application activity io.ionic.starter/io.ionic.starter.MainActivity...
[native-run] Run Successful
```

Output on Android device:



NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

PRACTICAL 7

AIM: Ionic Create, Generate and Add Pages

Theory:

1. Creating a new folder.

```
C:\Users\Dhwani>mkdir angular_ionic

C:\Users\Dhwani>cd angular_ionic

C:\Users\Dhwani\angular_ionic>
```

2. Starting a New Ionic App

   - To start a new app, open your terminal/command prompt and run:

     ionic start MyIonicProject tutorial --type=ionic-angular

     A. **start** will tell the CLI create a new app.
     B. **MyIonicProject** will be the directory name and the app name from your project.
     C. **tutorial** will be the starter template for your project.

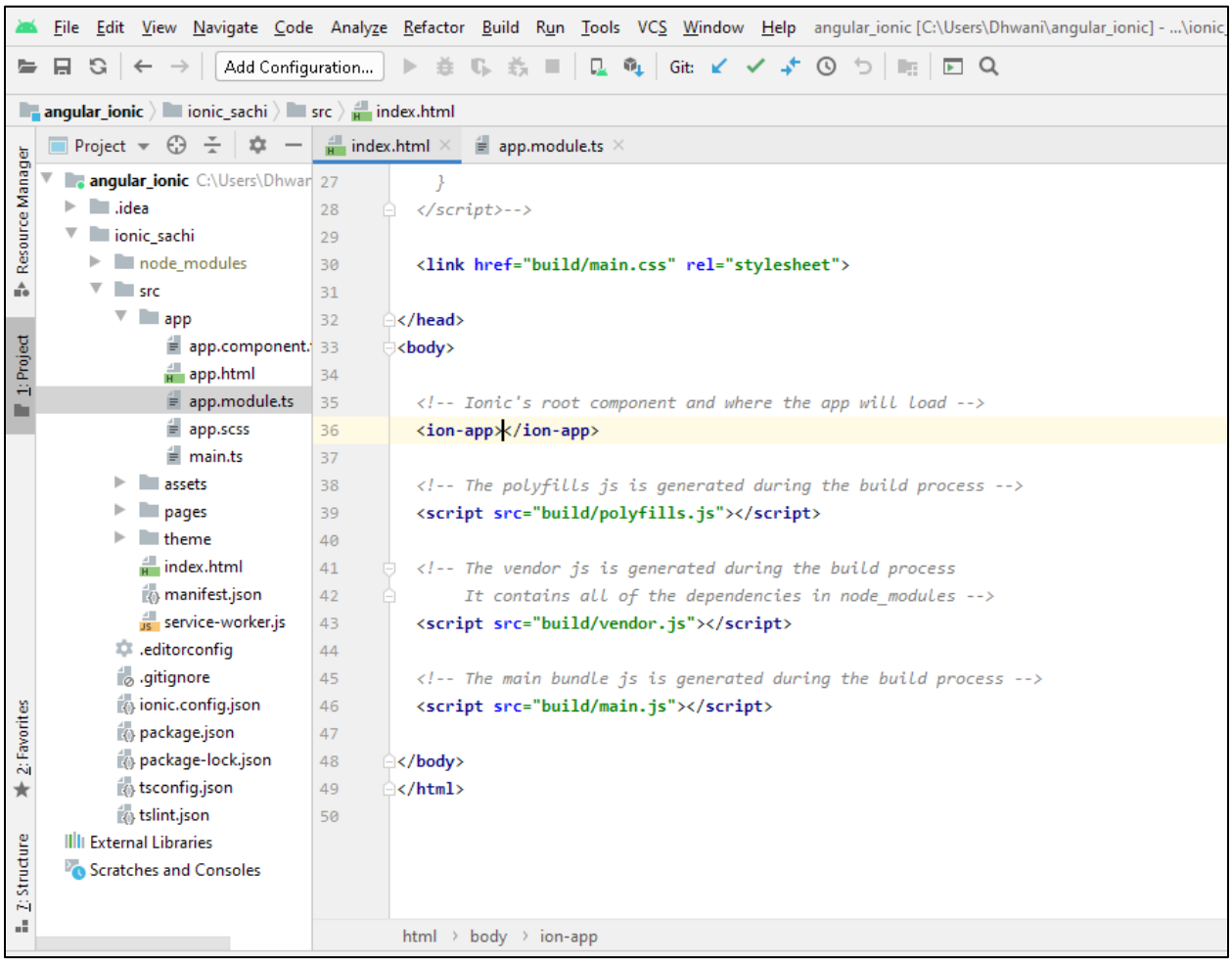
3. Along with creating your project, this will also install node modules for the application, and prompt you if you want Cordova set up.

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

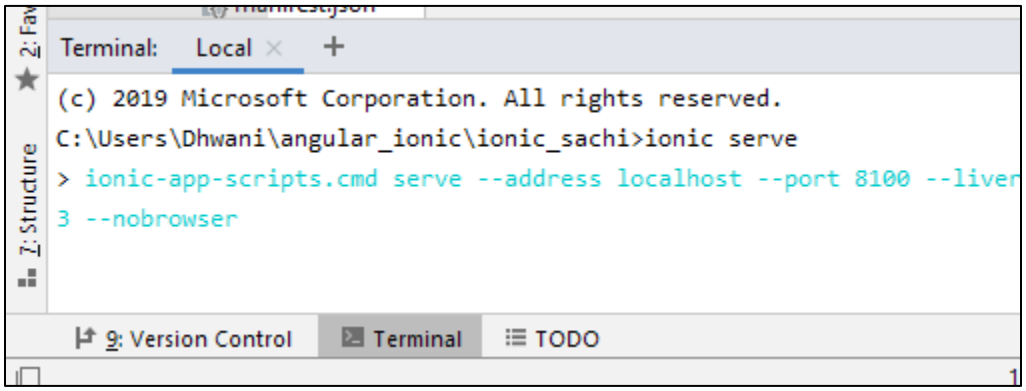
```
C:\Users\Dhwani\angular_ionic>
C:\Users\Dhwani\angular_ionic>ionic start ionic_sachi tutorial --type=ionic-angular
√ Preparing directory .\ionic_sachi - done!
√ Downloading and extracting tutorial starter - done!

Installing dependencies may take several minutes.
```

Code:

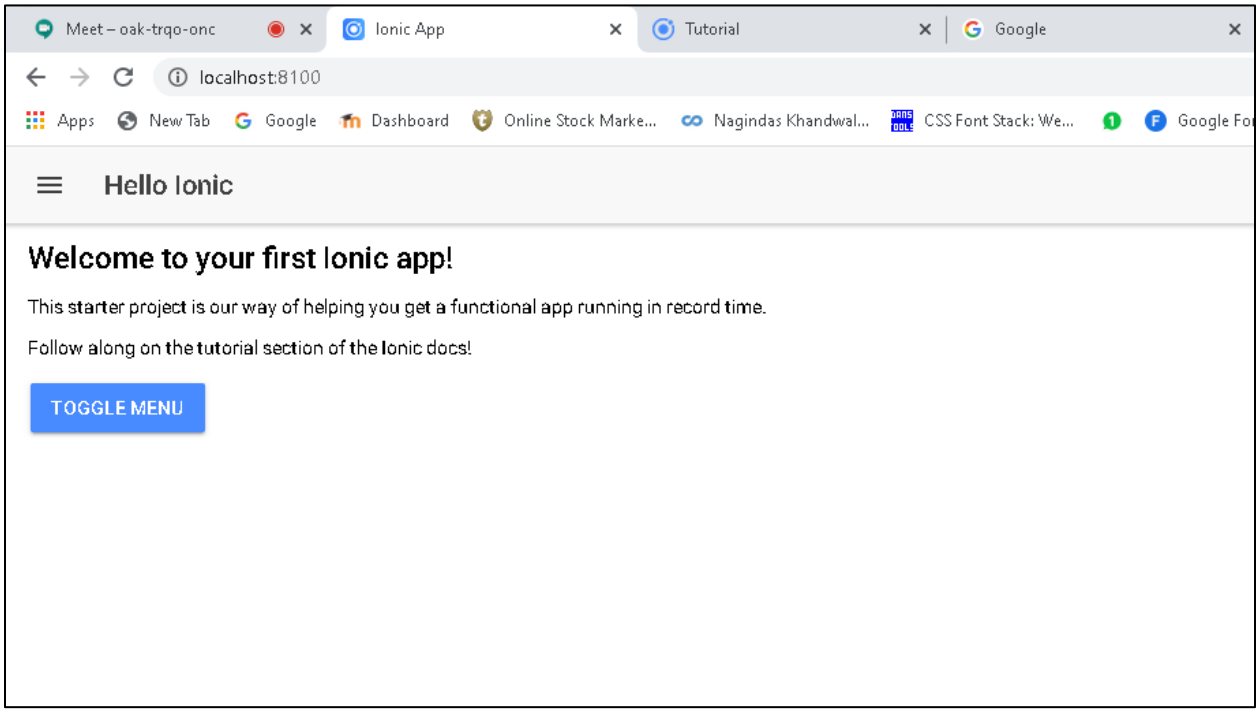


Viewing the app in a browser

Now, you can cd into the folder that was created. To get a quick preview of your app in the browser, use the serve command.



Output:

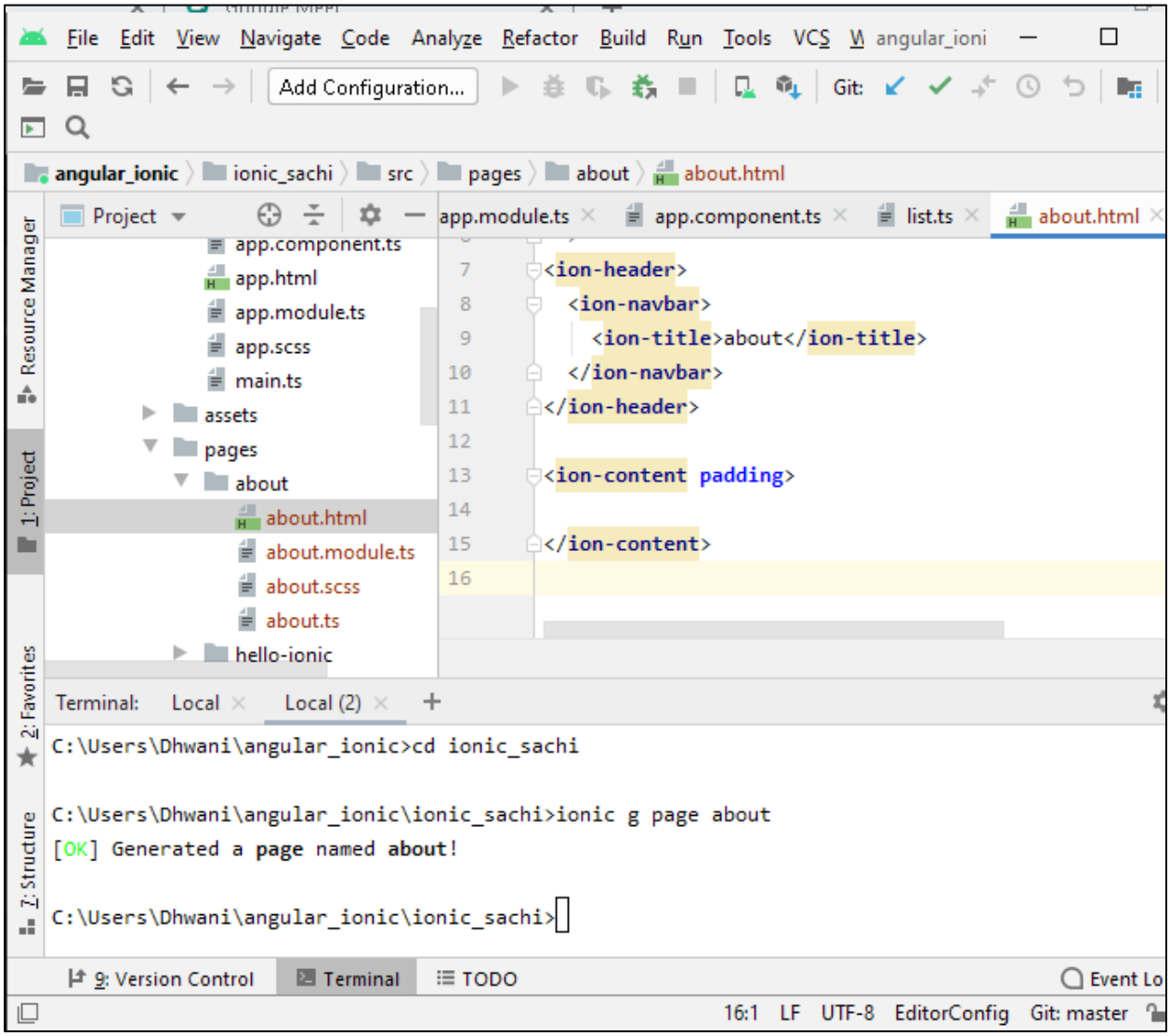


Creating a new page:

- **ionic generate <type> <name> [options]**
- This command uses the Angular CLI to generate features such as pages, components, directives, services, etc.

Command:

ionic g page about

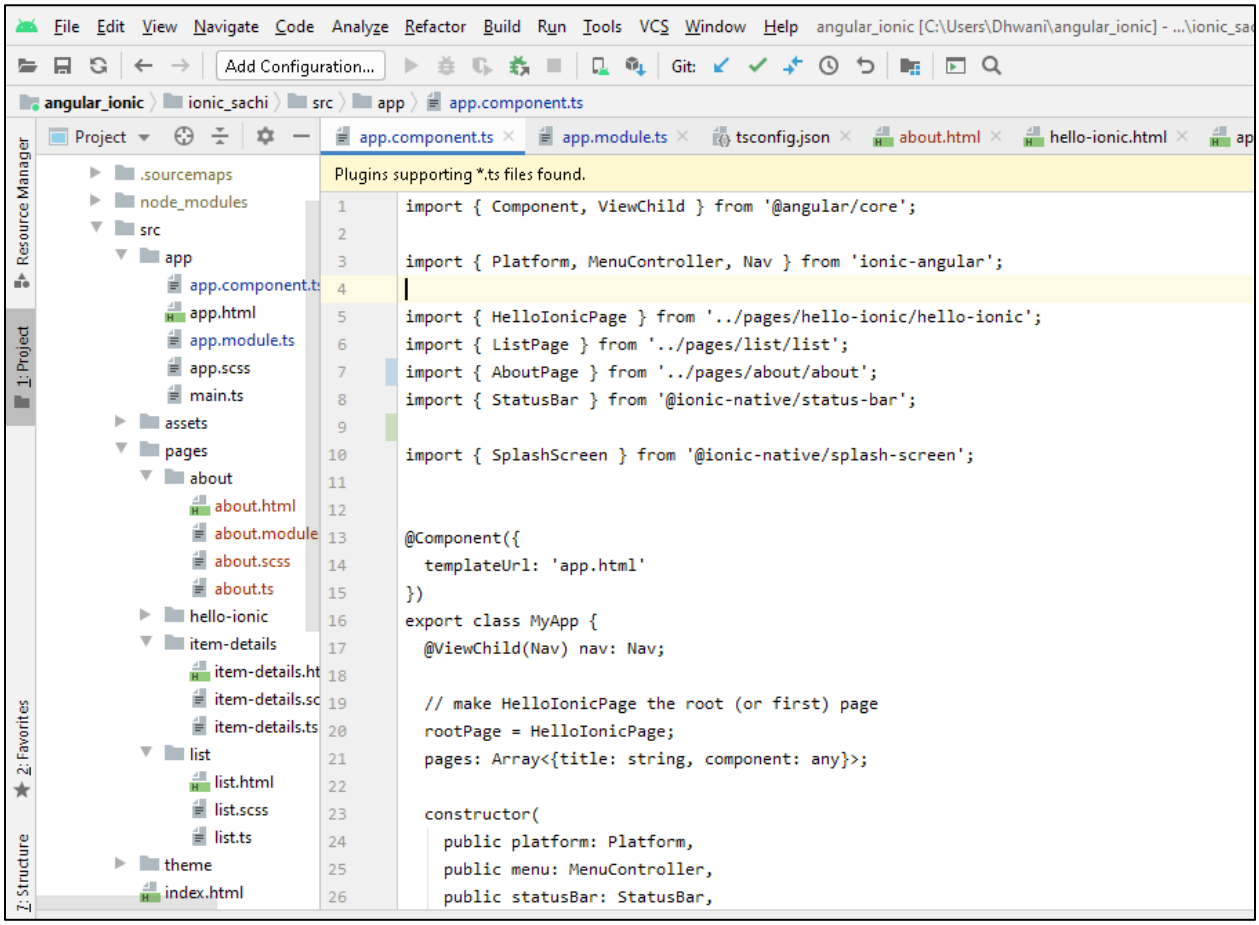


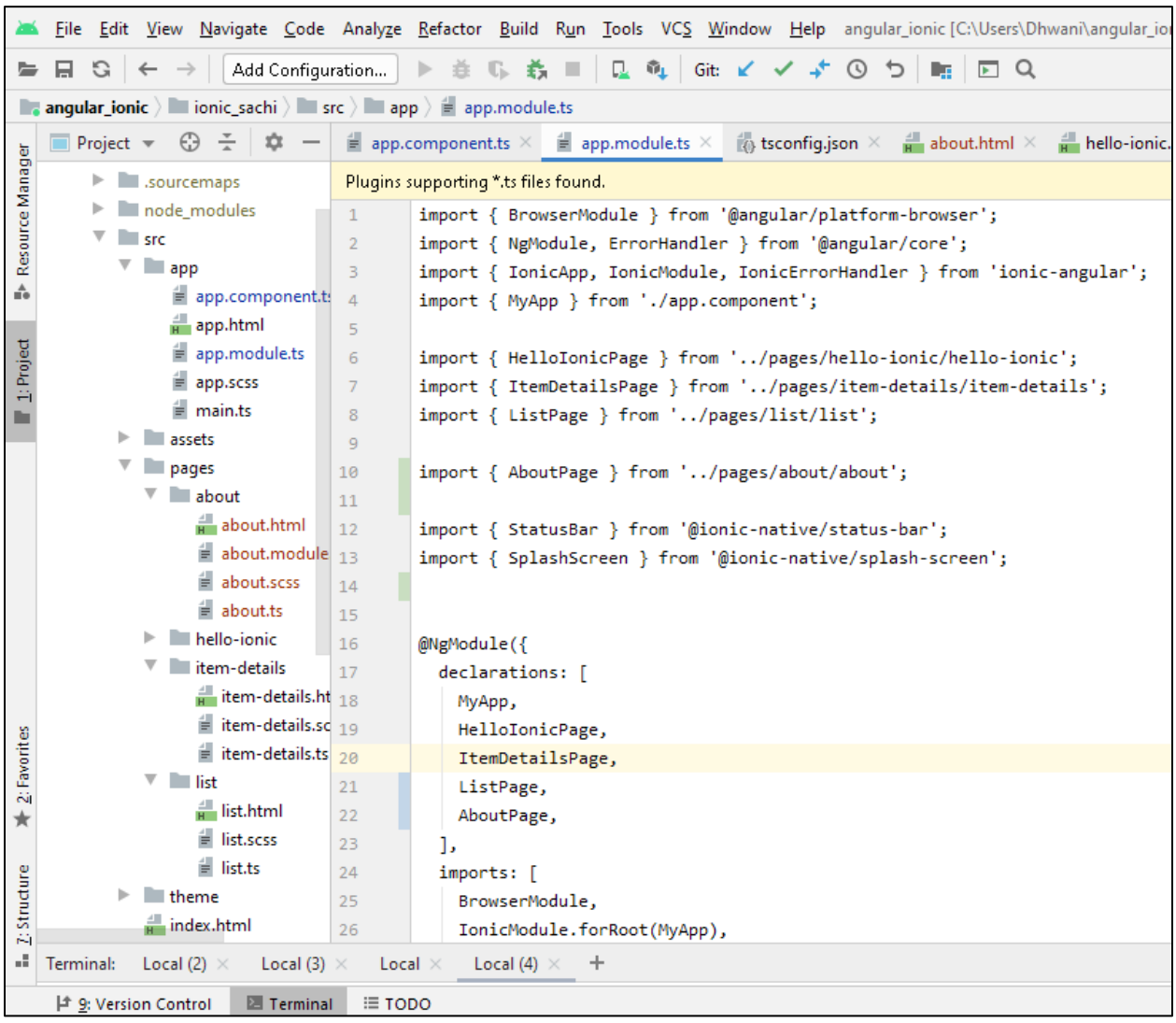
NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Code:

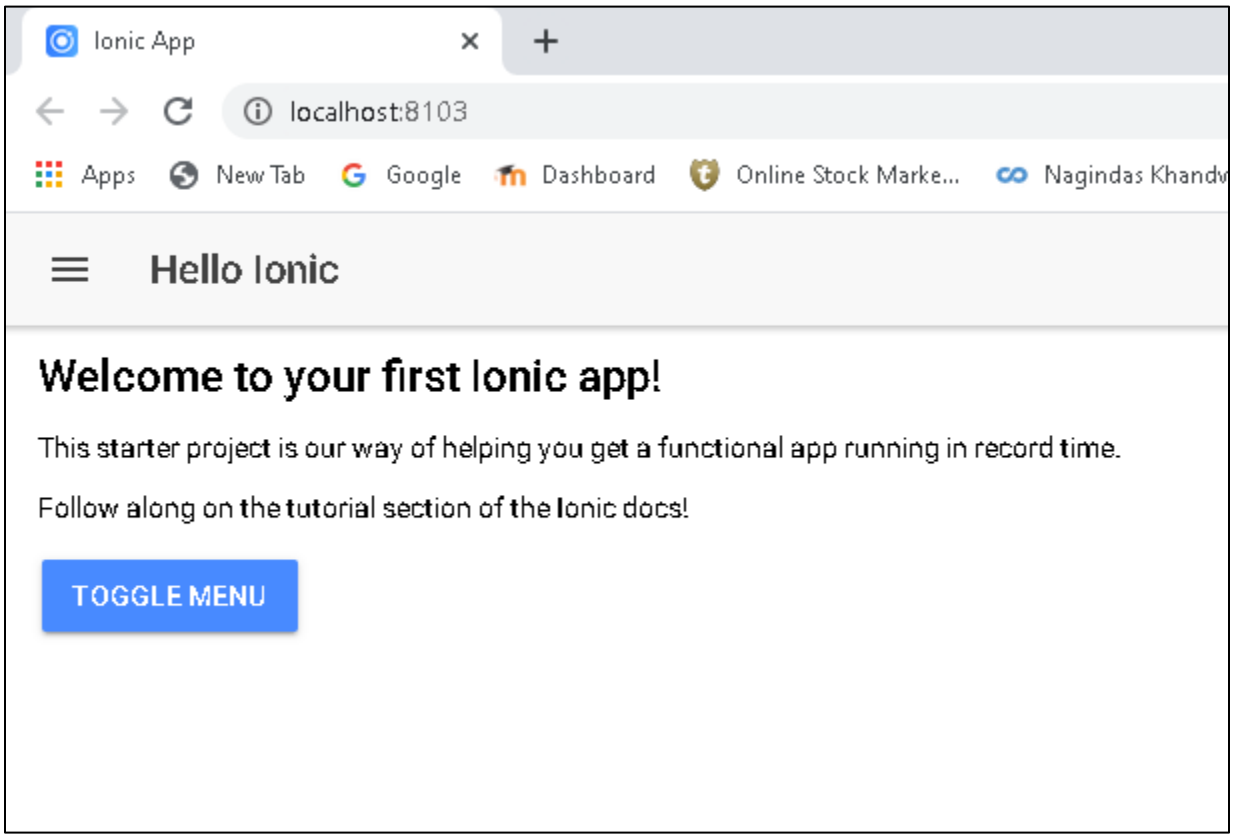
import {AboutPage} from '../about/about';

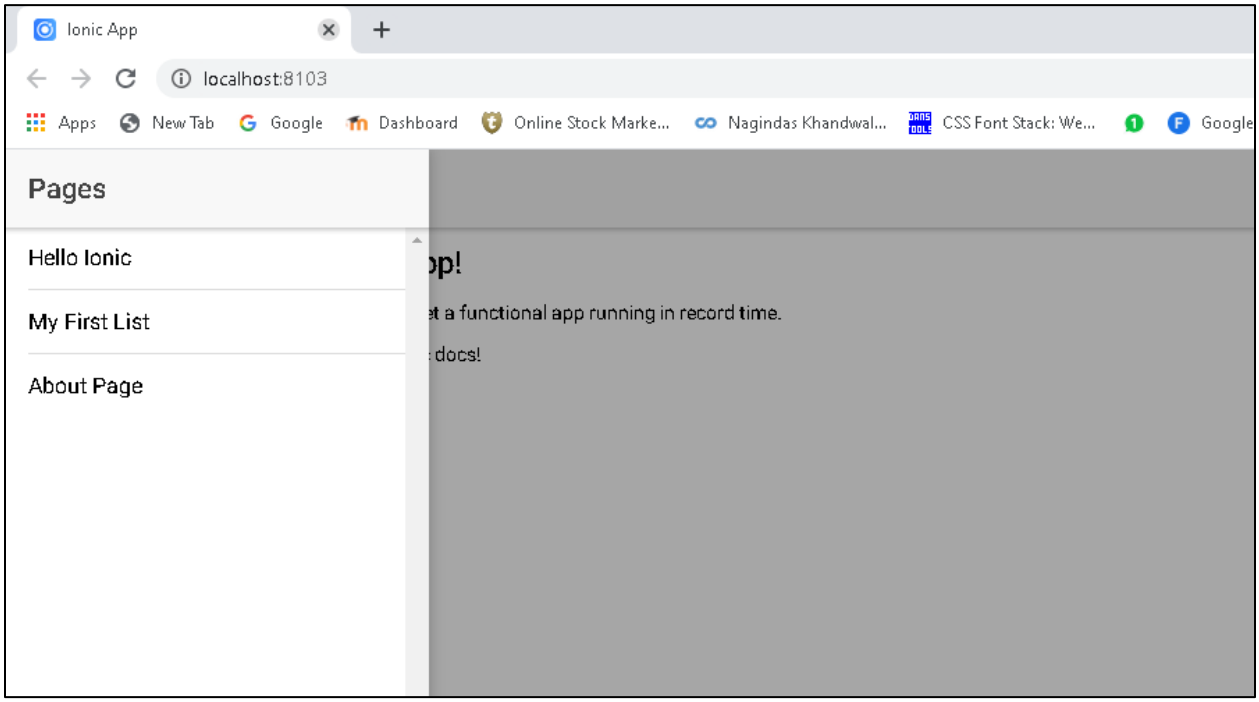


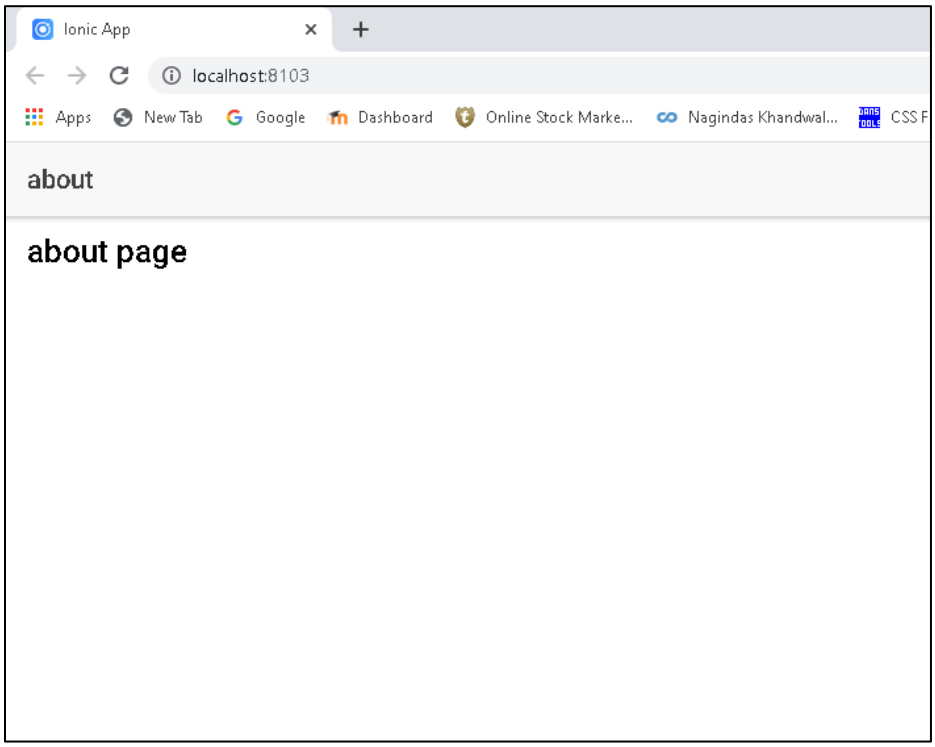



```
C:\Users\Dhwani\angular_ionic\ionic_sachi>ionic serve
> ionic-app-scripts.cmd serve --address localhost --port 8103 --livereload-port 35732 --dev-logger-port
[app-scripts] [15:59:25]  ionic-app-scripts 3.2.4
[app-scripts] [15:59:25]  watch started ...
[app-scripts] [15:59:25]  build dev started ...
[app-scripts] [15:59:25]  clean started ...
[app-scripts] [15:59:25]  clean finished in 145 ms
[app-scripts] [15:59:25]  copy started ...
[app-scripts] [15:59:26]  deeplinks started ...
[app-scripts] [15:59:26]  deeplinks finished in 31 ms
[app-scripts] [15:59:26]  transpile started ...
```

Output:


Ionic App
localhost:8103
Apps
New Tab
Google
Dashboard
Online Stock Marke...
Nagindas Khandv
Hello Ionic
Welcome to your first Ionic app!
This starter project is our way of helping you get a functional app running in record time.
Follow along on the tutorial section of the Ionic docs!
TOGGLE MENU


Ionic App
localhost:8103
Apps
New Tab
Google
Dashboard
Online Stock Marke...
Nagindas Khandwal...
CSS Font Stack: We...
Google
Pages
Hello Ionic
My First List
About Page
op!
et a functional app running in record time.
docs!


Ionic App
localhost:8103
Apps
New Tab
Google
Dashboard
Online Stock Marke...
Nagindas Khandwal...
about
about page

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Practical 8

Aim: Ionic Use Tabs starter command template and options

Theory:

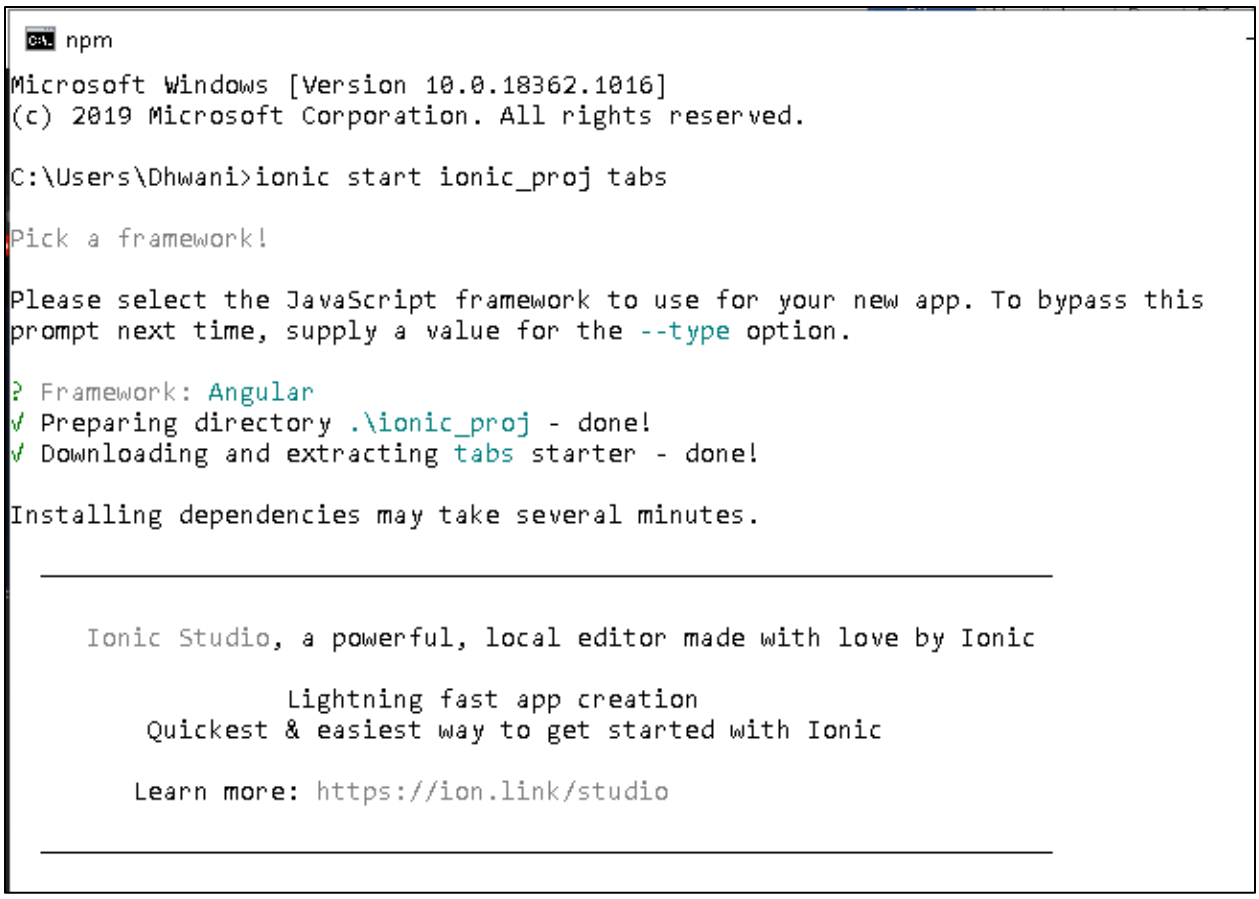
Angular templates are dynamic. When **Angular** renders them, it transforms the DOM according to the instructions given by directives

Along with the tutorial template, Ionic also provides the following official templates:

- tabs : a simple 3 tab layout
- sidemenu: a layout with a swipable menu on the side
- blank: a bare starter with a single page
- super: starter project with over 14 ready to use page designs
- tutorial: a guided starter project

If you don’t specify a template at the start, you will be prompted to pick one.

Creating a project with **tabs** template



Code for Tab1:

tab1.page.html | tab2.page.html | tab1.page.ts

src > app > tab1 > tab1.page.html > ion-content > ion-header

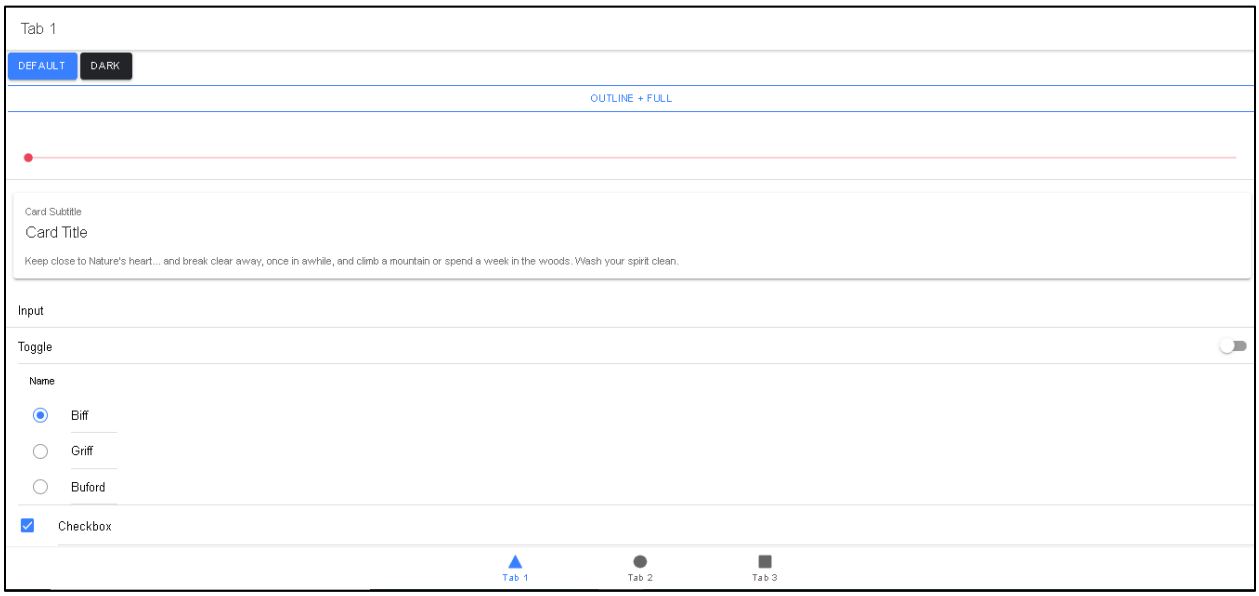
```
<ion-header [translucent]="true">
  <ion-toolbar>
    <ion-title>
      Tab 1
    </ion-title>
  </ion-toolbar>
</ion-header>

<ion-content [fullscreen]="true">
  <ion-header collapse="condense">
    <ion-toolbar>
      <ion-title size="large">Tab 1</ion-title>
    </ion-toolbar>
  </ion-header>

  <ion-button>Default</ion-button>
  <ion-button (click)='changeTab()' color="dark">Dark</ion-button>
  <ion-button expand="full" fill="outline">Outline + Full</ion-button>

  <ion-list>
    <ion-item>
      <ion-range color="danger" pin="true"></ion-range>
    </ion-item>
  </ion-list>

  <ion-card>
    <ion-card-header>
      <ion-card-subtitle>Card Subtitle</ion-card-subtitle>
      <ion-card-title>Card Title</ion-card-title>
    </ion-card-header>

    <ion-card-content>
      Keep close to Nature's heart... and break clear away, once in awhile,
      and climb a mountain or spend a week in the woods. Wash your spirit clean.
    </ion-card-content>
  </ion-card>

  <ion-list>
    <ion-item>
      <ion-label>Input</ion-label>
```

```
  <ion-list>
    <ion-item>
      <ion-label>Input</ion-label>
      <ion-input></ion-input>
    </ion-item>
    <ion-item>
      <ion-label>Toggle</ion-label>
      <ion-toggle slot="end"></ion-toggle>
    </ion-item>
    <ion-item>
      <ion-radio-group value="biff">
        <ion-list-header>
          <ion-label>Name</ion-label>
        </ion-list-header>

        <ion-item>
          <ion-label>Biff</ion-label>
          <ion-radio slot="start" value="biff"></ion-radio>
        </ion-item>

        <ion-item>
          <ion-label>Griff</ion-label>
          <ion-radio slot="start" value="griff"></ion-radio>
        </ion-item>

        <ion-item>
          <ion-label>Buford</ion-label>
          <ion-radio slot="start" value="buford"></ion-radio>
        </ion-item>
      </ion-radio-group>
    </ion-item>
    <ion-item>
      <ion-label>Checkbox</ion-label>
      <ion-checkbox slot="start"></ion-checkbox>
    </ion-item>
  </ion-list>

</ion-content>
```

NAME: SACHI VIRESH SHAH **NEW ROLL NO: 381**

CLASS: SYIT **OLD ROLL NO: 3066**

Output for Tab1:



Code for Tab2:

tab1.page.html | tab2.page.html | tab1.page.ts

src > app > tab2 > tab2.page.html > ion-content > ion-item > ion-input#myinput.ion-text-center

```
<ion-header>
  <ion-toolbar color="dark">
    <ion-title class="ion-text-center">To-Do App</ion-title>
  </ion-toolbar>

  <ion-toolbar>
    <ion-title class="ion-text-center">MY LIST</ion-title>
  </ion-toolbar>
</ion-header>

<ion-content [fullscreen]="true">
  <ion-item>
    <ion-input id="myinput" [(ngModel)]="todo" placeholder="Insert your todo here" type="text"
    class="ion-text-center"></ion-input>
    <ion-icon class="item-note" name="add" slot="end" (click)="add()"></ion-icon>
  </ion-item>

  <ion-list>
    <ion-item *ngFor="let item of todos">
      <ion-item>
        {{item}}
      </ion-item>

      <ion-item side="right" slot="end">
        <ion-button color="danger" (click)="delete(item)">
          <ion-icon name="trash" ></ion-icon>
        </ion-button>
      </ion-item>
    </ion-item>
  </ion-list>
</ion-content>
```

Output for Tab2: