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. Gunveen Kaur Bindra

Roll No: 368 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 fulfilment 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: 368**

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: Gunveen Bindra **Old Roll No:3001**
Class: SYIT **New Roll No: 368**

Hybrid App Development

Practical - 01

Aim: To write a program for AngularJS Data Binding

Theory:

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

Data Model - AngularJS applications usually have a data model. The data model is a collection of data available for the application.

HTML View

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: Gunveen Bindra **Old Roll No:3001**
Class: SYIT **New Roll No: 368**
CODE:

c: > Users > simra > OneDrive > Desktop > gunveen > SYIT > HAD > Practical1 > databinding.html > html

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

<div ng-app="myApp" ng-controller="myCtrl">
  Name: <input ng-model="firstname">
  <h1>{{firstname}}</h1>
</div>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
  $scope.firstname = "Gunveen";
  $scope.lastname = "Bindra";
});
</script>
</body>
</html>

```

OUTPUT:

Name: Gunveen Bindra **Old Roll No:3001**
Class: SYIT **New Roll No: 368**

Practical - 02

Aim: To write a program for 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

AngularJS directives are extended HTML attributes with the prefix ng-.

- The **ng-app** directive initializes an AngularJS application.
- The **ng-init** directive initializes application data.
- The **ng-model** directive binds the value of HTML controls (input, select, textarea) to application data.

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.

CODE:

directive.html

C: > Users > simra > OneDrive > Desktop > gunveen > SYIT > HAD > Practical1 > directive.html > ...

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

    <div ng-app="" ng-init="names=[
    {name:'Gunveen',country:'Norway'},
    {name:'Jasleen',country:'Sweden'},
    {name:'Simran',country:'Denmark'}]">

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

    </div>
</body>
</html>
```

OUTPUT:

Name: Gunveen Bindra **Old Roll No:3001**
Class: SYIT **New Roll No: 368**

Practical - 03

Aim: To write a program for AngularJS Controllers

Theory:

AngularJS Controllers

- AngularJS applications are controlled by controllers.
- AngularJS controllers **control the data** of AngularJS applications.
- AngularJS controllers are regular **JavaScript Objects**.
- The **ng-controller** directive defines the application controller.
- A controller is a **JavaScript Object**, created by a standard JavaScript **object constructor**.

In this program, the AngularJS application is defined by ng-app="myApp". The application runs inside the . The ng-controller="myCtrl" attribute is an AngularJS directive. It defines a controller. The myCtrl function is a JavaScript function. AngularJS will invoke the controller with a $scope object. In AngularJS, $scope is the application object (the owner of application variables and functions). The controller creates two properties (variables) in the scope (firstName and lastName). The ng-model directives bind the input fields to the controller properties (firstName and lastName).

CODE:

Controllers.html

c: > Users > simra > OneDrive > Desktop > gunveen > SYIT > HAD > Practical1 > Controllers.html > ...

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

<div ng-app="myApp" ng-controller="myCtrl">

First Name: <input type="text" ng-model="firstName"><br>
Last Name: <input type="text" ng-model="lastName"><br>
<br>
Full Name: {{firstName + " " + lastName}}

</div>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
  $scope.firstName = "Gunveen";
  $scope.lastName = "Bindra";
});
</script>
</body>
</html>

```

OUTPUT:

Practical - 04

Aim: To write a program for AngularJS Events

Theory:

AngularJS Events

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.

You can add AngularJS event listeners to your HTML elements by using one or more of these directives:

- ng-blur
- ng-change
- ng-click
- ng-copy
- ng-cut
- ng-dblclick
- ng-focus
- ng-keydown
- ng-keypress
- ng-keyup
- ng-mousedown
- ng-mouseenter
- ng-mouseleave
- ng-mousemove
- ng-mouseover
- ng-mouseup
- ng-paste

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

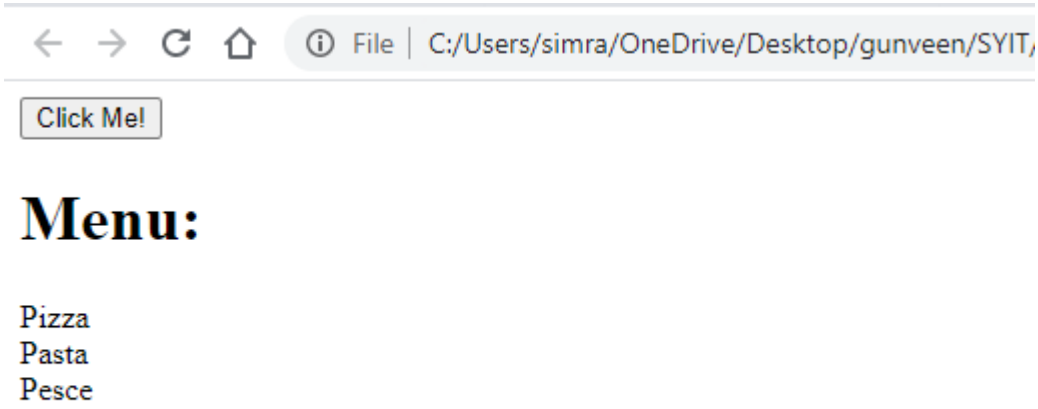
You can pass the $event object as an argument when calling the function.

CODE:

C: > Users > simra > OneDrive > Desktop > gunveen > SYIT > HAD > Practical1 > <> Events.html > ...

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

  <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>Pesce</div>
    </div>

    </div>
    <script>
    var app = angular.module('myApp', []);
    app.controller('myCtrl', function($scope) {
      $scope.showMe = false;
      $scope.myFunc = function() {
        $scope.showMe = !$scope.showMe;
      }
    });
    </script>
    </body>
</html>
```

OUTPUT:


File | C:/Users/simra/OneDrive/Desktop/gunveen/SYIT/
Click Me!
Menu:
Pizza
Pasta
Pesce

Practical - 05

Aim: Ionic 4-Create and build first project or application (Android and iOS)

Theory:

Ionic is open source framework used for developing mobile applications. It provides tools and services for building Mobile UI with native look and feel. Ionic framework needs native wrapper to be able to run on mobile devices. This is an introductory tutorial, which covers the basics of the Ionic Open Source Framework and explains how to deal with its various components and sub-components.

Steps to install Ionic and build your first project.

Step1: Go to command prompt and check if npm is installed or not by using the following command:

npm – v

This shows the version of npm, if it is not installed then install npm.

You can also check the node version by typing the command

node

Step 2: npm install -g ionic cordova.

Step 3: ionic start hello_world blank

Here, we have chosen blank as a template.

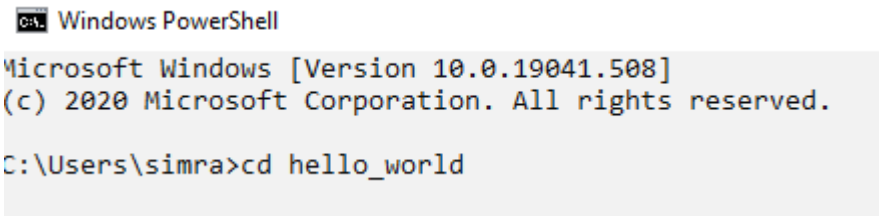
This command will create a folder name hello_world.

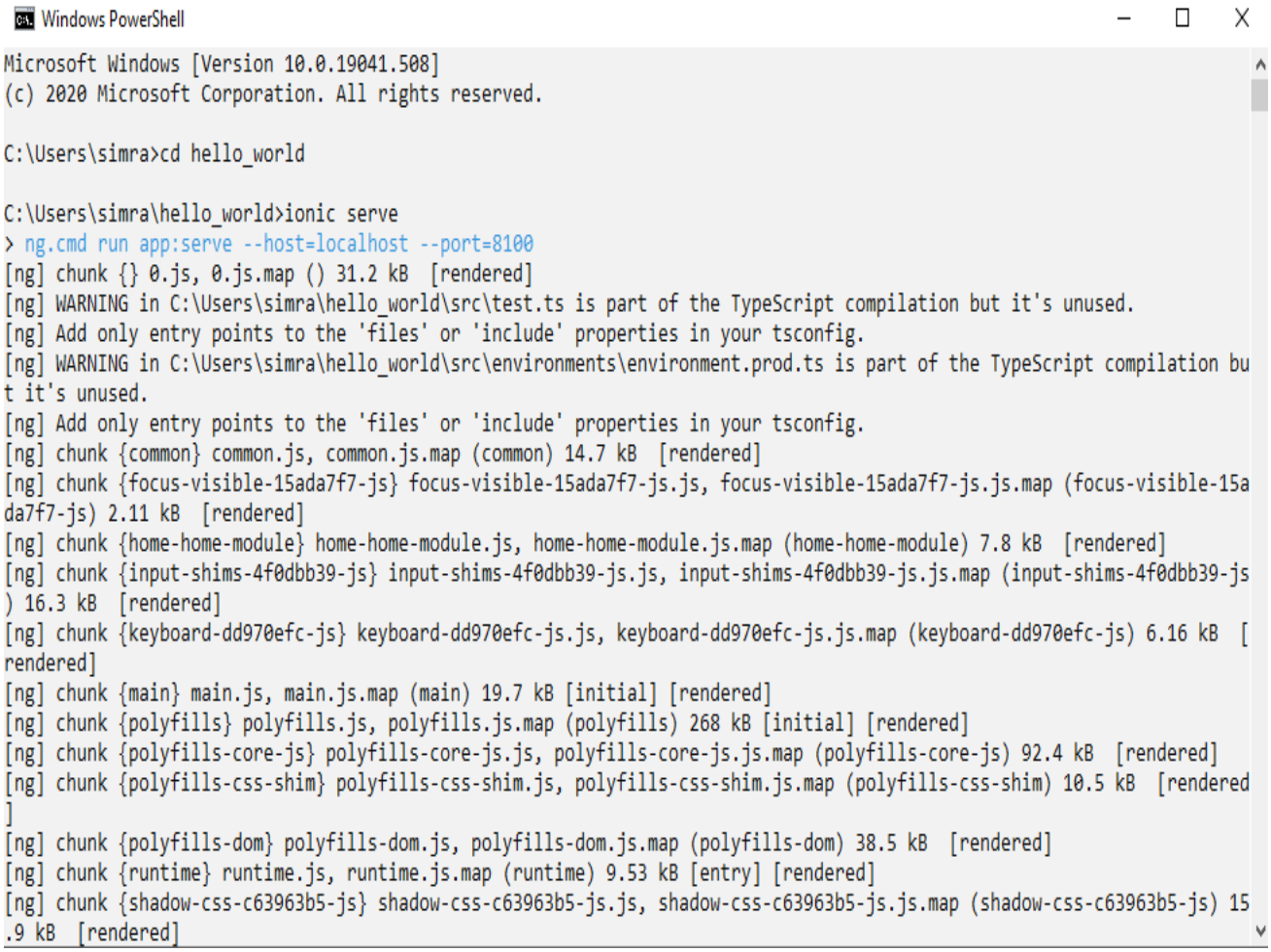
Step 4: cd hello_world

Now open new command prompt and type command cd hello_world. This command will take you to your hello world directory.

Step 5: ionic serve

It will give you the link of your localhost and you will get the following output.



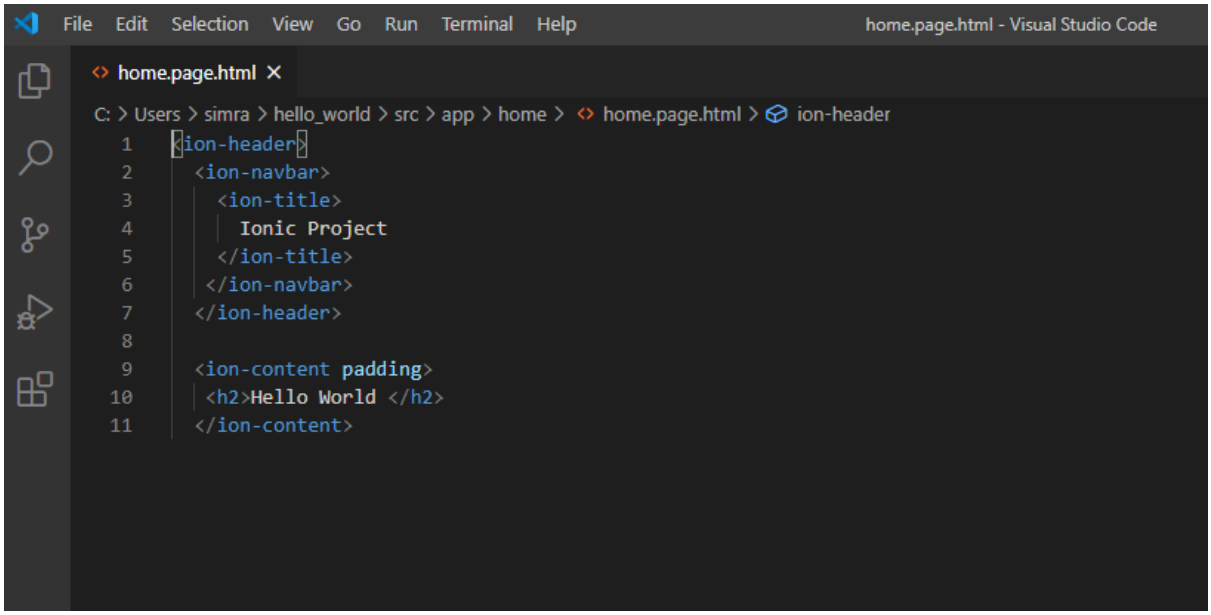


```
Windows PowerShell
[ng] chunk {polyfills-css-shim} polyfills-css-shim.js, polyfills-css-shim.js.map (polyfills-css-shim) 10.5 kB  [rendered
]
[ng] chunk {polyfills-dom} polyfills-dom.js, polyfills-dom.js.map (polyfills-dom) 38.5 kB  [rendered]
[ng] chunk {runtime} runtime.js, runtime.js.map (runtime) 9.53 kB [entry] [rendered]
[ng] chunk {shadow-css-c63963b5-js} shadow-css-c63963b5-js.js, shadow-css-c63963b5-js.js.map (shadow-css-c63963b5-js) 15
.9 kB  [rendered]
[ng] chunk {status-tap-0b3e89c4-js} status-tap-0b3e89c4-js.js, status-tap-0b3e89c4-js.js.map (status-tap-0b3e89c4-js) 1.
6 kB  [rendered]
[ng] chunk {styles} styles.js, styles.js.map (styles) 93 kB [initial] [rendered]
[ng] chunk {swipe-back-0a6a44c8-js} swipe-back-0a6a44c8-js.js, swipe-back-0a6a44c8-js.js.map (swipe-back-0a6a44c8-js) 3.
05 kB  [rendered]
[ng] chunk {swiper-bundle-95afeea2-js} swiper-bundle-95afeea2-js.js, swiper-bundle-95afeea2-js.js.map (swiper-bundle-95a
feea2-js) 200 kB  [rendered]
[ng] chunk {tap-click-252af35a-js} tap-click-252af35a-js.js, tap-click-252af35a-js.js.map (tap-click-252af35a-js) 6.22 k
B  [rendered]
[ng] chunk {vendor} vendor.js, vendor.js.map (vendor) 4.79 MB [initial] [rendered]
[ng] Date: 2020-09-15T06:30:04.820Z - Hash: 1913f2b5807aef1999b0 - Time: 97564ms
[INFO] ... and 42 additional chunks
[ng] : Compiled successfully.

[INFO]

      Local: http://localhost:8100

      Use Ctrl+C to quit this process

[INFO] Browser window opened to http://localhost:8100!
```

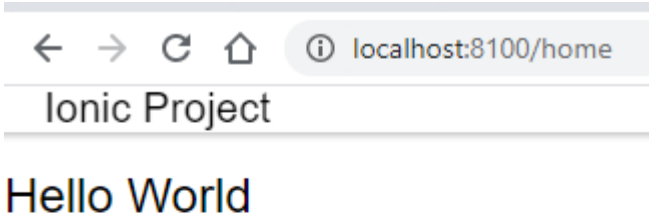
Now, open the text editor and open the project hello_world.

Write the code as shown below:

Homepage.html



OUTPUT:



Build android:

Pre-requisites:

1. Java jdk installation and JAVA_HOME
2. ANDROID_HOME
3. Android SDK build Tools
4. Android SDK platforms Tools.

Type the following commands in the prompt to build android apk

Step 6: ionic cordova build android

Step7: 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.

Practical - 06

Aim: Ionic 2/Ionic 4-Adding Cordova Android Platform

THEORY:

Ionic is open source framework used for developing mobile applications. It provides tools and services for building Mobile UI with native look and feel. Ionic framework needs native wrapper to be able to run on mobile devices. This is an introductory tutorial, which covers the basics of the Ionic Open Source Framework and explains how to deal with its various components and sub-components.

Steps to install Ionic and build your first project.

Step1: Go to command prompt and check if npm is installed or not by using the following command:

npm – v

This shows the version of npm, if it is not installed then install npm.

You can also check the node version by typing the command

node

Step 2: npm install -g ionic cordova.

Step 3: ionic start hello_world blank

Here, we have chosen blank as a template.

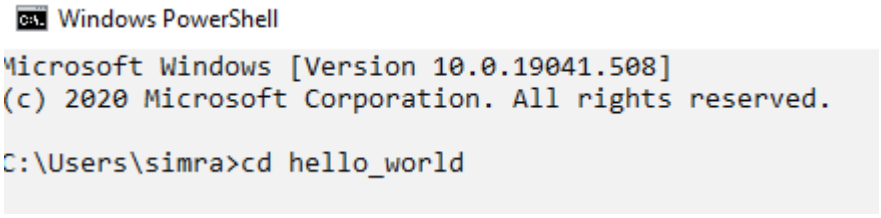
This command will create a folder name hello_world.

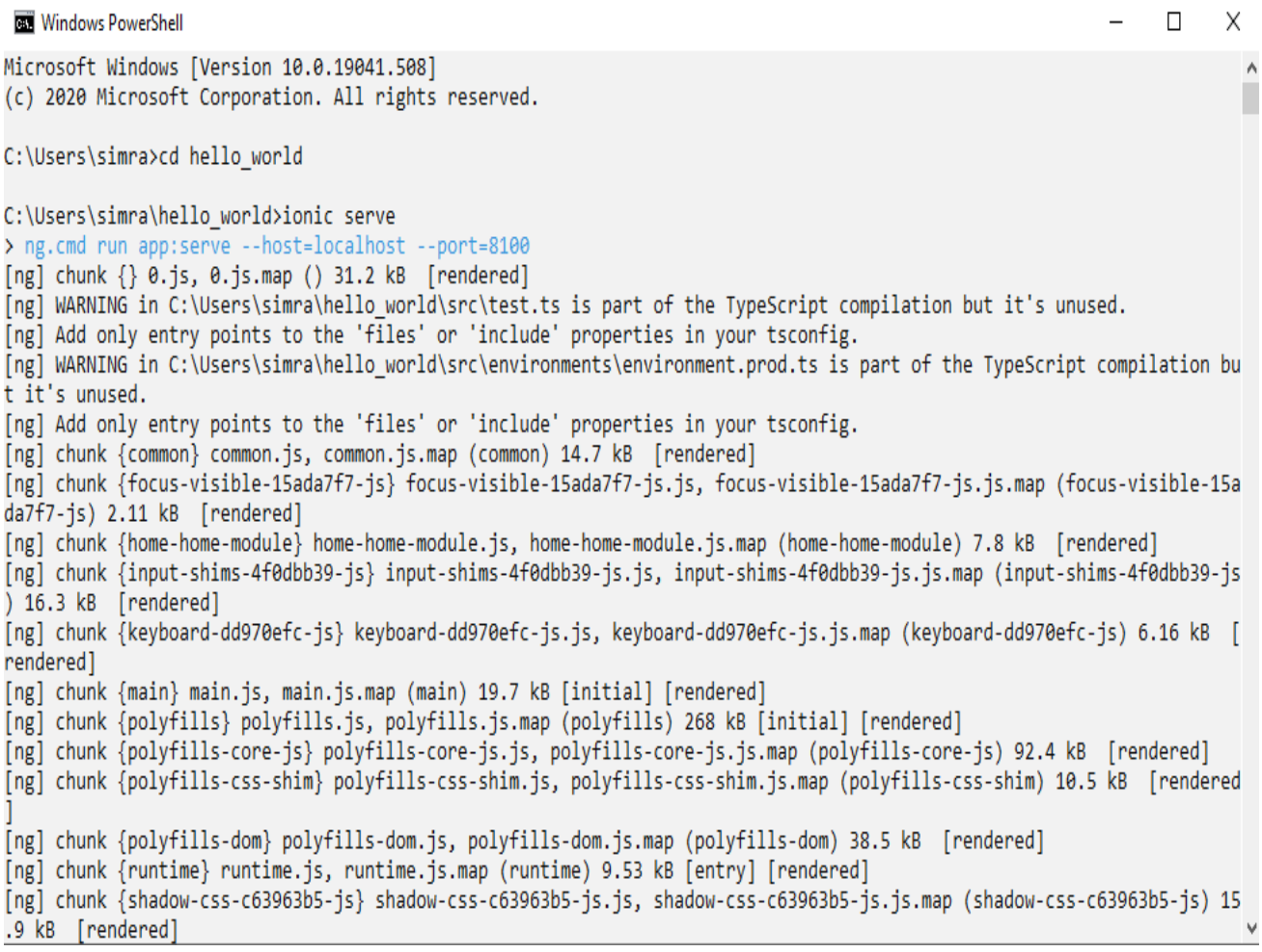
Step 4: cd hello_world

Now open new command prompt and type command cd hello_world. This command will take you to your hello world directory.

Step 5: ionic serve

It will give you the link of your localhost and you will get the following output.

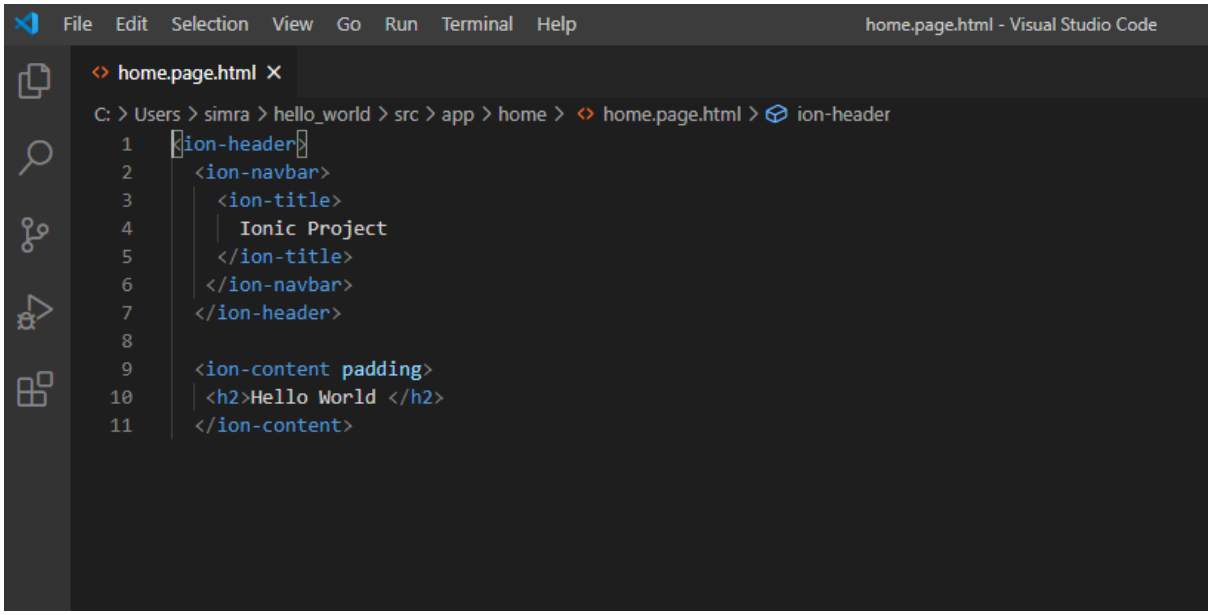




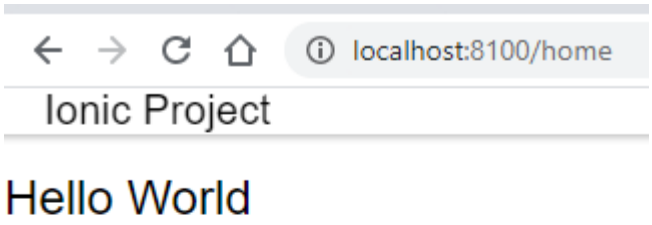
```
Windows PowerShell
[ng] chunk {polyfills-css-shim} polyfills-css-shim.js, polyfills-css-shim.js.map (polyfills-css-shim) 10.5 kB  [rendered
]
[ng] chunk {polyfills-dom} polyfills-dom.js, polyfills-dom.js.map (polyfills-dom) 38.5 kB  [rendered]
[ng] chunk {runtime} runtime.js, runtime.js.map (runtime) 9.53 kB [entry] [rendered]
[ng] chunk {shadow-css-c63963b5-js} shadow-css-c63963b5-js.js, shadow-css-c63963b5-js.js.map (shadow-css-c63963b5-js) 15
.9 kB  [rendered]
[ng] chunk {status-tap-0b3e89c4-js} status-tap-0b3e89c4-js.js, status-tap-0b3e89c4-js.js.map (status-tap-0b3e89c4-js) 1.
6 kB  [rendered]
[ng] chunk {styles} styles.js, styles.js.map (styles) 93 kB [initial] [rendered]
[ng] chunk {swipe-back-0a6a44c8-js} swipe-back-0a6a44c8-js.js, swipe-back-0a6a44c8-js.js.map (swipe-back-0a6a44c8-js) 3.
05 kB  [rendered]
[ng] chunk {swiper-bundle-95afeea2-js} swiper-bundle-95afeea2-js.js, swiper-bundle-95afeea2-js.js.map (swiper-bundle-95a
feea2-js) 200 kB  [rendered]
[ng] chunk {tap-click-252af35a-js} tap-click-252af35a-js.js, tap-click-252af35a-js.js.map (tap-click-252af35a-js) 6.22 k
B  [rendered]
[ng] chunk {vendor} vendor.js, vendor.js.map (vendor) 4.79 MB [initial] [rendered]
[ng] Date: 2020-09-15T06:30:04.820Z - Hash: 1913f2b5807aef1999b0 - Time: 97564ms
[INFO] ... and 42 additional chunks
[ng] : Compiled successfully.

[INFO]

      Local: http://localhost:8100

      Use Ctrl+C to quit this process

[INFO] Browser window opened to http://localhost:8100!
```

Now, open the text editor and open the project hello_world.

Write the code as shown below:



OUTPUT:



Build android:

Pre-requisites:

1. Java jdk installation and JAVA_HOME
2. ANDROID_HOME
3. Android SDK build Tools
4. Android SDK platforms Tools.

Type the following commands in the prompt to build android apk.

Step 6: ionic cordova build android

Step7: 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.

Practical - 07

Aim: Ionic 2/Ionic 4-Create, Generate and Add Pages

Theory:

Steps to create, generate and add pages

Step 1: mkdir angular_ionic

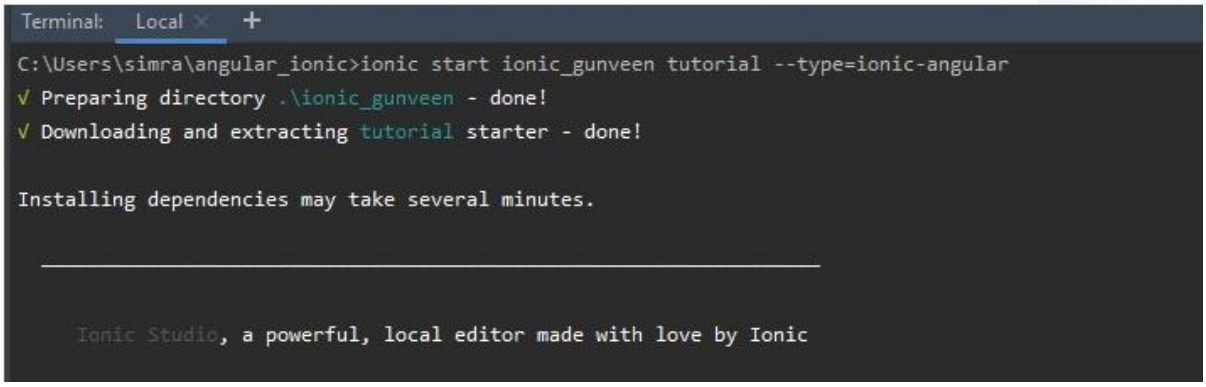
This command creates a directory named angular_ionic

Step 2: cd angular_ionic

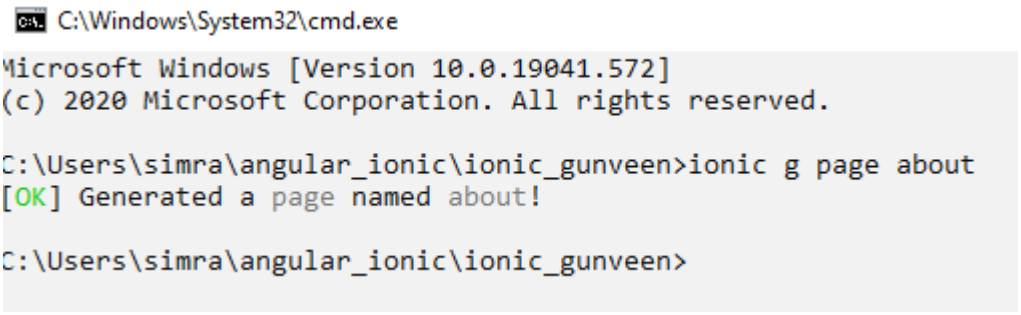
This command will take you to your angular_ionic directory.

Step 3: ionic start ionic_gunveen tutorial --type=ionic-angular

This command will create a new directory ionic_gunveen



Step 4: ionic g page about



This command will create a folder with name about and three files:

- about.html: contains UI components for the page.
- about.scss: contains sass styles to style your page.
- about.ts: contains TypeScript code.

CODE:

ADDING PAGES TO app.module.ts

ionic_gunveen > src > app > TS app.module.ts > AppModule

```
1   import { BrowserModule } from '@angular/platform-browser';
2   import { NgModule, ErrorHandler } from '@angular/core';
3   import { IonicApp, IonicModule, IonicErrorHandler } from 'ionic-angular';
4   import { MyApp } from './app.component';
5
6   import { HelloIonicPage } from '../pages/hello-ionic/hello-ionic';
7   import { ItemDetailsPage } from '../pages/item-details/item-details';
8   import { ListPage } from '../pages/list/list';
9   import {AboutPage} from '../pages/about/about';
10
11  import { StatusBar } from '@ionic-native/status-bar';
12  import { SplashScreen } from '@ionic-native/splash-screen';
13
14  @NgModule({
15    declarations: [
16      MyApp,
17      HelloIonicPage,
18      ItemDetailsPage,
19      ListPage,
20      AboutPage
21    ],
22    imports: [
23      BrowserModule,
24      IonicModule.forRoot(MyApp),
25    ],
```

```
  ],
  bootstrap: [IonicApp],
  entryComponents: [
    MyApp,
    HelloIonicPage,
    ItemDetailsPage,
    ListPage,
    AboutPage
  ],
  providers: [
    StatusBar,
    SplashScreen,
    {provide: ErrorHandler, useClass: IonicErrorHandler}
  ]
})
export class AppModule {}
```

about.html

```
ionic_gunveen > src > pages > about > about.html > ion-header > ion-navbar
<!--
  Generated template for the AboutPage page.

  See http://ionicframework.com/docs/components/#navigation for more info on
  Ionic pages and navigation.
-->
<ion-header>
  <ion-navbar>
    <button ion-button menuToggle>
      <ion-icon name="menu"></ion-icon>
    </button>
    <ion-title>about</ion-title>
  </ion-navbar>
</ion-header>

<ion-content padding>
  <h2>About Page</h2>
  <p>This is a page inside</p>
</ion-content>
```

About.module.ts

ionic_gunveen > src > pages > about > TS about.module.ts > ...

```ts
import { NgModule } from '@angular/core';
import { IonicPageModule } from 'ionic-angular';
import { AboutPage } from './about';

@NgModule({
  declarations: [
    AboutPage,
  ],
  imports: [
    IonicPageModule.forChild(AboutPage),
  ],
})
export class AboutPageModule {}

```

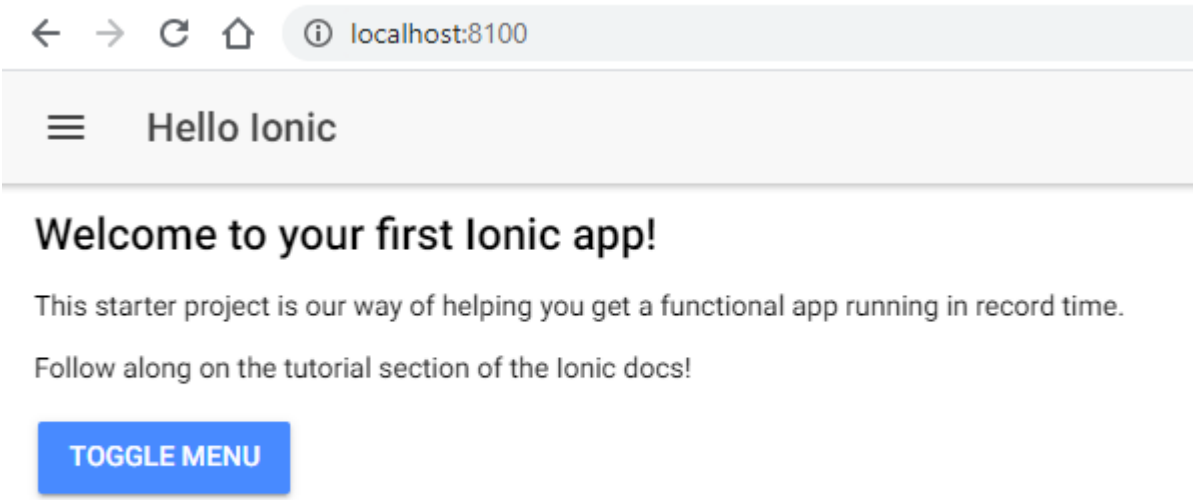
about.ts

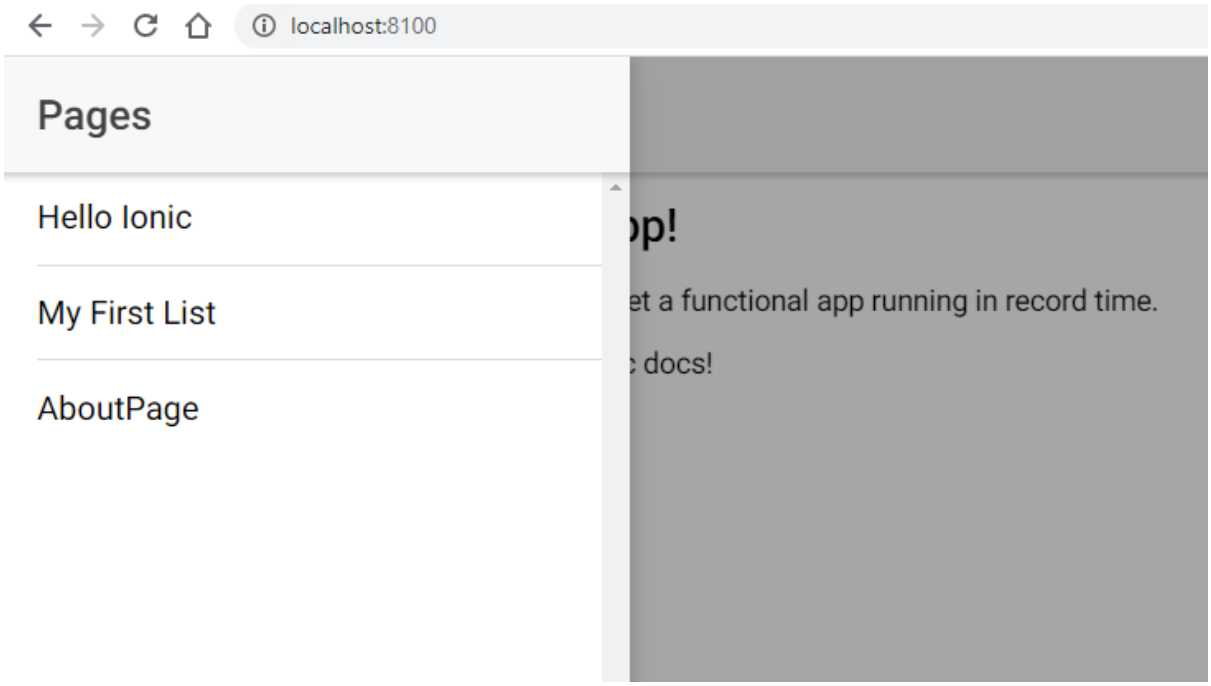
ionic_gunveen > src > pages > about > TS about.ts > ...

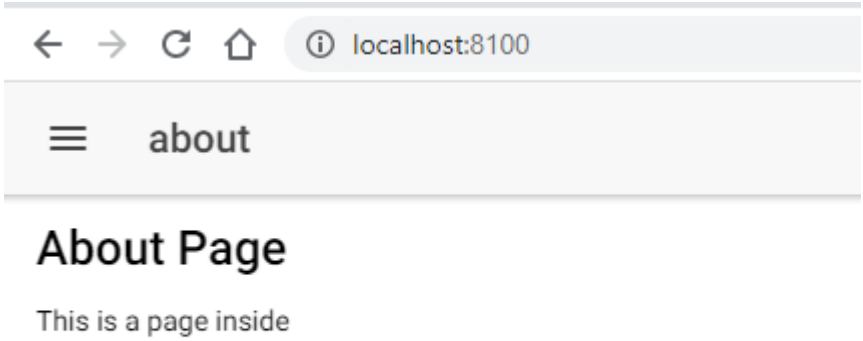
```ts
import { Component } from '@angular/core';
import { IonicPage, NavController, NavParams } from 'ionic-angular';

/**
 * Generated class for the AboutPage page.
 *
 * See https://ionicframework.com/docs/components/#navigation for more info on
 * Ionic pages and navigation.
 */

@IonicPage()
@Component({
  selector: 'page-about',
  templateUrl: 'about.html',
})
export class AboutPage {

  constructor(public navCtrl: NavController, public navParams: NavParams) {
  }

  ionViewDidLoad() {
    console.log('ionViewDidLoad AboutPage');
  }

}
```

OUTPUT:

```
C:\Users\simra\angular_ionic\ionic_gunveen>ionic serve
> ionic-app-scripts.cmd serve --address localhost --port 8100 --livereload-port 35729 --dev-logger-port
 53703 --nobrowser
[app-scripts] [19:36:14]  ionic-app-scripts 3.2.4
[app-scripts] [19:36:14]  watch started ...
[app-scripts] [19:36:14]  build dev started ...
[app-scripts] [19:36:15]  clean started ...
[app-scripts] [19:36:15]  clean finished in 112 ms
[app-scripts] [19:36:15]  copy started ...
[app-scripts] [19:36:16]  deeplinks started ...
[app-scripts] [19:36:16]  deeplinks finished in 61 ms
[app-scripts] [19:36:16]  transpile started ...
[app-scripts] [19:36:52]  transpile finished in 36.24 s
[app-scripts] [19:36:52]  preprocess started ...
[app-scripts] [19:36:52]  preprocess finished in 1 ms
[app-scripts] [19:36:52]  webpack started ...
[app-scripts] [19:37:25]  copy finished in 70.03 s
[app-scripts] [19:37:37]  webpack finished in 45.11 s
[app-scripts] [19:37:37]  sass started ...
[app-scripts] [19:38:11]  sass finished in 34.05 s
[app-scripts] [19:38:11]  postprocess started ...
[app-scripts] [19:38:13]  postprocess finished in 1.39 s
[app-scripts] [19:38:13]  lint started ...
[app-scripts] [19:38:28]  build dev finished in 133.75 s
[app-scripts] [19:38:29]  watch ready in 135.17 s

[INFO]

      Local: http://localhost:8100

      Use Ctrl+C to quit this process

[INFO] Browser window opened to http://localhost:8100!
```




localhost:8100
Pages
Hello Ionic
My First List
AboutPage
et a functional app running in record time.
docs!


localhost:8100
about
About Page
This is a page inside

Practical - 08

Aim: Ionic CLI v3-start command templates 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.

ionic start ionic_proj1 tabs

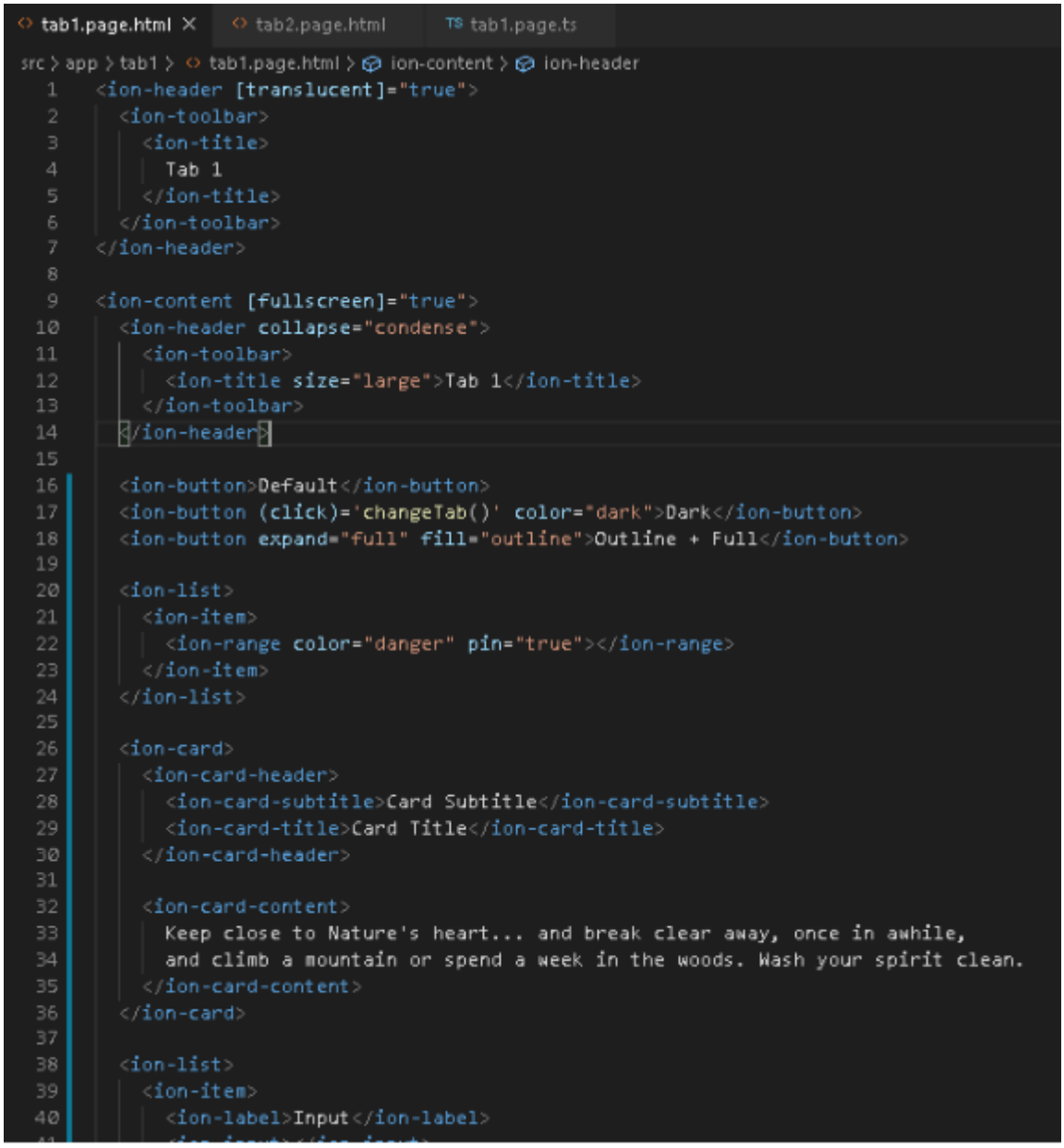
This command creates a working Ionic app. It installs dependencies for you and sets up your project.

Running ionic start without any arguments will prompt you for information about your new project.

The first argument is your app's name. Don't worry--you can always change this later. The --project-id is generated from name unless explicitly specified.

The second argument is the template from which to generate your app. You can list all templates with the --list option

CODE (Tab 1):



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

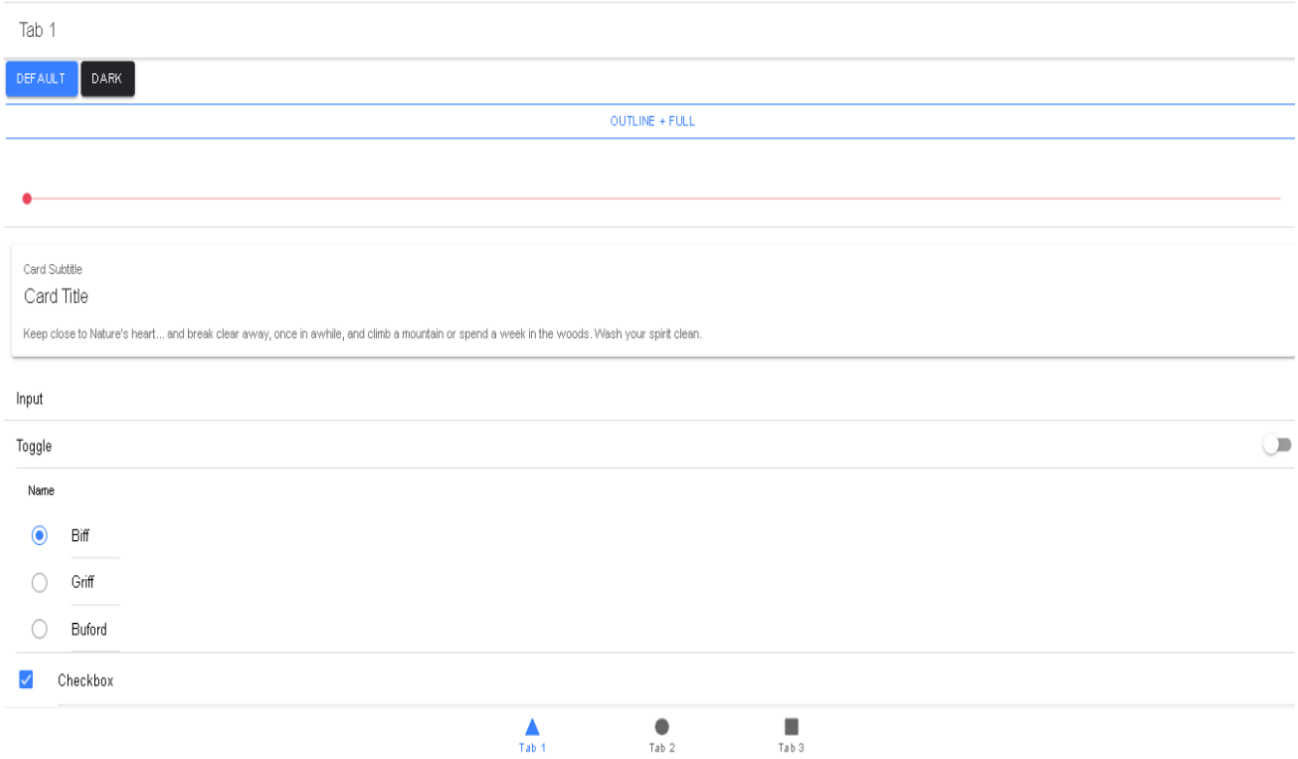
src > app > tab1 > tab1.page.html > ion-content > ion-header

```
<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>
```

CODE (Tab 2):

```
  <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>
```

OUTPUT:


Tab 1
DEFAULT
DARK
OUTLINE + FULL
Card Subtitle
Card Title
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.
Input
Toggle
Name
Biff
Griff
Buford
Checkbox
Tab 1
Tab 2
Tab 3