



Ionic CLI Basics

Project Management

<code>ionic start</code>	Create a new Ionic project.
	Example: <code>ionic start myApp blank --type=angular</code>
<code>ionic serve</code>	Run the app in the browser for development.
	Example: <code>ionic serve --lab</code> (with Ionic Lab)
<code>ionic build</code>	Build the app for a specific platform (e.g., web).
	Example: <code>ionic build --prod</code> (production build)
<code>ionic cordova platform add <platform></code>	Add a platform (e.g., android, ios) to the project.
	Example: <code>ionic cordova platform add android</code>
<code>ionic cordova run <platform></code>	Run the app on a connected device or emulator.
	Example: <code>ionic cordova run android</code>
<code>ionic generate <type> <name></code>	Generate components, pages, services, etc.
	Example: <code>ionic generate component my-component</code>

Configuration

The <code>ionic.config.json</code> file stores project configuration.
Key settings include <code>name</code> , <code>integrations</code> , and <code>type</code> .
Environment variables can be set using <code>.env</code> files and accessed via <code>process.env</code> .
Use <code>ionic config get <property></code> and <code>ionic config set <property> <value></code> to manage configuration.

Core Components

Basic UI Elements

<code><ion-header></code>	The header of a page. Contains <code><ion-toolbar></code> , <code><ion-title></code> , and <code><ion-buttons></code> .
<code><ion-content></code>	The main content area of a page. Used for displaying text, images, and other components.
<code><ion-footer></code>	The footer of a page. Used for displaying navigation or additional information.
<code><ion-button></code>	A clickable button element. Supports various styles and sizes.
<code><ion-input></code>	A text input field. Supports different types of input (text, number, email, etc.).
<code><ion-list></code>	A container for list items. Used with <code><ion-item></code> and related components.

Navigation

<code><ion-nav></code>	A container for managing navigation history.
<code><ion-router-outlet></code>	Displays the current page based on the router configuration.
<code>routerLink</code>	Angular directive for navigating to a specific route.
	Example: <code><ion-button routerLink="/home">Home</ion-button></code>
<code>NavController</code>	Service for programmatic navigation.
	Example: <code>this.navCtrl.navigateForward('/about');</code>

Ionic Native & Plugins

Working with Native Features

Ionic Native provides a wrapper around Cordova plugins, making them easier to use in Angular projects.

Install plugins via Cordova or Capacitor CLI:

```
ionic cordova plugin add cordova-plugin-camera  
npm install @ionic-native/camera
```

Import and use plugins in your Angular components:

```
import { Camera, CameraOptions } from '@ionic-native/camera/ngx';
```

```
constructor(private camera: Camera) { }
```

```
...
```

```
const options: CameraOptions = { ... };  
this.camera.getPicture(options).then((imageData) => { ... });
```

Common Plugins

`@ionic-native/camera`

Access the device camera for taking pictures and videos.

`@ionic-native/geolocation`

Access the device's GPS for location information.

`@ionic-native/barcode-scanner`

Scan barcodes and QR codes.

`@ionic-native/splash-screen`

Manage the app's splash screen.

`@ionic-native/status-bar`

Control the device's status bar appearance.

`@ionic-native/file`

Access the device's file system.

Theming and Styling

CSS Variables (Custom Properties)

Ionic uses CSS variables for theming. You can override these variables to customize the look and feel of your app.

Example:

```
:root {  
  --ion-color-primary: #3880ff;  
  --ion-color-primary-rgb: 56, 128, 255;  
}
```

Theme variables are defined in the `variables.scss` file.

Modes

Ionic supports different modes (MD for Material Design, iOS for iOS style).

The mode is automatically set based on the platform.

You can force a specific mode:

```
<ion-app  
mode="md">  
  ...  
</ion-app>
```

Global Styles

Global styles can be defined in the `global.scss` file.

Use CSS classes and selectors to style your components.

Example:

```
.my-custom-class {  
  color: red;  
}
```

```
<ion-button class="my-custom-class">Click Me</ion-button>
```