**Platform Classes**

Ionic automagically adds classes to the <body> of your project based on the device you are using to view the project. This means if you are viewing your app on an iOS device, the <body> will have the platform-ios class applied. Some information about the different classes can be found below.

**Platform Device Classes**

[ionic.Platform](http://ionicframework.com/docs/v1/api/utility/ionic.Platform/) is used to retrieve the device information and apply classes to the <body> of the project. These classes are used by the Ionic SDK to give the project a platform specific look and feel, and can be used by you to override styles.

|  |  |  |
| --- | --- | --- |
| Platform | Class | Description |
| Browser | platform-browser | Whether the application is running on the Desktop Browser - applied if you are running ionic serve |
| Cordova | platform-cordova | Whether the application is running on the device, as the device uses Cordova to display the application |
| Webview | platform-webview | Whether the application is running within a webview on the device within a native application |
| iOS | platform-ios | The device is iOS; therefore, the "look and feel" will be given the iOS treatment |
| iPad | platform-ipad | iOS device is an iPad - this class is applied, in addition to platform-ios |
| Android | platform-android | The device is Android; therefore, the "look and feel" will be given the Android treatment |
| Windows Phone | platform-windowsphone | The device is Windows Phone; therefore, the "look and feel" will be given the Windows Phone treatment |

**Platform OS Version Classes**

Ionic also adds classes based on the OS version. This can be used to customize your project for a specific OS version. A class is added for the major version and the major + minor versions.

The class is created by looking at the User Agent of the OS version. The major + minor class is created by replacing the . separating the major and minor version numbers with a \_. So, if you are running your project on a device running iOS 8.4, it will add the platform-ios8and platform-ios8\_4 classes. Below are some example classes, but you are not limited to these.

|  |  |  |
| --- | --- | --- |
| Platform OS Version | Class | Description |
| iOS 8 | platform-ios8 | The OS on the device is running iOS 8 |
| iOS 8.4 | platform-ios8\_4 | The OS on the device is running iOS 8.4. This class is applied in addition to platform-ios8 |
| Android 4 | platform-android4 | The OS on the device is running Android 4 |
| Android 4.4 | platform-android4\_4 | The OS on the device is running Android 4.4. This class is applied, in addition to platform-android4 |

**Using Platform Classes to Override Styling**

You can use any of the above classes to override Ionic styling. For example, if you wanted to override the header title on Android to be uppercase, you could use the following code:

.platform-android .bar-header {

text-transform: uppercase;

}

This would uppercase the title for Android, but any other device would remain lowercase.

You could even go as far as customizing the side menu style based on the platform. The possibilities are endless!