

Bootcamp Java Developer

Fase 2 - Java Web Developer
Módulo 15



Introducción a Frameworks

Frameworks y conceptos

Un framework nos permite hacer nuestro **CSS mucho más sencillo**. De hecho, estos sistemas nos permiten trabajar con **grillas, componentes pre armados, hasta elementos que incorporan de manera sencilla JS**.

Eso no significa que no tenemos que trabajar en absoluto, sino por el contrario, la idea es adaptar este framework a nuestras necesidades, muchas veces sobrescribiendo lo necesario para adaptarlo a nuestro *Look and Feel*, sino todo se vería igual.

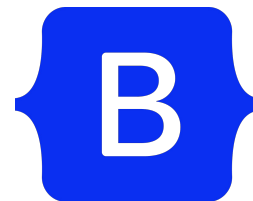
Frameworks muy populares:

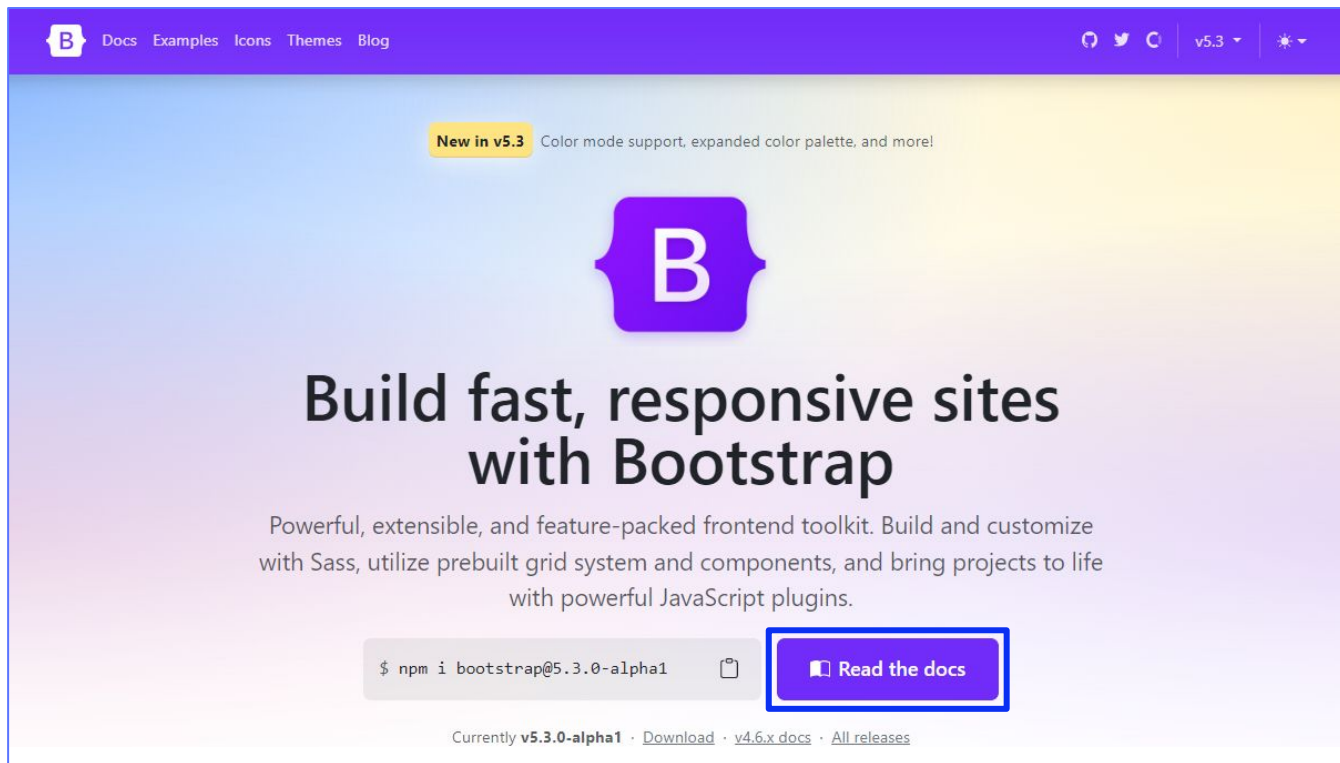
- [Bootstrap](#).
- [Foundation](#).
- [Materialize](#).
- [Bulma](#).

Bootstrap

Introducción

1. Para empezar a trabajar, lo primero que debemos hacer es ir a la página oficial:
<http://getbootstrap.com/>.
2. Nos encontraremos con la imagen que se muestra en la siguiente pantalla; donde haremos clic en el botón violeta, llamado ***Read the docs***.






The screenshot shows the Bootstrap website with a purple header. The navigation bar includes links for Docs, Examples, Icons, Themes, and Blog, along with social media icons and version information (v5.3). A yellow badge highlights 'New in v5.3' with the text 'Color mode support, expanded color palette, and more!'. The main content area features the Bootstrap logo, a large heading 'Build fast, responsive sites with Bootstrap', and a descriptive paragraph. At the bottom, there is a terminal snippet showing the command to install Bootstrap, a 'Read the docs' button, and links for the current version (v5.3.0-alpha1), previous versions (v4.6.x docs), and all releases.

Docs Examples Icons Themes Blog

New in v5.3 Color mode support, expanded color palette, and more!



Build fast, responsive sites with Bootstrap

Powerful, extensible, and feature-packed frontend toolkit. Build and customize with Sass, utilize prebuilt grid system and components, and bring projects to life with powerful JavaScript plugins.

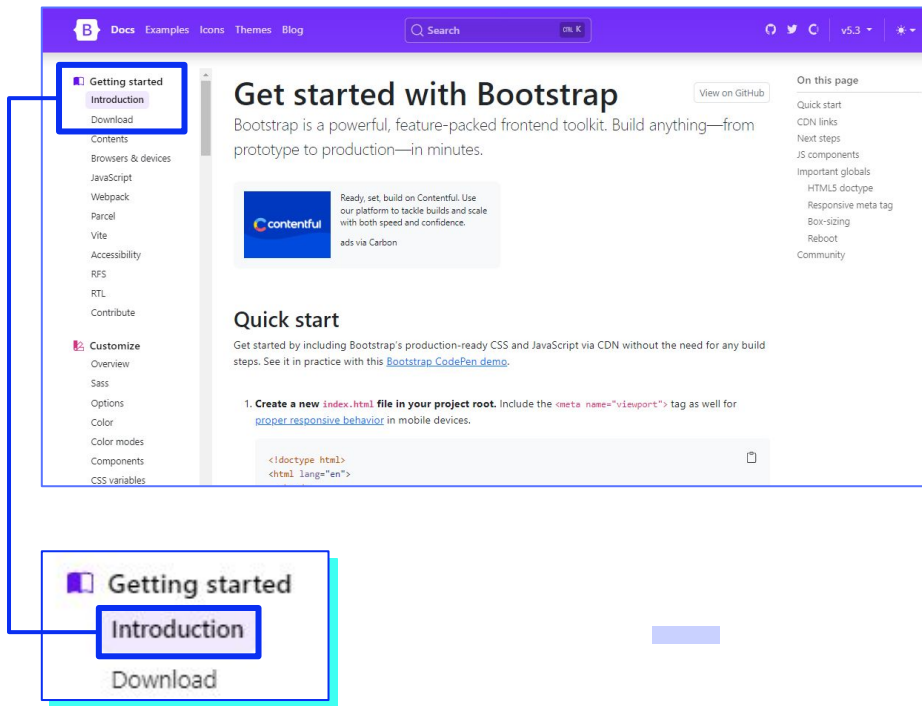
```
$ npm i bootstrap@5.3.0-alpha1
```

[Read the docs](#)

Currently **v5.3.0-alpha1** · [Download](#) · [v4.6.x docs](#) · [All releases](#)

3. La acción anterior nos llevará a la pantalla ***Get started with Bootstrap*** (Comienza con *Bootstrap*), que se muestra en la imagen de la derecha.

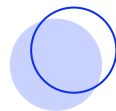
Allí, podemos usar el menú del lado izquierdo para navegar en la misma. O también, en esta primera instancia, sugerimos acceder a la sección denominada ***Introduction*** (Introducción).



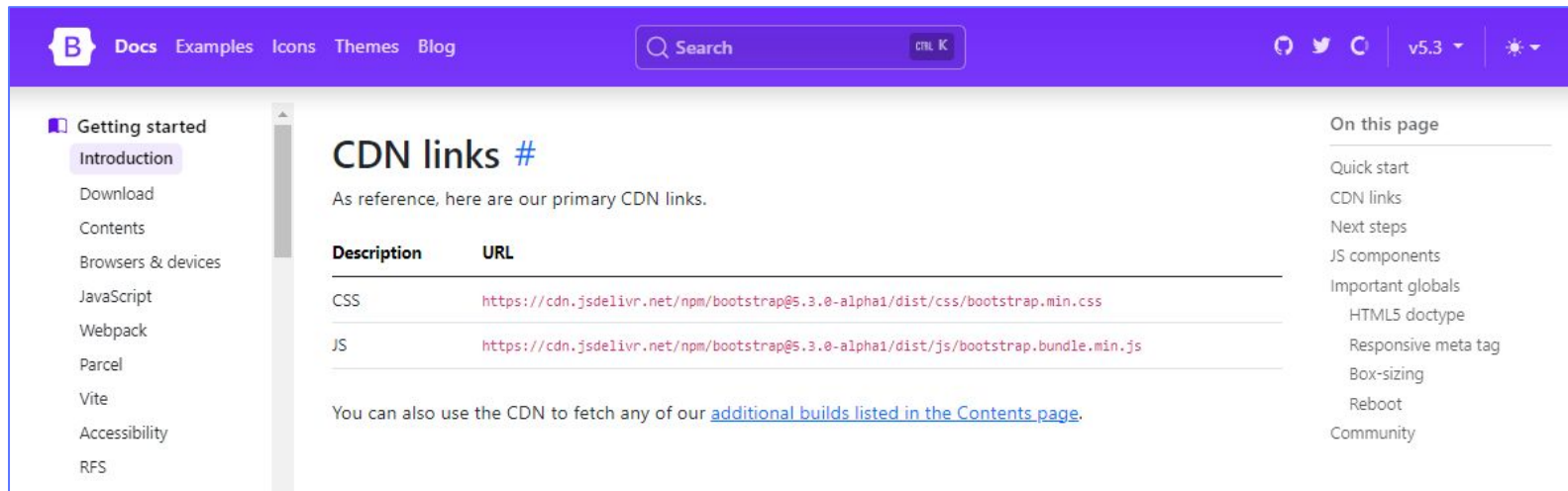
Comenzar a trabajar

4. El siguiente paso es **asegurarnos que en nuestro archivo .html que se vincularía con Bootstrap, debe contener el meta viewport**. La ausencia de este meta, como en un proceso normal de maquetación no permitirá la detección de la detección de los diferentes tamaños de pantalla o *breakpoints*.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```



5. Ahora debemos tomar de la página oficial de Bootstrap la [vinculación o CDN](#) correspondiente:



The screenshot shows the Bootstrap 5.3.0-alpha1 documentation page for CDN links. The page has a purple header with navigation links: Docs, Examples, Icons, Themes, and Blog. A search bar and a 'CTRL K' shortcut are also present. On the left, a sidebar lists 'Getting started' options: Introduction (selected), Download, Contents, Browsers & devices, JavaScript, Webpack, Parcel, Vite, Accessibility, and RFS. The main content area is titled 'CDN links #' and includes a sub-header 'As reference, here are our primary CDN links.' Below this is a table with two columns: 'Description' and 'URL'. The table lists two entries: 'CSS' with the URL 'https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css' and 'JS' with the URL 'https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js'. Below the table, a note states: 'You can also use the CDN to fetch any of our [additional builds listed in the Contents page](#).' On the right side, a 'On this page' section lists links to various parts of the documentation: Quick start, CDN links, Next steps, JS components, Important globals, HTML5 doctype, Responsive meta tag, Box-sizing, Reboot, and Community.

Description	URL
CSS	https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css
JS	https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js

6. En nuestro *archivo.html* el resultado será como el de la imagen debajo:

```
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Introducción a Bootstrap</title>
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css">
  <link rel="stylesheet" href="css/estilos.css">
</head>
```



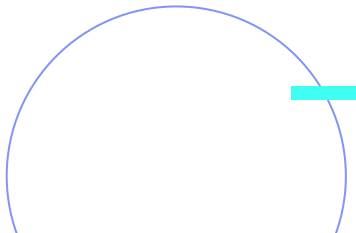
Nota: Es importante entender que siempre nuestra hoja de estilos debe ir debajo, dado que seguramente **nuestra intención es sobrescribir reglas de Bootstrap** y, para tal fin, debemos recordar que siempre el orden de los estilos es importante al momento de trabajar. Por esa razón **siempre nuestra propia hoja debe ir en último lugar.**

Contenedores

Bootstrap posee un uso masivo de clases que nos permiten implementar estilos específicos. Pero lo más importante al momento de comenzar a trabajar es el uso de contenedores. Los **contenedores setean el marco adecuado de trabajo**. Existen diversos tipos de contenedores según los tamaños de pantalla.

Tipos de contenedores más importantes:

- **.container**: esta clase indica el trabajo con un contenedor **centrado**.
- **.container-fluid**: un contenedor fluido es aquel con un width al 100%.



Veamos un ejemplo desde el editor:

```
<div class="container-fluid"> contenedor fluído </div>  
<div class="container"> contendor centrado </div>
```

Luego, desde nuestra propia hoja de estilos vamos a agregar color:

```
.container-fluid {background-color: ■ red;}  
.container { background-color: ■ green;}
```

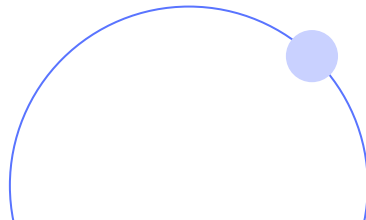


El **resultado** del paso anterior similar a esta imagen:



De esta forma, **podemos elegir diversos contenedores según nuestro diseño.**

Se pueden **reutilizar varios tipos de contenedores en la misma interfaz**, y su uso es fundamental como punto de partida para el resto de las clases que vamos a utilizar desde Bootstrap.

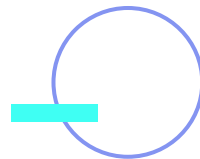


Breakpoints

Si bien al momento de recorrer la página de Bootstrap este concepto no es mostrado en los primeros documentos, consideramos necesario invertir el proceso para poder explicar de mejor manera las clases asociadas a contenedores.

Bootstrap trabaja con un **sistema de *breakpoints* asociados al trabajo responsivo**. Estos breakpoints son enmarcados dentro de un **proceso general de maquetación adaptable**. Por lo tanto, no son privativos u originarios de Bootstrap, sino que el framework toma éstos

para **utilizar conceptos generales de maquetación responsiva en el trabajo general**, que se puede realizar desde el mismo. El sistema toma ciertos rangos de viewport y estos son la base de todo el proceso general de maquetación realizable desde Bootstrap.



Clase container

En el caso de los contenedores, a la clase ya vista llamada **.container**, que si bien ya nos brinda herramientas responsivas, se le agrega la posibilidad de **indicar hasta qué breakpoint estos contenedores estarán centrados o pasarán a ser fluidos**.

```
.container-{breakpoint}, width: 100%
```

Breakpoint	Class infix	Dimensions
X-Small	<i>None</i>	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px



De esta manera, el proceso sería el siguiente:

```
<div class="container-sm">100% hasta small breakpoint</div>
<div class="container-md">100% hasta medium breakpoint</div>
<div class="container-lg">100% hasta large breakpoint</div>
<div class="container-xl">100% hasta extra large breakpoint</div>
<div class="container-xxl">100% hasta extra extra large breakpoint</div>
```

Nota: Bootstrap trabaja con **Sass**. Este preprocesador brinda los valores iniciales de las variables para poder determinar a qué breakpoint corresponde cada medida, por ejemplo, lo que se ve en el código de la derecha:

```
$grid-breakpoints: (
  xs: 0,
  sm: 576px,
  md: 768px,
  lg: 992px,
  xl: 1200px,
  xxl: 1400px
);
```


**¡Sigamos
trabajando!**

