B	
Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web.	

Bootstrap Resources from discoveryvip

Download bootstrap at http://getbootstrap.com/

http://getbootstrap.com/css/

Grid options

	Extra small devices Phones (<768px)	**Small devices Tablets (≥768px)**	**Medium devices Desktops (≥992px)**	**Large devices Desktops (≥1200px)**
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
Container width	None (auto)	750px	970px	1170px
Class prefix	`.col-xs-`	`.col-sm-`	`.col-md-`	`.col-lg-`
# of columns	12			
Column width	Auto	~62px	~81px	~97px
Gutter width	30px (15px on each side of a column)			
Nestable	Yes			
Offsets	Yes			
Column ordering	Yes			

Media queries

We use the following media queries in our Less files to create the key breakpoints in our grid system.

Copy

```
/* Extra small devices (phones, less than 768px) */ /* No media query since this is
the default in Bootstrap */ /* Small devices (tablets, 768px and up) */ @media (min-
width: @screen-sm-min) { ... } /* Medium devices (desktops, 992px and up) */ @media
(min-width: @screen-md-min) { ... } /* Large devices (large desktops, 1200px and up)
*/ @media (min-width: @screen-lg-min) { ... }
```

We occasionally expand on these media queries to include a `max-width` to limit CSS to a narrower set of devices.

Copy

```
@media (max-width: @screen-xs-max) { ... } @media (min-width: @screen-sm-min) and
(max-width: @screen-sm-max) { ... } @media (min-width: @screen-md-min) and (max-
width: @screen-md-max) { ... } @media (min-width: @screen-lg-min) { ... }
```

Example: Stacked-to-horizontal

Using a single set of `.col-md-*` grid classes, you can create a basic grid system that starts out stacked on mobile devices and tablet devices (the extra small to small range) before becoming horizontal on desktop (medium) devices. Place grid columns in any `.row`.

Copy

```
<div class="row"> <div class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-
md-1</div> <div class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-md-
1</div> <div class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-md-1</div>
<div class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-md-1</div> <div
class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-md-1</div> <div
class="col-md-1">.col-md-1</div> <div class="col-md-1">.col-md-1</div> </div> <div
class="row"> <div class="col-md-8">.col-md-8</div> <div class="col-md-4">.col-md-
4</div> </div> <div class="row"> <div class="col-md-4">.col-md-4</div> <div
class="col-md-4">.col-md-4</div> <div class="col-md-4">.col-md-4</div> </div> <div
class="row"> <div class="col-md-6">.col-md-6</div> <div class="col-md-6">.col-md-
6</div> </div>
```

Example: Fluid container

Turn any fixed-width grid layout into a full-width layout by changing your outermost `.container` to `.container-fluid`.

Copy

```
<div class="container-fluid"> <div class="row"> ... </div> </div>
```