Introduction

To develop a parallax scrolling website can be a bit scary especially if you do it for the first time, bu as you know the practice makes perfect. Take this parallax tutorial as your first step to developing great looking parallax website.

Please note, that this tutorial is suitable for a more advanced developers and attached files are the **final files**. Simply open `index.html`, `main.css` and `_main.js` and follow the sections below.

1. Include and initiate Skrollr.js

As a first step we need to include Skrollr.js preferably before the closing `body` tag. This plugin will the magic and will animate the element properties on page scroll. Skrollr is a stand-alone parallax scrolling JavaScript library for mobile (Android, iOS, etc.) and desktop. **No jQuery. Just plain JavaScript.**

```
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.
<script>window.jQuery || document.write('<script src="
<script src="js/skrollr.js"></script>
<script src="js/_main.js"></script>
```

Initiate the Skrollr inside of the `_main.js` file. You can log the current scroll position if you need to work out a precise `timing` and `positioning` of your animations.

```
( function( $ ) {
    // Init Skrollr
    var s = skrollr.init({
        render: function(data) {
            //Debugging - Log the current scroll posit
            //console.log(data.curTop);
        }
    });
} )( jQuery );
```

Now lets have a look at the markup and Skrollr settings of the individual slides.

2. Slide #1 – Fade out elements

Section height – 100% of the viewport, resized on page load

```
/* CSS */
.hsContainer {
    display: table;
    table-layout: fixed;
    width: 100%;
    height: 100%;
    overflow: hidden;
    position: relative;
    opacity: 0;
}
.hsContent {
    max-width: 450px;
    margin: -150px auto 0 auto;
    display: table-cell;
    vertical-align: middle;
    color: #ebebeb;
    padding: 0 8%;
    text-align: center
}
.bcg {
    background-position: center center;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: cover;
    height: 100%;
    width: 100%;
}
/* Slide 1 */
#slide-1 .bcg {background-image:url('../img/bcg_slide-
```

```
<!-- HTML -->
<section id="slide-1" class="homeSlide">
    <div class="bcg"
        data-center="background-position: 50% 0px;"
        data-top-bottom="background-position: 50% -100
        data-anchor-target="#slide-1"
    >
        <div class="hsContainer">
            <div class="hsContent"
                data-center="opacity: 1"
                data-106-top="opacity: 0"
                data-anchor-target="#slide-1 h2"
            >
                <h2>Fade out elements before <br>they
            </div>
    </div>
</section>
```

.hsContainer and .hsContent are two nested containers helping us to center the content vertica on the page. .bcg is a container which takes up 100% width and height of each section and contains our background image.

Background animation

The background image of .bcg is animating from the initial position (data-center) 50% 0px to 50% -100px. This means that the background image moves up by 10 between the start of the scrolling and when the bottom of the #slide-1 hits the top of the viewport (data-top-bottom).

Content fading in and out

The content of the slide starts at full opacity as specified in data-center attribute and fades out to opacity: 0 when the #slide-1 h2 hits 106 pixels from the top of the viewport, where 106 is t height of our header.

3. Slide #2 – Background color animation

Section height – 310px fixed height

```
/* CSS - Slide 2 */
#slide-2 .bcg {
    background: none;
    background-color: #010101;
    height: 310px;
    text-align: center
}
```

```
<!-- HTML -->
<section id="slide-2">
    <div class="bcg"
        data-0="background-color:rgb(1,27,59);"
        data-top="background-color:(0,0,0);"
        data-anchor-target="#slide-2"
    >
```

```
            <div class="hsContent">
                <h2
                    data--200-bottom="opacity: 0"
                    data-center="opacity: 1"
                    data-206-top="opacity: 1"
                    data-106-top="opacity: 0"
                    data-anchor-target="#slide-2 h2"
                >
                    Fade me in and out
                </h2>
            </div>
        </div>
    </div>
</section>
```

Background animation

We are simply animating the background color from dark blue to black. `data-0` contains the initial background color and `data-top` contains the background color to which we are animating when the `#slide-2` hits the top of the viewport.

Content fading in and out

The content fades in when `#slide-2 h2` is 206 pixels from the bottom of the viewport and fades o similar way as the `#slide-1` content.

4. Slide #3 – Move background image horizontally

Section height – 100% of the viewport, resized on page load

```
/* CSS - Slide 3 */
#slide-3 .bcg {background-image:url('../img/bcg_slide-
```

```
<!-- HTML -->
<section id="slide-3" class="homeSlide">
    <div class="bcg"
        data-center="background-position: 0px 50%;"
        data-bottom-top="background-position: 0px 40%;
        data-top-bottom="background-position: -40px 50
        data-anchor-target="#slide-3"
    >
        <div class="hsContainer">
            <div class="hsContent">
```

```
                    data-106-top="opacity: 0"
                    data-bottom="opacity: 1; position:
                    data--30p-top="opacity: 1;"
                    data--60p-top="opacity: 0;"
                    data-anchor-target="#slide-3"
                >
                    <h2>Fixed element fading in and ou
                </div>
            </div>
        </div>
    </div>
</section>
```

Background animation

The background image on this slide is only slightly moving up until the slide is centered in the viewport. Then it's moving `40 pixels` left as specified in the ending position `data-top-bottom="background-position: -40px 50%;"`.

Fixed content

The content is this time fixed to `206px` from the top of the slide and doesn't move. It fades in when the `#slide-3` is 106 pixel from the top of the viewport and stays at full opacity until the slide is `30%` above the top of the viewport. Then it fades out when `60%` of the slide is out of the view.

Note: Using the `percentage` instead of pixels is very handy especially if you don't know how tall your sections will be – e.g. if you are using javascript to keep sections 100% height of your viewpo on window resize.

5. Slide #4 – Curtain effect

Section height – 200% of the viewport, resized on page load

```
/* Slide 4 */
.curtainContainer {
    width: 100%; height: 100%;
    position: relative;
}
.curtainContainer .curtain {
    width: 300%; height: 1px;
    background-color: #000000;
    position: absolute; top: 25%; left: 0;
    opacity: 0
```

```
.curtainContainer .copy {
    position: absolute;
    bottom: 30%; left: 0;
    width: 100%; text-align: center;
}
#slide-4 .bcg {background-image:url('../img/bcg_slide-
```

```
<!-- HTML -->
<section id="slide-4" class="homeSlide homeSlideTall">
    <div class="bcg"
        data-center="background-position: 50% 0px;"
        data-bottom-top="background-position: 50% 100p
        data-top-bottom="background-position: 50% -100
        data-anchor-target="#slide-4"
    >
        <div class="curtainContainer">
            <div class="curtain"
                data-bottom-top="opacity: 0"
                data-106-top="height: 1%; top: -10%; o
                data-center="height: 100%; top: 0%; op
                data-anchor-target="#slide-4"
            ></div>
            <div class="copy"
                data-bottom-top="opacity: 0"
                data--100-bottom="opacity: 0"
                data--280-bottom="opacity: 1;"
                data-280-top="opacity: 1;"
                data-106-top="opacity: 0;"
                data-anchor-target="#slide-4 .copy"
            >
                <h2>Curtain effect while you scroll</h
            </div>

        </div>
    </div>
</section>
```

This one is my favourite. Why? Because it was **made in the browser**, by trying a few things.

Background animation

The background animation is similar to our first slide, so lets jump straight the curtain effect. **Note** t this section has additional class `.homeSlideTall` which give this section a double the height.

Curtain animation

The `.curtain` is basically a div with a `background-color` set to `#000`. It's invisible (`data-bottom-top`) until the top of the `#slide-4` reaches `106` pixels from the top of the viewport (`data-106-top`).

It's `height` grows from `1%` and becomes `100%` when this slide is centered in the viewport (`data-center`). Remember that we've doubled the height of this slide? Now you know why. It give us more room to show and fade out the `.copy`.

Copy animation

The `.copy` fades in just after the curtain covers the whole screen and fades out before it reaches header (`data-106-top`).

6. Slide #5 – Cross-fade to blur effect

Section height – 300% of the viewport, resized on page load

```
/* Slide 5 */
#slide-5 {position: relative;}
#slide-5 .bcg {background-image:url('../img/bcg_slide-
#slide-5 .bcg2 {
    background-image:url('../img/bcg_slide-5b.jpg');
    position: fixed; bottom: 0; left: 0;
    opacity: 0; z-index: 1
}
#slide-5 .bcg3 {
    background: none; background-color: #010101;
    position: fixed; bottom: 0; left: 0;
    opacity: 0; z-index: 2
}
```

```
<!-- HTML -->
<section id="slide-5" class="homeSlide homeSlideTall2"
    <div class="bcg">&nbsp;</div>
    <div class="bcg bcg2"
        data-bottom-top="opacity: 0;"
        data--33p-top="opacity: 0;"
        data--66p-top="opacity: 1;"
        data-anchor-target="#slide-5"
    >
        <div class="hsContainer">
            <div class="hsContent"
                data-bottom-top="opacity: 0;"
                data-anchor-target="#slide-5"
            >
                <h2>Fixed element fading in and out</h
            </div>
        </div>
    </div>
    <div class="bcg bcg3"
        data-300-bottom="opacity: 0;"
        data-100-bottom="opacity: 1;"
        data-anchor-target="#slide-5"
    >
        <div class="hsContainer">
            <div class="hsContent"
                data-100-bottom="opacity: 0;"
                data-bottom="opacity: 1;"
                data-anchor-target="#slide-5"
            >
                <h2>The End</h2>
            </div>
        </div>
    </div>
</section>
```

Finally lets have a look at the last section and congratulations if you've made it this far.

`.bcg` is the beautiful sharp image of a waterfall in Sapa, Vietnam. On top of it is a blurred image `.bcg2` (with z-index: 1) which starts fading in when the top of `#slide-5` is `33%` above the t of the viewport (`data--33p-top`). It becomes fully visible when the section is `66%` out of the view (`data--66p-top`).

`.bcg3` is the final screen which fades in between `300` and `100` pixels from the bottom of the pag and it's content `.hsContent` then fades in right at the end.

VIEW DEMO → DOWNLOAD FILES ↓

Part 2 – Released!

In part two you will learn **how to add a one page website navigation** using jQuery Waypoints

Go to Part 2 – Adding one page website navigation.

Conclusion

If you want to create a **parallax scrolling website** that works and looks great, keep these points in mind:

- **Less is more** – avoid lots of elements flying quickly through the viewport. Subtle movements softened by fading in and out usually look the best.
- **Keep it natural** – avoid cars animating vertically, their natural way is horizontal movement. A car coming from the side into the viewport will be less distracting than car falling down from t top.
- **Make it readable** – try to avoid text animating over objects with the same color, it will becom unreadable.
- **Timing is everything** – make sure that your content is perfectly aligned and everything look the best when the section is centered in the viewport. Having it messy and out of place befor and after is part of the beauty of parallax scrolling websites.
- **Have fun** – you will create an amazing things when you will enjoy playing with different settings and effect.
- **Keep trying** – you might not get it right the first time, just keep trying and things will improve.

Let me know in the comments below what are your **tricks or struggles**when it comes to developir parallax scrolling websites. I would love to hear your thoughts.