**Advanced CSS and Sass: Take Your CSS to the Next Level!**  
<https://www.udemy.com/advanced-css-and-sass/learn/v4/content>

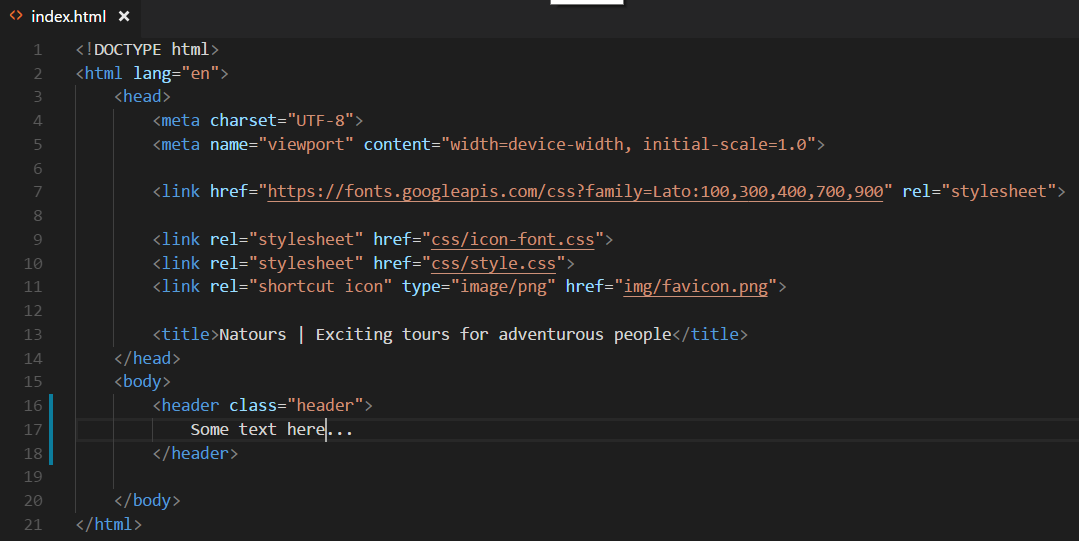
👉 [Course material and instructions on GitHub](https://github.com/jonasschmedtmann/advanced-css-course)

<https://github.com/jonasschmedtmann/advanced-css-course>

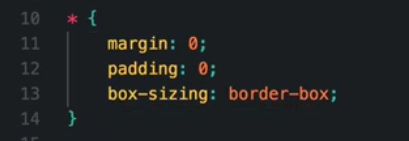
During the course, we use a couple of online tools and resources, which are all on my resources page. Students love this page, because it contains tons of useful design and coding resources. Here is the link:

[👉 Jonas' resources page](http://codingheroes.io/resources/)

👉 If you want to get updates on new courses or other stuff, just follow me on twitter [@jonasschmedtman](https://twitter.com/jonasschmedtman). I'm also on [facebook](https://www.facebook.com/codingheroes/" \t "_blank) if you prefer that.

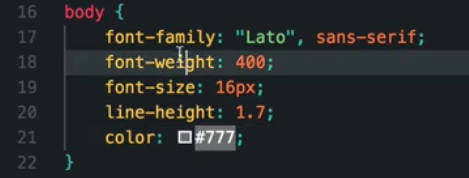


Start working on CSS, **Basic Reset** with **universal selector \***:



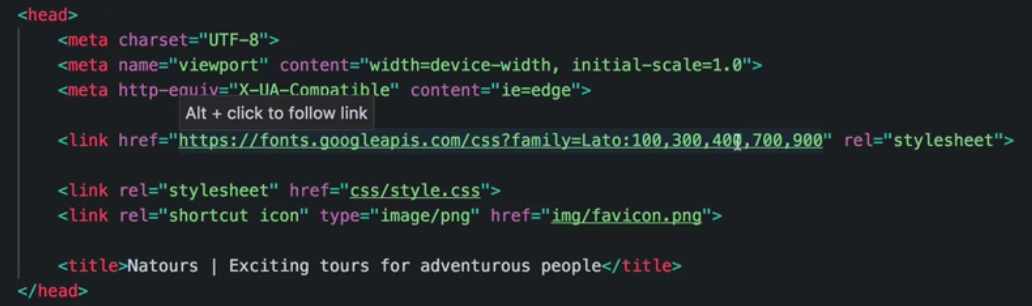
**margin** 0px, **padding** the same 0px.  
Then set **box-sizing** property to border-box – change box model, not to add the borders and paddings to total width or total height that is specified for the box.  
<https://www.w3schools.com/cssref/css3_pr_box-sizing.asp>

Add **font properties** to **body** element selector – because properties related to **font are usually inherited**. Power of inheritance is used here – more efficient for font element – not user universal selector here.



**Line-height 1.7** is 1.7 times bigger then predefined line height.

**Font-family “Lato”** can be used, because provided in head element with different weight 100, 300, 400, 700, 900:

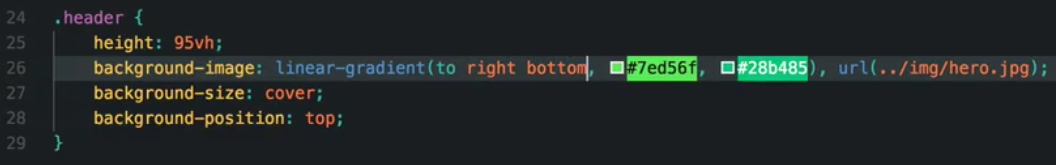


Start formatting header element – with header class:



Define **height** as **95vh** – 95% of viewport height.  
**background-size** to **cover** – whatever with of viewport, or element, it will try to fit the element inside of the box. <https://www.w3schools.com/cssref/css3_pr_background.asp>   
And make sure that top of the image stays the same with: **background-position** set to **top**

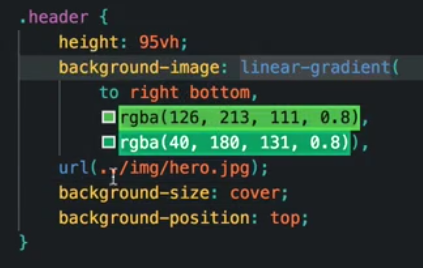
To specify **green gradient all over that image** – we want it to be on the top, so must be specified as first background-image (coma saparated) – and we will use linear-gradient (<https://www.w3schools.com/css/css3_gradients.asp>)   
‘to right’ – mean that starts on the left and then go to the right.   
‘to right bottom’ – mean that starts on the left-top and then go to the right-bottom.



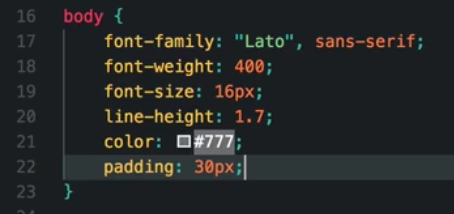
But now we cannot see image in the bottom of the gradient:



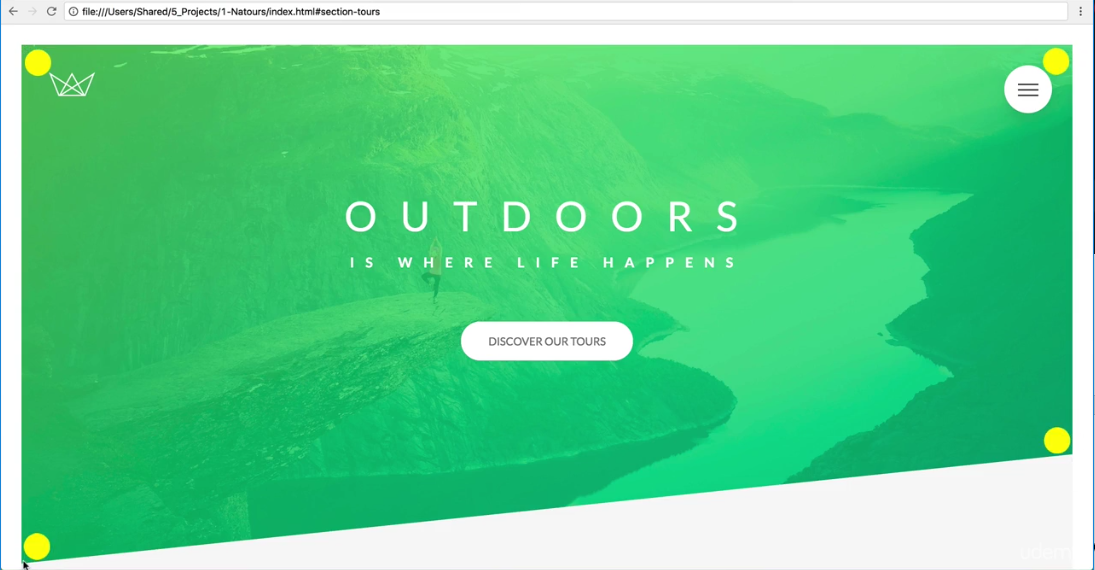
Need to **change opacity** of this colors to 0.8 - (https://www.w3schools.com/css/css3\_colors.asp).



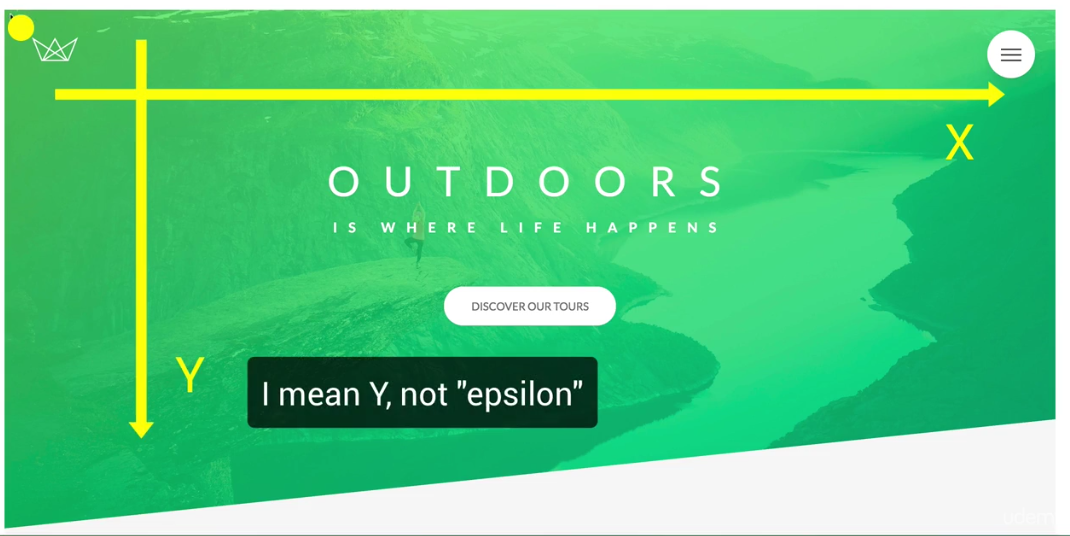
Now add padding around all body equal 30px (padding is not inherited)



**Clip-path polygon** help us with carving of our background (specify which part of the background will be visible) we need to specify 4 corners (their coordinates) started from upper left corner, and go clockwise.

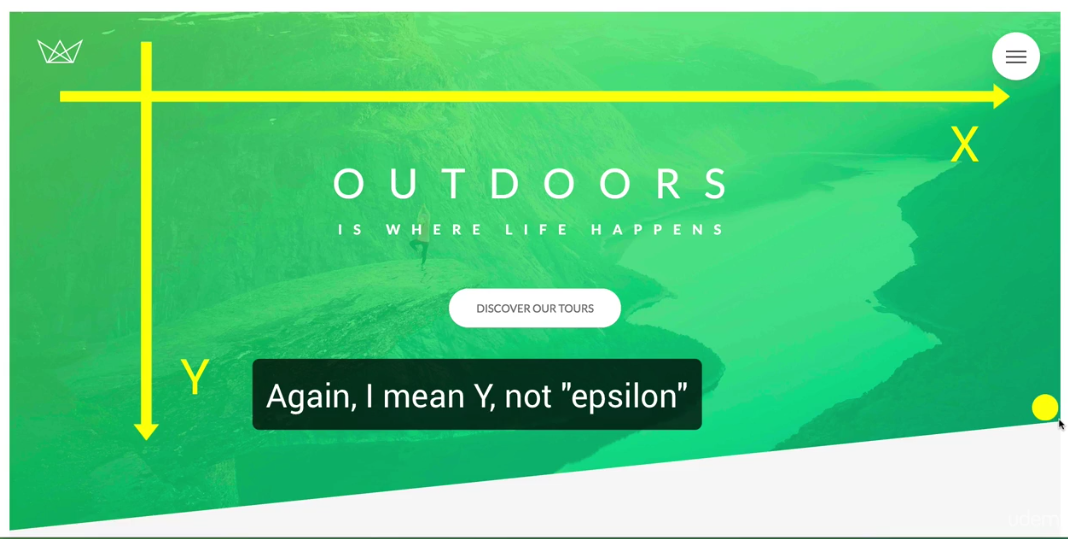


First point: (x y) => (0 0)



Second point – we want only x value change here, y stays the same as in fist point: (x y) => (100% 0)

Third point: x - stay the same as in second point (100%), y can change to any value like 90%, 200px or viewport height 75% of vh - counting from the top: (x y) => (100% 75vh)



The last one, back to 0 for x and take 100% on y: (x y) => (0 100%)



Grate tool to work with clip-path, called **clippy**: <https://bennettfeely.com/clippy/>