

## Module 10-UI UX,CSS Debugging, Future Strategy

### 10-1 Introduction to UI UX, Different layout with example websites

#### UI/UX ডিজাইন কি ?

UI হল User Interface যার মাধ্যমে ব্যবহারকারীদের কোন জিনিষের সাথে সংযোগ স্থাপন করা হয়। অর্থাৎ উদাহরণ হিসেবে কোন ওয়েব সাইট বা এপ এ কার জন্য কোথায় কি কি থাকবে এবং কোন অপশনের ইন্টারফেস কেমন হবে ইত্যাদি ডিজাইন করা। আর UX হল User Experience। কোনো পণ্যের টার্গেট কাস্টমার কারা, তারা কেন আপনার পণ্য কিনবে আর পণ্য কেনার পর সেই পণ্যের সাথে তাদের অভিজ্ঞতা কেমন ইত্যাদি বিষয় নিয়ে কাজ করাই হল UX ডিজাইন। মূলত UI ডিজাইন UX ডিজাইনের একটি অংশ।

একটি সুন্দর ও আকর্ষণীয় UI তৈরি করা একটি চ্যালেঞ্জ, বিশেষ করে এটি সহজে বোধগম্য হওয়া জরুরী। যেমন ধরুন, এক জরিপে কিছু লোককে জিজ্ঞাসা করা হয় স্যামসাং বা অ্যাপল মোবাইলে মধ্যে কোনটি আপনাদের বেশী পছন্দ, অধিকাংশের মতামত হল পছন্দের ক্ষেত্রে তারা অ্যাপল ফোনকেই বেশী পছন্দ করে। এর প্রধান কারন এটি অনেক বেশি ইউজার ফ্রেন্ডলী।

### 10-2 Rules to follow as best practice while designing a website

### 10-3 Five things to avoid for a good website design

### 10-4 Design Resources, figma edit and design secrets

### 10-5 CSS specificity, style priority and !important

If there are two or more conflicting CSS rules that point to the same element, the browser follows some rules to determine which one is most specific and therefore wins out.

## Specificity Hierarchy

Every selector has its place in the specificity hierarchy. There are four categories which define the specificity level of a selector:

**Inline styles** - An inline style is attached directly to the element to be styled.  
Example: `<h1 style="color: #ffffff;">`.

**IDs** - An ID is a unique identifier for the page elements, such as `#navbar`.

**Classes, attributes and pseudo-classes** - This category includes `.classes`, `[attributes]` and pseudo-classes such as `:hover`, `:focus` etc.

**Elements and pseudo-elements** - This category includes element names and pseudo-elements, such as h1, div, :before and :after.

## What is !important?

The `!important` rule in CSS is used to add more importance to a property/value than normal.

In fact, if you use the `!important` rule, it will override ALL previous styling rules for that specific property on that element!

The only way to override an `!important` rule is to include another `!important` rule on a declaration with the same (or higher) specificity in the source code - and here the problem starts! This makes the CSS code confusing and the debugging will be hard, especially if you have a large style sheet!

### 10-6 (advanced) CSS Custom property and use var, calc

calc=>The `calc()` function performs a calculation to be used as the property value.

Var=>The `var()` function is used to insert the value of a CSS variable.

### 10-7 CSS debugging using chrome devtool Elements tab

### 10-8 (Optional) Create a simple mega menu using HTML and CSS

Link: [https://www.w3schools.com/howto/howto\\_css\\_mega\\_menu.asp](https://www.w3schools.com/howto/howto_css_mega_menu.asp)

### 10-9 Advanced selectors, airbnb css style guides, BEM

Link:

1. <http://getbem.com/introduction/>
2. <https://www.smashingmagazine.com/2009/08/taming-advanced-css-selectors/>
3. <https://github.com/airbnb/css>
4. <https://css-tricks.com/bem-101/>

### 10-10 Your Future strategy for HTML(HTML5) and CSS(CSS3)

Link:

1. <https://www.tutorialrepublic.com/html-reference/html5-tags.php>
2. <https://www.tutorialrepublic.com/css-reference/css3-properties.php>

