# Responsive Webpage (Media Queries)

## Objective(s):

* To understand how to use media Queries.
* To understand how to make responsive web pages using media queries.

## Tool(s) used:

For example: Brackets.

## Media query

Media query is a CSS technique introduced in CSS3.

It uses the @media rule to include a block of CSS properties only if a certain condition is true. Media queries in CSS3 extend the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.

Media queries can be used to check many things, such as:

* width and height of the viewport
* width and height of the device
* Orientation (is the tablet/phone in landscape or portrait mode?)
* resolution

Using media queries are a popular technique for delivering a tailored style sheet to tablets, iPhone, and Androids.

## Viewport:

The viewport is the user's visible area of a web page. The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

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

* A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling.
* The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).
* The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

## Example:

@media only screen and (max-width: 500px) {  
    body {  
        background-color: lightblue;  
    } }

## Add a Breakpoint:

Earlier in this tutorial we made a web page with rows and columns, and it was responsive, but it did not look good on a small screen Media queries can help with that. We can add a breakpoint where certain parts of the design will behave differently on each side of the breakpoint.

## Example:

/\* For desktop: \*/  
.col-1 {width: 8.33%;}  
.col-2 {width: 16.66%;}  
.col-3 {width: 25%;}  
.col-4 {width: 33.33%;}  
.col-5 {width: 41.66%;}  
.col-6 {width: 50%;}  
.col-7 {width: 58.33%;}  
.col-8 {width: 66.66%;}  
.col-9 {width: 75%;}  
.col-10 {width: 83.33%;}  
.col-11 {width: 91.66%;}  
.col-12 {width: 100%;}  
  
@media only screen and (max-width: 768px) {  
    /\* For mobile phones: \*/  
    [class\*="col-"] {  
        width: 100%;  
    }  
}

## Always Design for Mobile First:

Mobile First means designing for mobile before designing for desktop or any other device (This will make the page display faster on smaller devices).

This means that we must make some changes in our CSS.

Instead of changing styles when the width gets *smaller* than 768px, we should change the design when the width gets *larger* than 768px. This will make our design Mobile First.

## Example:

/\* For mobile phones: \*/  
[class\*="col-"] {  
    width: 100%;  
}  
@media only screen and (min-width: 768px) {  
    /\* For desktop: \*/  
    .col-1 {width: 8.33%;}  
    .col-2 {width: 16.66%;}  
    .col-3 {width: 25%;}  
    .col-4 {width: 33.33%;}  
    .col-5 {width: 41.66%;}  
    .col-6 {width: 50%;}  
    .col-7 {width: 58.33%;}  
    .col-8 {width: 66.66%;}  
    .col-9 {width: 75%;}  
    .col-10 {width: 83.33%;}  
    .col-11 {width: 91.66%;}  
    .col-12 {width: 100%;}  
}

## Tasks:

### Task 01. Time: 30 Minutes

Make this page responsive

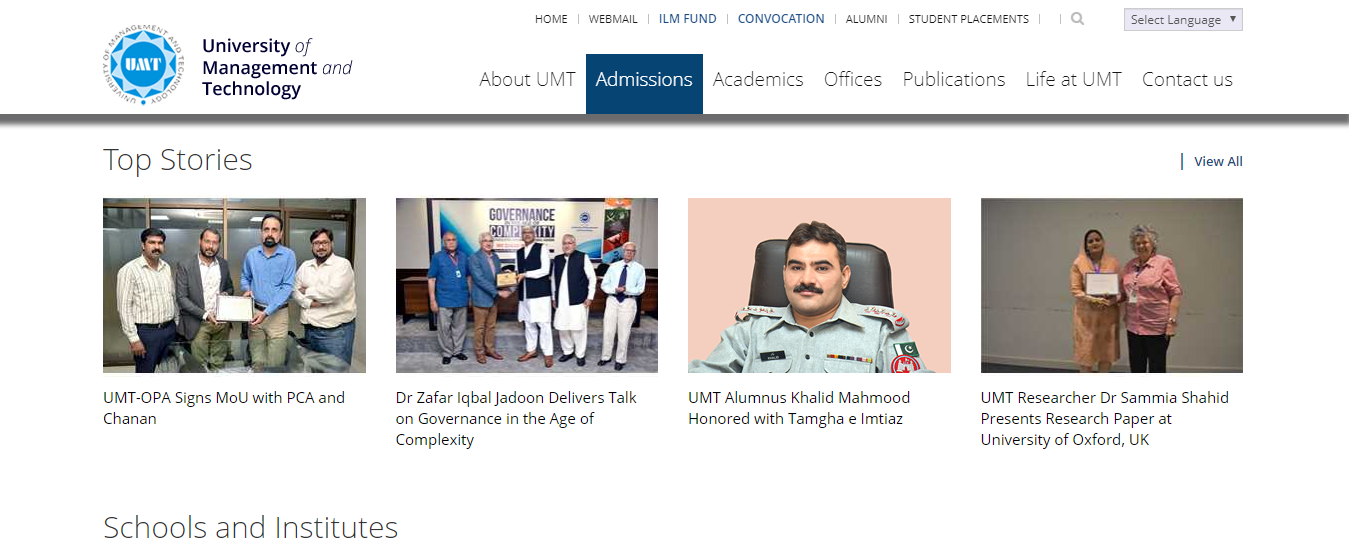


Figure 06‑1

### Task 02. Time: 70 Minutes

Using previous lab web pages(given below) make them responsive.

* Navigation bar should be responsive.
* Picture in web page also should be responsive.

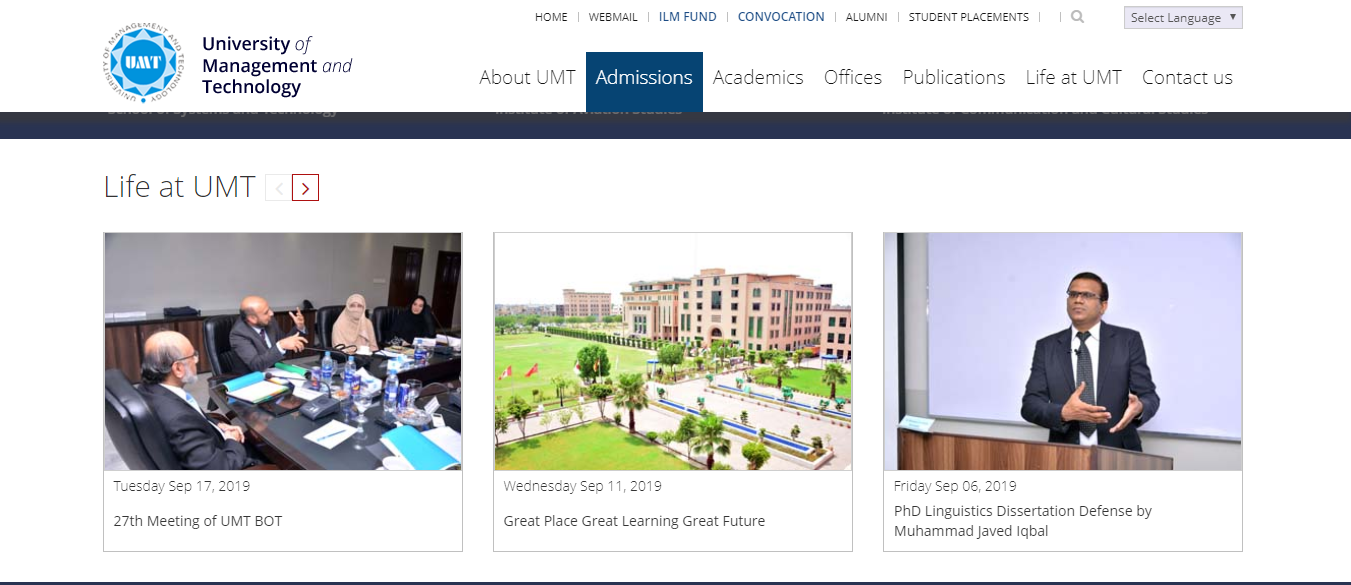
i). 

Figure 06‑2

### ii).

## 

Figure 06‑3