**UNIT-I**

1. HTML Basics

**HTML Documents**

All HTML documents must start with a document type declaration: <!DOCTYPE html>.

The HTML document itself begins with <html> and ends with </html>.

The visible part of the HTML document is between <body> and </body>.

Example

<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My First Heading</h1>  
<p>My first paragraph.</p>  
  
</body>  
</html>

**HTML Headings**

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading:

Example

<h1>This is heading 1</h1>  
<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>

**HTML Paragraphs**

HTML paragraphs are defined with the <p> tag:

Example

<p>This is a paragraph.</p>  
<p>This is another paragraph.</p>

**HTML Links**

HTML links are defined with the <a> tag:

Example

<a href="https://www.w3schools.com">This is a link</a>

The link's destination is specified in the href attribute.

Attributes are used to provide additional information about HTML elements.

**HTML Images**

HTML images are defined with the <img> tag.

The source file (src), alternative text (alt), width, and height are provided as attributes:

Example

<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">

**HTML Buttons**

HTML buttons are defined with the <button> tag:

Example

<button>Click me</button>

**HTML Lists**

HTML lists are defined with the <ul> (unordered/bullet list) or the <ol> (ordered/numbered list) tag, followed by <li> tags (list items):

Example

<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>  
<ol>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ol>

1. **HTML RESPONSIVE**

What is Responsive Web Design?

Responsive Web Design is about using HTML and CSS to automatically resize, hide, shrink, or enlarge, a website, to make it look good on all devices (desktops, tablets, and phones):

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_page)

**Note:** A web page should look good on **any device**!

Setting The Viewport

When making responsive web pages, add the following <meta> element in all your web pages:

Example

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

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_viewport)

This will set the viewport of your page, which will give the browser instructions on how to control the page's dimensions and scaling.

Here is an example of a web page *without* the viewport meta tag, and the same web page *with* the viewport meta tag:

Without the viewport meta tag: With the viewport meta tag:

[](https://www.w3schools.com/html/example_withoutviewport.htm) 

**Tip:** If you are browsing this page on a phone or a tablet, you can click on the two links above to see the difference.

Responsive Images

Responsive images are images that scale nicely to fit any browser size.

Using the width Property

If the CSS width property is set to 100%, the image will be responsive and scale up and down:



Example

<img src="img\_girl.jpg" **style="width:100%;"**>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_image)

Notice that in the example above, the image can be scaled up to be larger than its original size. A better solution, in many cases, will be to use the max-width property instead.

Using the max-width Property

If the max-width property is set to 100%, the image will scale down if it has to, but never scale up to be larger than its original size:



Example

<img src="img\_girl.jpg" style="**max-width:100%;**height:auto;">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_image_maxwidth)

Show Different Images Depending on Browser Width

The HTML <picture> element allows you to define different images for different browser window sizes.

Resize the browser window to see how the image below change depending on the width:



Example

<picture>  
  <source srcset="img\_smallflower.jpg" media="(max-width: 600px)">  
  <source srcset="img\_flowers.jpg" media="(max-width: 1500px)">  
  <source srcset="flowers.jpg">  
  <img src="img\_smallflower.jpg" alt="Flowers">  
</picture>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_picture)

Responsive Text Size

The text size can be set with a "vw" unit, which means the "viewport width".

That way the text size will follow the size of the browser window:

Hello World

Resize the browser window to see how the text size scales.

Example

<h1 style="**font-size:10vw**">Hello World</h1>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_text)

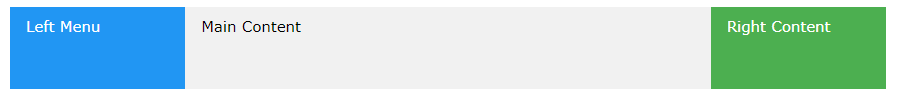
Viewport is the browser window size. 1vw = 1% of viewport width. If the viewport is 50cm wide, 1vw is 0.5cm.

Media Queries

In addition to resize text and images, it is also common to use media queries in responsive web pages.

With media queries you can define completely different styles for different browser sizes.

Example: resize the browser window to see that the three div elements below will display horizontally on large screens and stacked vertically on small screens:



Example

<style>  
.left, .right {  
  float: left;  
  width: 20%; /\* The width is 20%, by default \*/  
}  
  
.main {  
  float: left;  
  width: 60%; /\* The width is 60%, by default \*/  
}  
  
/\* Use a media query to add a breakpoint at 800px: \*/  
@media screen and (max-width: 800px) {  
  .left, .main, .right {  
    width: 100%; /\* The width is 100%, when the viewport is 800px or smaller \*/  
  }  
}  
</style>

**Tip:** To learn more about Media Queries and Responsive Web Design, read our [RWD Tutorial](https://www.w3schools.com/css/css_rwd_intro.asp).

Responsive Web Page - Full Example

A responsive web page should look good on large desktop screens and on small mobile phones.

**Responsive Web Design - Frameworks**

There are many existing CSS Frameworks that offer Responsive Design.

They are free, and easy to use.

**Using W3.CSS**

A great way to create a responsive design, is to use a responsive style sheet, like [W3.CSS](https://www.w3schools.com/w3css/default.asp)

W3.CSS makes it easy to develop sites that look nice at any size; desktop, laptop, tablet, or phone:



Example

<!DOCTYPE html>  
<html>  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">  
<body>  
  
<div class="w3-container w3-green">  
  <h1>W3Schools Demo</h1>  
  <p>Resize this responsive page!</p>  
</div>  
  
<div class="w3-row-padding">  
  <div class="w3-third">  
    <h2>London</h2>  
    <p>London is the capital city of England.</p>  
    <p>It is the most populous city in the United Kingdom,  
    with a metropolitan area of over 13 million inhabitants.</p>  
  </div>  
  
  <div class="w3-third">  
    <h2>Paris</h2>  
    <p>Paris is the capital of France.</p>  
    <p>The Paris area is one of the largest population centers in Europe,  
    with more than 12 million inhabitants.</p>  
  </div>  
  
  <div class="w3-third">  
    <h2>Tokyo</h2>  
    <p>Tokyo is the capital of Japan.</p>  
    <p>It is the center of the Greater Tokyo Area,  
    and the most populous metropolitan area in the world.</p>  
  </div>  
</div>  
  
</body>  
</html>

**Bootstrap**

Another popular framework is Bootstrap, it uses HTML, CSS and jQuery to make responsive web pages.

Example

<!DOCTYPE html>  
<html lang="en">  
<head>  
<title>Bootstrap Example</title>  
<meta charset="utf-8">  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js"></script>  
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  
</head>  
<body>  
  
<div class="container">  
  <div class="jumbotron">  
    <h1>My First Bootstrap Page</h1>  
  </div>  
  <div class="row">  
    <div class="col-sm-4">  
      ...  
    </div>  
    <div class="col-sm-4">  
      ...  
    </div>  
    <div class="col-sm-4">  
    ...  
    </div>  
  </div>  
</div>  
  
</body>  
</html>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_bootstrap)

Click on below link to see the output:

<https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_bootstrap>

# Responsive Web Design - Introduction

What is Responsive Web Design?

Responsive web design makes your web page look good on all devices.

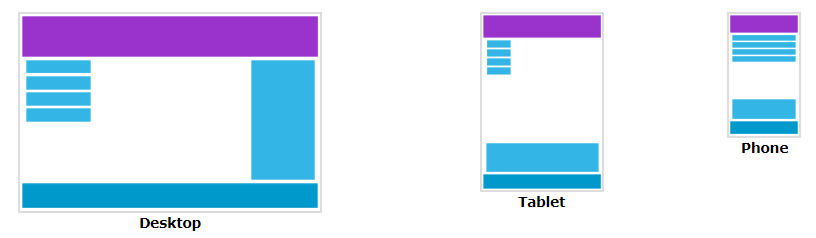
Responsive web design uses only HTML and CSS.

Responsive web design is not a program or a JavaScript.

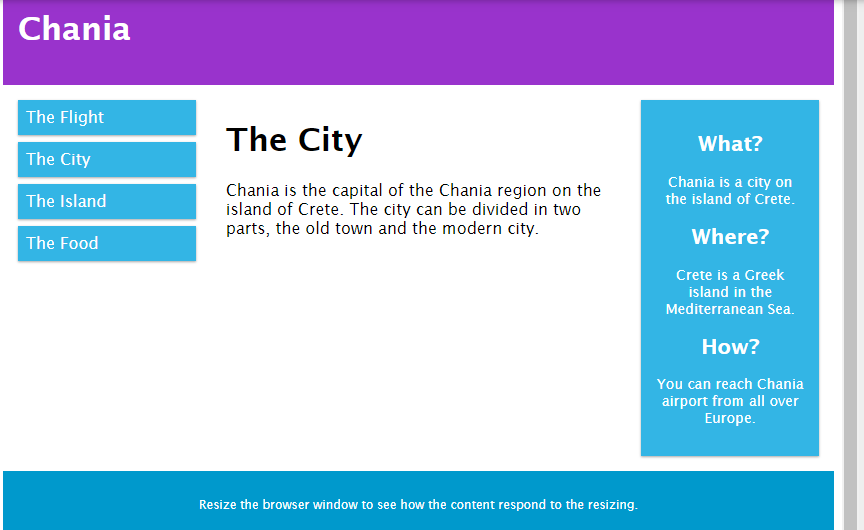
**Designing For The Best Experience For All Users**

Web pages can be viewed using many different devices: desktops, tablets, and phones. Your web page should look good, and be easy to use, regardless of the device.

Web pages should not leave out information to fit smaller devices, but rather adapt its content to fit any device:



It is called responsive web design when you use CSS and HTML to resize, hide, shrink, enlarge, or move the content to make it look good on any screen.



What is The 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.

Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.

Then, when we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

This was not perfect!! But a quick fix.

Setting The Viewport

HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

You should include the following <meta> viewport element in all your web pages:

<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.

Here is an example of a web page *without* the viewport meta tag, and the same web page *with* the viewport meta tag:

**Tip:** If you are browsing this page with a phone or a tablet, you can click on the two links below to see the difference.

[[](https://www.w3schools.com/css/example_withoutviewport.htm)    
  
**Without the viewport meta tag**](https://www.w3schools.com/css/example_withoutviewport.htm) [**With the viewport meta tag**](https://www.w3schools.com/css/example_withviewport.htm)

Size Content to The Viewport

Users are used to scroll websites vertically on both desktop and mobile devices - but not horizontally! So, if the user is forced to scroll horizontally, or zoom out, to see the whole web page it results in a poor user experience.

Some additional rules to follow:

**1. Do NOT use large fixed width elements -**For example, if an image is displayed at a width wider than the viewport it can cause the viewport to scroll horizontally. Remember to adjust this content to fit within the width of the viewport.

**2. Do NOT let the content rely on a particular viewport width to render well** - Since screen dimensions and width in CSS pixels vary widely between devices, content should not rely on a particular viewport width to render well.

**3. Use CSS media queries to apply different styling for small and large screens** - Setting large absolute CSS widths for page elements will cause the element to be too wide for the viewport on a smaller device. Instead, consider using relative width values, such as width: 100%. Also, be careful of using large absolute positioning values. It may cause the element to fall outside the viewport on small devices.

# Responsive Web Design - Grid-View

## What is a Grid-View?

Many web pages are based on a grid-view, which means that the page is divided into columns: