**WEB -DEVELOPMENT PROJECT**

**Table of contents**

What is HTML?

What to Expect from HTML Projects?

Features of HTML

Why HTML projects?

10 Best HTML Projects for Beginners

1. Survey Form

2. Website Products Display

3. Technical Documentation Page

4. Adding Animation

5. Animated Website Home Page

6. Video Background Site

7. Create an Image Slider

8. Interactive Restaurant Website

**What is HTML?**

HTML is the heart of the entire web. Whatever content we see online is based on HTML. If you simply right-click on any web page and click on View Source, you will be able to see loads of tags, the first one being <html> which means the document is an HTML file.

**The basic structure of HTML:**

<!DOCTYPE HTML>

<html>

<head>

<title>Title of the page</title>

</head>

<body>

<form action="some\_action" method="POST">

</form>

</body>

</html>

Features of HTML:

Before we get onto projects, let us quickly recap the highlights:

Platform-independent mark-up language.

Easy to use and learn.

No need for any setup or installation.

Links, images, video, audio, animations can be added with little coding.

Not case-sensitive.

Easy to write custom code in other languages and integrate with HTML.

Supported by all browsers.

Free!

Why HTML projects?

Projects are the most practical way to learn any language, and HTML is no exception. To build any website, you need a basic understanding of HTML and if you want interactive custom websites, just knowing the cheat sheet is not enough – you should be able to implement your knowledge practically.

**HTML Projects :**

**Let us now take a look at the best HTML projects to enhance your web development skills.**

**1. Survey Form:**

The survey form is one of the most straightforward HTML projects you can start. In this project, we will create a simple survey form and then submit the information captured. You can open any text editor to write your code, and save your file with .html extension. For example, survey.html

You will learn about basic input tags, form creation, radio buttons, checkboxes, etc.

Here is the source code for the project with the necessary explanations of the tags. You can add elements one by one to see how they work.

Every file should start with the <html> tag and end with </html> end tag.

<!DOCTYPE HTML>

<!-- This is how HTML comments look like -->

<html>

<!-- the title will appear on the page-->

<head>

<title>Employee Interests Survey</title>

</head>

<body>

<!-- as it is a survey form, we will need to submit the details, hence we use form -->

<!-- We can give absolute url, or relative url like /nextpage.jsp, and specify POST or GET method -->

<form action="http://google.co.in">

<!-- If we remove this, every thing will move to the left of the page-->

<div align="center">

<!--Adds a heading to the form-->

<h1>Employee Interests Survey form</h1>

Enter your name:

<!-- Input type text for small texts, specify size -->

<input type="text" name="UserName" size=35 maxlength=35 value="">

<!--Adds spaces - remove and see what happens -->

</br></br>

Enter your department:

<input type="text" name="Deptt" size=35 maxlength=35 value=""> </br> </br>

Tell us a little about yourself:

<!-- For writing lot of text like descriptions with text wrapping,

if you dont want text wrapping, you can add wrap = "off" (horizontal scrollbar -->

<textarea name="Comments" cols=30 rows=4></textarea> </br> </br>

Do you exercise at home?

<!-- Radio buttons help you choose one out of the many values -->

<input type="radio" name="exe" value=1>Yes

<input type="radio" name="exe" value=2>No

</p>

How do you like to read about your favorite topics?

<p>

<!--Checkbox lets you select multiple options -->

<input type="checkbox" name="Books">Books

<input type="checkbox" name="Web">Online resources

<input type="checkbox" name="Phone">Phone apps

<input type="checkbox" name="Magazines">Magazines

</p>

What genre of movies do you like?

<!--Select box lets you choose one of the multiple dropdown options-->

<select name="moviepref" ><option>

<option value=1 selected = "true">comedy

<option value=2 >romance

<option value=3 >thriller

<option value=4 >horror

<option value=5 >biopic

</select>

</br></br>

<!--submits the information entered in the form by the user -->

<input type=submit value="Submit form">

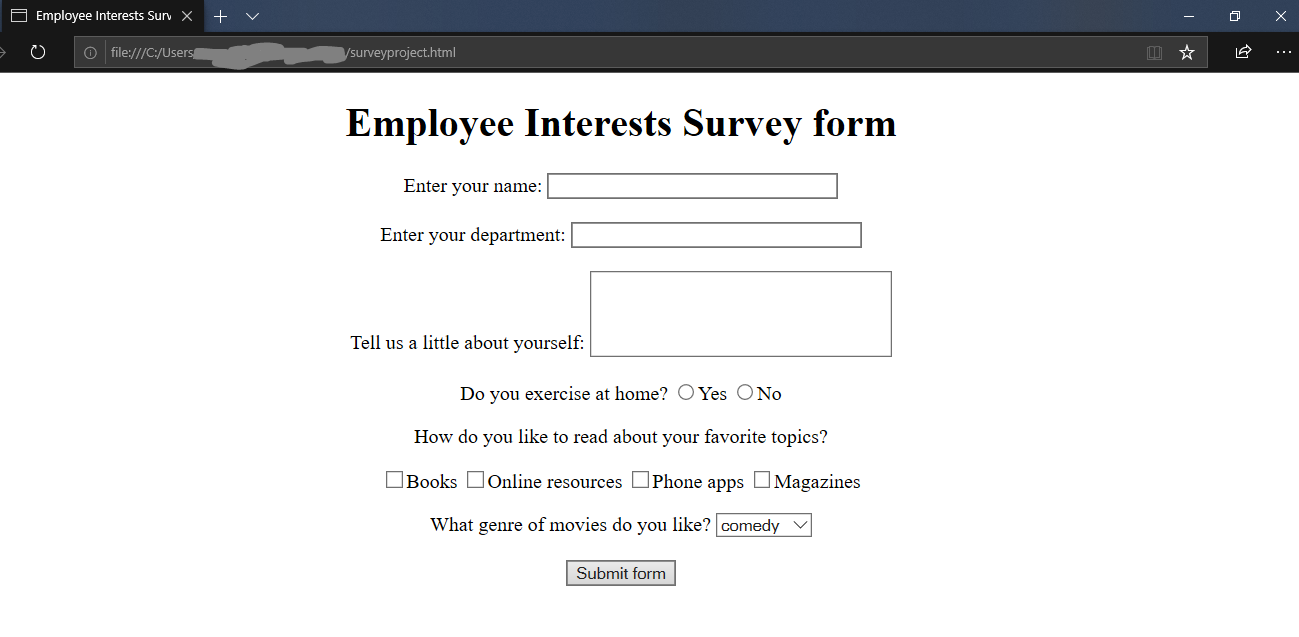
</div>

</form>

</body>

</html>

Here is how your page will look like:



**2. Website Products Display :**

Product display or product landing page is an exciting project, where you will learn how products are displayed on a website, most interactively.

The below video displays multiple products, one after the other. You will be amazed at how the page is wholly transformed using simple CSS styling.

Display products side by side

In another video mentioned below, you will learn how to create a product card, i.e. detailed information about a product.

Product card that displays product information

These two are cool projects through which you will learn how easily HTML and CSS go hand in hand to create beautiful UI.

**3. Technical Documentation Page:**

Every language, software, hardware, etc. has technical documentation for information and help.Here is how you can create a simple documentation page, with internal linking from the left navigation to the content on the right of the page. You can check out the documentation for Python, Java, Arduino, etc. for reference on how these pages look. Most technical documentation has multiple hierarchies and pages; however, for simplicity, we have created a simple project to start. You can expand it further and add it as per your wish. Technical documentation could be a beneficial HTML project if you are just starting with your HTML career.

<html>

<!--This example uses the default bootstrap stylesheet-->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"

integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">

<!--provides a full-width container that can expand or collapse based on the size of viewport-->

<div class="container-fluid">

<div class="row">

<div class="col-md-2 col-sm-12 col-xs-12">

<nav id="navbar">

<h3>Technical Documentation</h3>

<!--content stacking for smaller screens-->

<ul class="nav nav-pills nav-stacked">

<!--internal linking to the respective sections-->

<a class="nav-link" href="#Introduction" rel="internal">

<li>Introduction</li>

</a>

<a class="nav-link" href="#What\_you\_should\_already\_know" rel="internal">

<li>What you should already know</li>

</a>

<a class="nav-link" href="#About\_Topic" rel="internal">

<li>About the topic</li>

</a>

<a class="nav-link" href="#Topic\_1" rel="internal">

<li>Topic 1</li>

</a>

<a class="nav-link" href="#Topic\_2" rel="internal">

<li>Topic 2</li>

</a>

</ul>

</nav>

</div>

<div class="col-md-10 col-sm-12 col-xs-12">

<main id="main-doc">

<section class="main-section" id="Introduction" >

<!--basic styling for the headings, better practice to do the same in a css file as the styling is same for all headers-->

<h3 style = "background: black; color: white">Introduction</h3>

<article>

<p>Some content about the main topic, for example Java documentation introduction about the language

</article>

</section>

<section class="main-section" id="What\_you\_should\_already\_know">

<h3 style = "background: black; color: white">What you should already know</h3>

<article>

<p>Background information before getting into the topic:</p>

<!--creating list using html-->

<li>Some list content</li>

<li>Prerequisites.</li>

<li>Workings and assumptions</li>

<p>Any other content to be covered before learning this topic</p>

</artice>

</section>

<section class="main-section" id="About\_Topic">

<h3 style = "background: black; color: white">About topic</h3>

<article>

<p>More lines about the topic. For example, how the basic functionality works, features etc...</p>

<p>

Technical documentation should be thorough and to the point

</p>

<p>

Write about features, comparisons with other languages etc

</p>

</article>

</section>

<section class="main-section" id="Topic\_1">

<h3 style = "background: black; color: white">Topic 1</h3>

<article>

Getting started with the actual documentation content

<code>This would come in a different color and font indicating lines of code

</code>

</article>

</section>

<section class="main-section" id="Topic\_2">

<h3 style = "background: black; color: white">Topic 2</h3>

<p>

Another topic about the main topic, for example, if the topic is Java, this could be variables or data types in Java

</p>

</section>

</main>

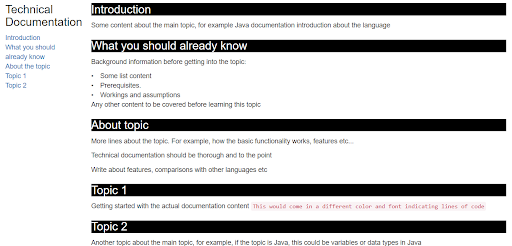
</div>

</div>

</div>

</html>

You should get a page like this:



**4. Adding Animation:**

In this project, we will create a simple page, where a car will appear to move on a road. The simple project is a perfect example of how animations can be made easily and quickly. Look out for the use of <marquee> tag for creating this simple animation. Try adding more objects and making them move in different directions for more fun.

**5. Animated Website Home Page:**

In this project, we will create a homepage that will display a logo and some menu items. You can also have some fun with the keyframes tag for animations. Note how the navigation menu and the rest of the page are separated using the height property. You can apply a background image, and change the height to 85vh or 100vh and see the difference it makes.

<html>

<head>

<title></title>

<!-- Apply styles to html elements -->

<style>

\*{

margin:0; padding:0; boxsizing:border-box;

}

header{

width: 100%; height: 100vh;

background-color: white;

background-repeat: no-repeat;

background-size: cover;

}

nav{

width: 100%; height: 15vh;

background: black;

display: flex; justify-content: space-between;

align-items: center;

}

nav .mainmenu{

width: 40%;

display: flex; justify-content: space-around;

}

main{

width: 100%; height: 85vh;

display: flex; justify-content: center;

align-items: center;

text-align: center;

}

section h3{

letter-spacing: 3px; font-weight: 200;

}

section h1{

text-transform: uppercase;

}

section div{

animation:changeborder 10s infinite linear;

border: 7px solid red;

}

@keyframes changeborder{

0%

20%

40%

}

</style>

</head>

<body>

<!--Let us create a simple menu using the navigation tags-->

<!--Use header to indicate that manu will be a part of header -->

<header>

<nav>

<div class = "logo" <h3 style="color:white;">MYLOGO</h3></div>

<!--Lets define the menu items now-->

<div class = "mainmenu">

<a href="https://hackr.io/tutorials/learn-html-5">Home</a>

<a href="https://hackr.io/tutorials/learn-html-5">About Us</a>

<a href="https://hackr.io/tutorials/learn-html-5">Contact Us</a>

</div>

</nav>

<!--Let us create the main section now, if you are not using html5, use div tags-->

<main>

<section>

<!--Check out the styling elements for this div class - change\_text -->

<div class = "change\_text"><b>WELCOME TO XXX ELECTRONICS</b></div>

<!--make text italic-->

<p><i>All your electronics needs in one place</i></p><br>

<!--create a button, if there is a form, you can specify an action on click-->

<input type = button value = "view more">

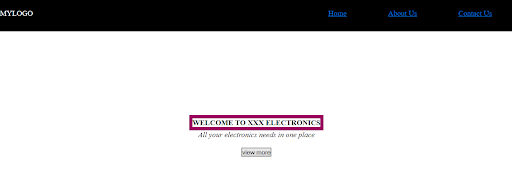
</section>

</main</header>

</body>

</html>

The above code will produce the following output. The color of the box will vary.



**6.video background animation:**

This project explains with full source code, how we can add a video background to a web page. The foreground contains some text. The project uses HTML5. The speaker also tells a simple trick to avoid scrollbars. You will also learn to add text at the forefront and style it to appear in the centre of the page. If you resize the page, the video resizes and comes as per the screen size by setting simple CSS values. Videos in the background can give a lovely effect to your website and make it interactive. View the video background animation for the code and explanations.

**7. Create an Image Slider :**

Image slider or slide show is a prevalent functionality that most websites show today. In any site, for example, fashion, food, services, etc., you can observe that most of the content is shown through multiple images. If you know how to display images as a slideshow, the other content can be developed by using plain input and other data types. The entire project has been done using only HTML and CSS, and no JavaScript, thus avoiding any cross-browser issues. The speaker explains the functionality in detail here.

**8. Interactive Restaurant Website:**

In this project, we will create an interactive homepage of a restaurant, where the background image will change when we hover over a menu option. For example, if you hover over menu navigation, it will display a menu card, if you hover over the home, it will show the default. You have to give your image names and locations for this to work. This functionality is done through jQuery. Same way, the ‘order now’ button will keep flashing. We have done this by using @keyframes functionality which we have used in our animated website project too. Here is the code for this:

<html>

<head>

<style>

body{

font-size:20px;

color: white;

background-size: cover;

}

.box{ width: 900px;

float:right;

border:1px solid none;}

.box ul li{

width: 120px;

float:left;

margin: 10px auto;

text-align: center;

}

.mainmenu

.mainmenu a

.mainmenu a:hover

.mainmenu img{

position: fixed;

z-index: -1;

top:0px; left:0px; width:100%; height: 100vh;

opacity: 0.9;

/\*object-fit:cover;\*/

transition: all ease 0.5s;

}

.wd{

width: 300px;

height: 539px;

background-color: black;

opacity: 0.8;

padding: 55px;

}

.wd h1{

text-align: center;

text-transform: uppercase;

font-weight: 300px;

}

.wd h4{

text-align: justify;

color:darkgray;

font-weight: 100px;

}

.wd h2{

text-align: center;

text-transform: uppercase;

font-weight: normal;

margin: 40px auto;

}

.opt form , input[type="button"]{

background-color: black;

color:white;

/\* padding:10px;\*/

margin:-14px auto;

padding-left: 50px;

padding-right: 50px;

text-align: center;

font-size: 16px;

}

form, input[type="button"] {

animation: glowing 300ms infinite;

font-weight: 500%;

}

@keyframes glowing {

0% {

background-color: red;

}

50% {

background-color: orange;

}

100% {

background-color: blue;

}

}

</style>

</head>

<body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>

<script>

$(function(){

var image = $(".mainmenu").find('img').attr('src');

$(".mainmenu a").mouseover(function(){

var newimg = $(this).attr('data-image');

$(this).parent().find('img').attr("src", newimg);

});

});

</script>

<div class="box">

<div class="mainmenu">

<img src="food.png">

<a data-image = "food.png" href=""> Home</a>

<a data-image = "menucard.jpg" href=""> Menu</a>

<a href=""> FAQ</a>

<a href=""> Contact</a>

</div>

</div>

<div class="wd">

<h1> Welcome to xxx</h1>

<h4> <i>Order delicious food online, 100% quality, safety and taste assured.</i></h4>

<h2> Call 999999999 for reservations</h2>

<div class="opt">

<form action="" method="post">

<input type="button" value="ORDER NOW">

</form>

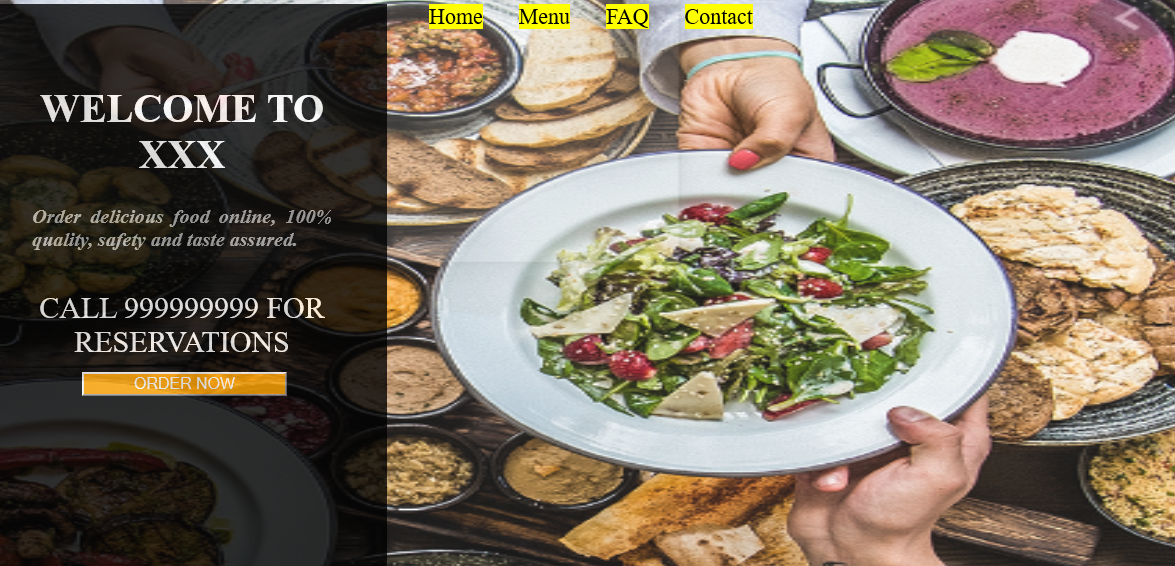
</div>

</div>

</body>

</html>

Output:



**Submitted by:**

**S.Parkavi Saravanan.**

**Department of ECE.**