

.NET Syllabus

For Apprenticeship Training Program



HINDTECH
LEARNING POINT



कौशल भारत - कुशल भारत



Transforming the skill landscape

What is .NET?

.NET is a software development framework created by Microsoft that supports building and running applications on Windows, Linux, and macOS. It provides a large class library called the .NET Class Library, and supports several programming languages, primarily C#, VB.NET, and F#.

Key Features:

- Cross-platform: Build apps that run on Windows, Linux, and macOS.
- Multiple Language Support: Mainly C#, but also supports VB.NET, F#, etc.
- Rich Libraries: Offers built-in libraries for security, file I/O, database access, etc.
- Unified Development: One platform for Web, Desktop, Mobile, and Cloud apps.
- Modern & Fast: Uses the latest programming features and performs well.

Common Uses of .NET:

- Web applications (using ASP.NET or Blazor)
- Desktop applications (using Windows Forms or WPF)
- Mobile apps (using .NET MAUI or Xamarin)
- Cloud services (Azure integration)
- APIs and microservices

.NET is a free, open-source, cross-platform software development platform developed by Microsoft that allows developers to build a wide range of applications, including web, desktop, mobile, cloud, and IoT solutions. It provides a unified environment where different programming languages, such as C#, VB.NET, and F#, can be used to write applications. Originally introduced as the .NET Framework for Windows-only development, the platform has evolved into a modern and powerful system with the release of .NET Core and now the unified ".NET" version (starting from .NET 5 and beyond), which supports Windows, Linux, and macOS. The core part of .NET is the Common Language Runtime (CLR), which manages memory, handles exceptions, and ensures security. Alongside this, the .NET Class Library offers a large collection of reusable code that simplifies common programming tasks like file handling, database interaction, web requests, and more.

.NET also supports modern development patterns like object-oriented programming, dependency injection, asynchronous programming, and RESTful APIs.

It includes powerful tools like Visual Studio and Visual Studio Code, which provide developers with intelligent code completion, debugging, and integrated version control. One of the most widely used components of .NET is ASP.NET Core, which allows developers to build high-performance web applications and APIs.



HTML : MODULE 1**Introduction:**

- What is HTML?
 1. Definition and importance of HTML
 2. Role of HTML in web development
 3. Overview of how HTML works with CSS and JavaScript
- History Of HTML ?
 1. Evolution of HTML from HTML 1.0 to HTML5
 2. Key milestones and developments in HTML history
 3. Impact of HTML advancements on web development
- Basic Structures of a Web Page
 1. Fundamental structure of an HTML document
 2. Understanding the `<!DOCTYPE html>` declaration
 3. Introduction to HTML elements and attributes

HTML Head Tags:

- Purpose and Use of the `<head>` Section
 1. Common head tags: `<title>`, `<meta>`, `<link>`, `<style>`, `<script>`, `<meta viewport="">`
 2. Using meta tags for SEO and responsive design

HTML Body Tags:

- Structure and Purpose of the `<body>` Section
 1. Creating and organising content within the body
 2. Basic HTML tags for structuring web pages

HTML Basic Tags:

- `
` and `<hr>` Tags
 1. Usage and examples of line breaks with `
`
 2. Creating horizontal rules with `<hr>`
- `<p>`, `<h1>` to `<h6>`, `<a>`, ``, ``, ``, ``, `<div>`, ``, ``, ``, `<table>`, `<tr>`, `<td>`, `<th>`, `<form>`, `<input>`, `<button>`

HTML Link Creation:

- HTML Anchor Tags
 1. Creating hyperlinks with the `<a>` tag
 2. Understanding `href` attribute and link destinations
 3. Opening links in new tabs or windows with `target` attribute

HTML Tables

- HTML Tables
 1. Constructing tables with `<table>`, `<tr>`, `<td>`, and `<th>` tags
 2. Basic table structure and usage

```

<!DOCTYPE html>
<html>
  <head>
    <title>Wikitechy div tag</title>
  </head>
  <body>
    <h2>Wikitechy HTML div tag.
    <div style="color: red">
      <h2>This is a heading in
      <p> This is a paragraph i
    </div>
  </body>
</html>
  
```

- HTML Nested Tables
 1. Creating tables within tables
 2. Managing nested table layouts
- HTML Merging Cells
 1. Merging table cells with **colspan** and **rowspan** attributes
 2. Practical examples of cell merging
- HTML Text Wrapping
 1. Wrapping text within table cells
 2. Handling overflow and text alignment
- HTML Table Background Image
 1. Adding background images to tables or cells
 2. Customising table aesthetics with background images
- HTML Table Cell Alignment
 1. Aligning text and content within table cells
 2. Using **align** and **valign** attributes

HTML : MODULE 2

HTML Lists:

- HTML Unordered Lists
 1. Creating bulleted lists with ****
 2. Using **** for list items
 3. Styling unordered lists
- HTML Ordered Lists
 1. Creating numbered lists with ****
 2. List item customization and attributes
 3. Nested lists and their formatting

HTML Image Insertion:

- Inserting Images with the **** Tag
 1. Understanding **src** and **alt** attributes
 2. Image dimensions and alignment

HTML Embedding Videos:

- Embedding Videos Using the **<video>** Tag
 1. Video attributes: **controls**, **autoplay**, **loop**, **muted**
 2. Supporting different video formats

HTML Forms:

- Introduction to Forms
 1. Purpose and structure of HTML forms
 2. Basic form elements and their roles

The screenshot shows a web form titled "Application Form" with a blue header and a light blue body. The form is divided into two main sections: "A. General Questions" and "B. Type of Health Coverage".

Section A: General Questions

- Proposed Insured's Name: (text input)
- Birth Date: (text input with slashes for day/month/year)
- Gender: (radio buttons for Male and Female)
- Address: (text input)
- Phone Number: (text input)
- Email Address: (text input)
- ID Number: (text input)
- Social Security Number: (text input)
- Status: (radio buttons for Single, Married, Divorced, Others)
- Occupation: (text input)
- Are you a retiree?: (radio buttons for Yes and No)

Section B: Type of Health Coverage

- Employee: (radio buttons for Yes and No)
- Spouse: (radio buttons for Yes and No)
- Children: (radio buttons for Yes and No)
- Plan Choice: (text input)

Below these sections is a table for "Complete if Spouse/Children are Proposed for Insurance:"

Name	SSN No.	Relationship to proposed insured	Birth Date	Age	Sex

- HTML Form Tags and Attributes
 1. Essential form tags: `<form>`, `<input>`, `<label>`, `<fieldset>`
 2. Understanding form attributes: `action`, `method`, `name`, `id`
- HTML Forms - Post vs. Get
 1. Differences between `POST` and `GET` methods
 2. Appropriate use cases for each method
- HTML Forms - Input Text Fields
 1. Creating text input fields with `<input type="text">`
 2. Attributes for text fields: `placeholder`, `maxlength`, `required`
- HTML Forms - Select Menus
 1. Creating drop-down lists with `<select>` and `<option>` tags
 2. Handling multi-select menus
- HTML Forms - Checkboxes and Radio Buttons
 1. Using `<input type="checkbox">` and `<input type="radio">`
 2. Grouping radio buttons for exclusive selection
- HTML Forms - Text Areas and Buttons
 1. Creating multi-line text input with `<textarea>`
 2. Adding buttons: `<button>`, `<input type="submit">`, `<input type="reset">`

HTML iframes:

- Embedding External Content with the `<iframe>` Tag
 1. Adjusting iframe attributes: `src`, `width`, `height`, `frameborder`
 2. Security and sandboxing considerations

CSS : MODULE 1

Introduction:

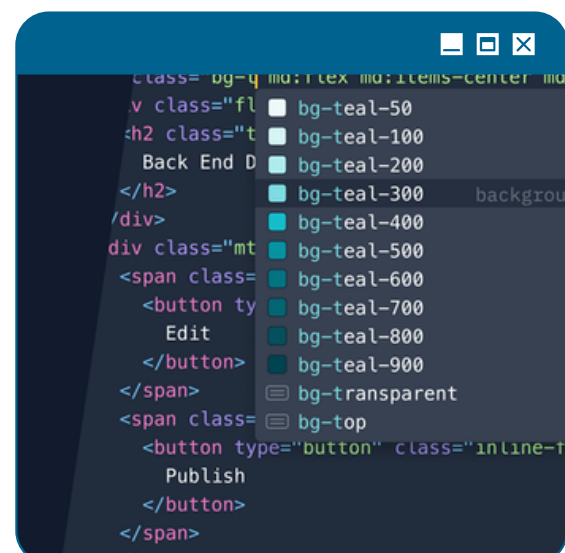
- What is CSS?
- Why use CSS?
- CSS syntax and structure

Types of CSS:

- Inline CSS
- Internal/Embedded CSS
- External CSS

CSS Selectors:

- Element selectors
- Class selectors
- ID selectors
- Attribute selectors
- Grouping selectors



CSS : MODULE 2

CSS Box Model:

- Understanding the box model (margin, border, padding, content)
- Box-sizing property

Layout Techniques:

- Block vs. inline elements
- Display property (block, inline, inline-block, flex, grid)
- Positioning (static, relative, absolute, fixed)

Typography:

- Fonts and text properties
- Text alignment
- Text decoration and transformation

Typography:

- Fonts and text properties
- Text alignment
- Text decoration and transformation

CSS : MODULE 3

Colors and Backgrounds:

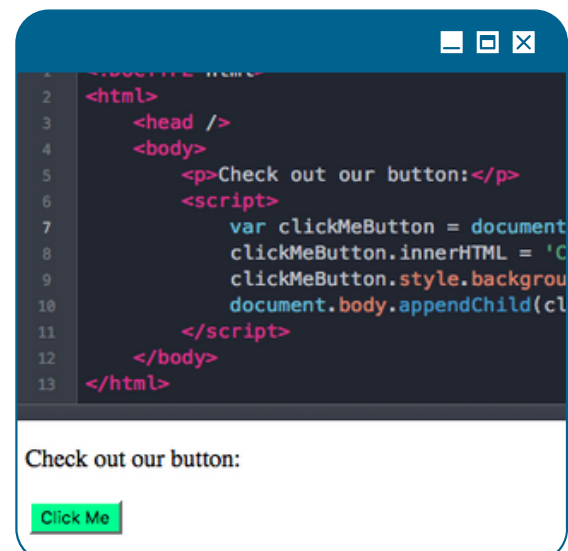
- Color properties (color, background-color)
- Using gradients
- Background images and positioning

Responsive Design:

- Media queries
- Viewport units (vw, vh)
- Responsive units (em, rem)

CSS Transitions and Animations:

- Transition properties (transition, transition-property, etc.)
- Keyframes and animation properties
- Creating basic animations



BOOTSTRAP : MODULE 1

Foundations of Bootstrap:

- **Introduction to Bootstrap:** Overview of Bootstrap and its importance in web development.
- **Embedding Bootstrap:** Methods to include Bootstrap in web projects (CDN, local files).
- **Bootstrap: Basic Page Structure:** Setting up the basic HTML structure for a Bootstrap project.

Layout and Grid System:

- **Bootstrap Grid System:** Understanding the grid system, containers, rows, and columns.
- **Bootstrap Three Column Layouts:** Creating responsive three-column layouts.

Typography and Tables:

- **Bootstrap Typography:** Styling text using Bootstrap classes.
- **Bootstrap Tables:** Creating and styling tables.

BOOTSTRAP : MODULE 2

Images and Utilities:

- **Bootstrap Styling Images:** Enhancing images with Bootstrap classes.
- **Bootstrap Jumbotron:** Creating attention-grabbing content blocks.

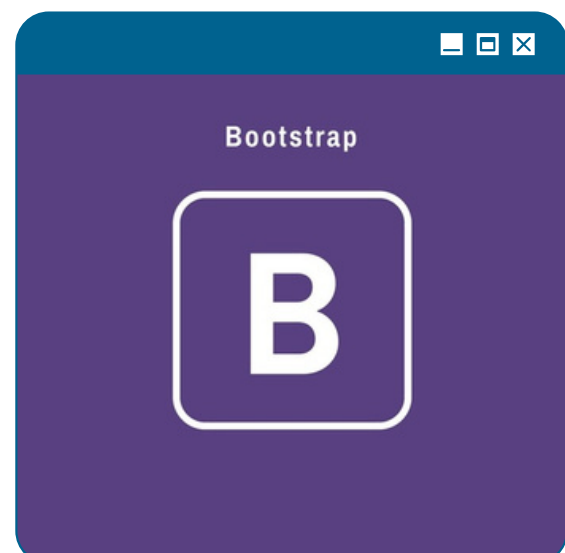
Alerts and Buttons:

- **Bootstrap Alerts:** Implementing alert messages.
- **Bootstrap Buttons:** Styling and customizing buttons.
- **Bootstrap Button Groups and Justified Button Groups:** Grouping and justifying buttons.

BOOTSTRAP : MODULE 3

Navigation Components:

- **Bootstrap Pagination and Pager Pagination:** Implementing pagination controls.
- **Bootstrap List Groups:** Creating list groups for content.



- **Bootstrap Dropdown Menus:** Creating dropdown menus.

Tabs and Menus:

- **Bootstrap Tab Menus and Pill Menus:** Implementing tabbed navigation.
- **Bootstrap Dynamic Tabs and Pills:** Creating dynamic tabs and pills.
- **Bootstrap Dropdown Menus:** Creating dropdown menus.

Forms and Inputs:

- **Bootstrap Forms:** Vertical and Inline: Designing forms using Bootstrap.
- **Bootstrap Inputs: Customizing form inputs.**
- **Bootstrap Form Control States:** Managing input states (valid, invalid, disabled).
- **Bootstrap Input Sizing:** Adjusting the size of form inputs.

JAVASCRIPT : MODULE 1

Introduction: In this module we will learn about what JavaScript is and the benefits of the language. We can also learn how to write the first JavaScript program.

- JavaScript Introduction
- First Javascript program
- Benefits of Javascript

JAVASCRIPT : MODULE 2

Language Syntax: In this module we will learn about JavaScript language syntax and how to use the program level. And we can also learn to understand the arrays.

- Variable declaration
- Operators
- Control Statements
- Error Handling
- Understanding arrays
- Function Declaration

```

16 const LOCALE = globalThis.navigator.language
17
18 const div = document.body.appendChild(document.createElement('div'))
19 const list = div.appendChild(document.createElement('ol'))
20
21 const dayNames = new Map()
22
23 for (let i = 0; i < 7; ++i) {
24   const d = Temporal.PlainDate.from({
25     year: Temporal.Now.plainDateISO().year,
26     month: 1,
27     day: i + 1,
28   })
29   dayNames.set(d.dayOfWeek, d.toLocaleString(LOCALE, {
30   })
31 }
32
33 for (const num of [...dayNames.keys()].sort((a, b) => a < b)) {
34   list.appendChild(Object.assign(
35     document.createElement('li'),
36     {.textContent: dayNames.get(num)}),
37   ))
38 }

```


JAVASCRIPT : MODULE 3

HTML Forms: In this module we will learn about how to use the forms. What are the properties in JavaScript? HTML form validation can be done by JavaScript.

- HTML Document object Model
- Working with HTML form and its elements

JAVASCRIPT : MODULE 4

HTML DOM: In this module we will learn about how JavaScript can access and change all the elements of an HTML document.

- HTML Document object Model
- Working with HTML form and its elements

INTRODUCTION TO AJAX AND BASIC REQUESTS : MODULE 1

Introduction to AJAX:

- Understanding asynchronous communication in web development.
- Benefits of AJAX for enhancing user experience.

Making Asynchronous Requests:

- Using JavaScript to initiate basic AJAX requests (GET method).
- Handling simple data retrieval from external sources.

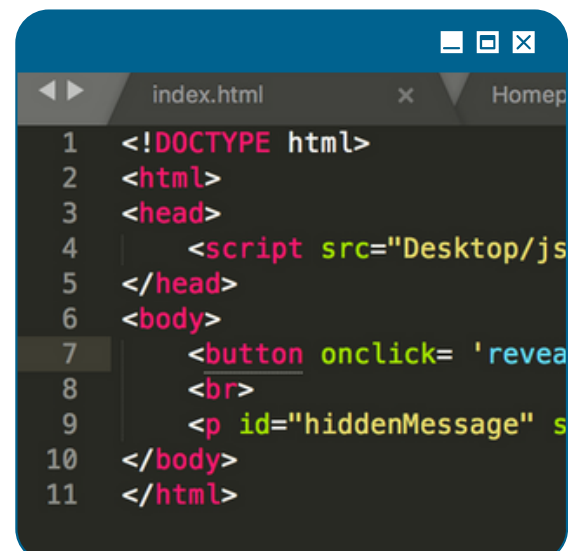
Handling AJAX Responses:

- Processing and displaying JSON data received from external APIs.
- Basic error handling for AJAX requests.

INTERMEDIATE AJAX TECHNIQUES : MODULE 2

Exploring HTTP Methods:

- Implementing POST requests for data submission.
- Understanding and using other HTTP methods (PUT, DELETE) as needed.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <script src="Desktop/js"
5 </head>
6 <body>
7     <button onclick= 'revea
8     <br>
9     <p id="hiddenMessage" s
10 </body>
11 </html>
```

DOM Manipulation with AJAX:

- Updating specific parts of a web page dynamically using AJAX responses.
- Manipulating document elements based on AJAX data updates.

AJAX AND EXTERNAL APIS INTEGRATION : MODULE 3**Security Considerations:**

- Addressing CORS issues and implementing secure AJAX.
- Mitigating XSS and CSRF vulnerabilities in AJAX implementations.

.NET INTRODUCTION AND ENVIRONMENT SETUP**Getting Started**

- Introduction
- MVC Architecture Pattern
- Setting up the development Environment
- Your First ASP.NET MVC App
- MVC in Action
- Adding a Theme

ASP.NET MVC Fundamentals

- Introduction
- Action Results
- Action Parameters
- Convention-based Routing
- Attribute Routing
- Passing Data to Views
- View Models
- Razor Syntax
- Partial Views
- Cheat Sheet
- Exercise
- Solution
- Summary

ASP.NET MVC Fundamentals

- Introduction
- Entity Framework
- Database-first vs Code-first
- Code-first Migrations
- Changing the Model

```
var xhr = new XMLHttpRequest();
xhr.onreadystatechange = function() {
    if (xhr.readyState === 4) {
        if (xhr.status === 200) {
            // any response code
        }
    }
}
xhr.open(HttpMethod, url, async);
xhr.send(string);
```

Application Form

A. General Questions

Proposed Insured's Name:

(Please use capital letters)

Birth Date: / / Gender: ☐ Male ☐ Female

Address:

Phone Number: Email Address:

ID Number: Social Security Number:

Status: ☐ Single ☐ Married ☐ Divorced ☐ Others

Occupation: Are you a retiree? ☐ Yes ☐ No

B. Type of Health Coverage

Employee: ☐ Yes ☐ No
Plan Choice:

Spouse: ☐ Yes ☐ No
Plan Choice:

Children: ☐ Yes ☐ No
Plan Choice:

Complete if Spouse/Children are Proposed for Insurance:

Name	SSN No.	Relationship to proposed insured	Birth Date	Age	Sex

Building RESTful Services with .ASP .NET Web API

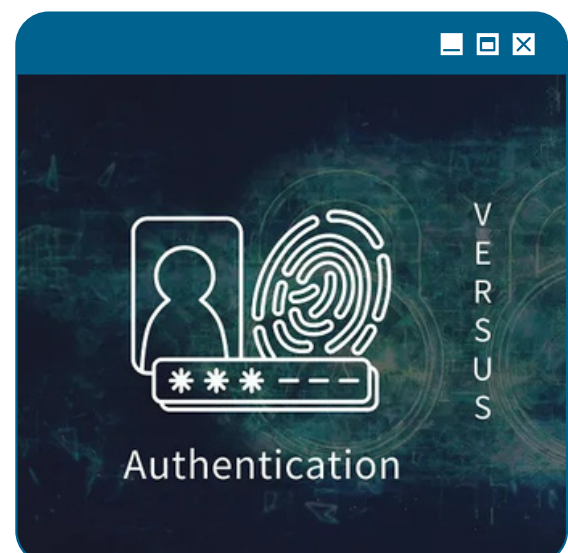
- Introduction
- What is a Web API?
- RESTful Convention
- Building an API
- Testing an API
- Data Transfer Objects
- Auto Mapper
- Using Camel Notation
- IHttpActionResult
- Cheat Sheet
- Exercise
- Summary

Client Side Development

- Introduction
- Calling an API Using jQuery
- Bootbox Plug-in
- Optimizing jQuery Code
- DataTables Plug-in
- DataTables with Ajax Source
- Returning Hierarchical Data
- DataTables: Removing Records
- DataTables: Discussion Points
- Single Page Applications (SPAs)
- Cheat Sheet
- Exercise
- Summary

Authentication and Authorization

- Introduction
- The Problem
- Authentication Options
- ASP.NET Identity
- Restricting Access
- Seeding Users and Roles
- Working with Roles
- Adding Profile Data
- OAuth
- Social Logins
- Cheat Sheet
- Exercise
- Summary



Performance Optimization

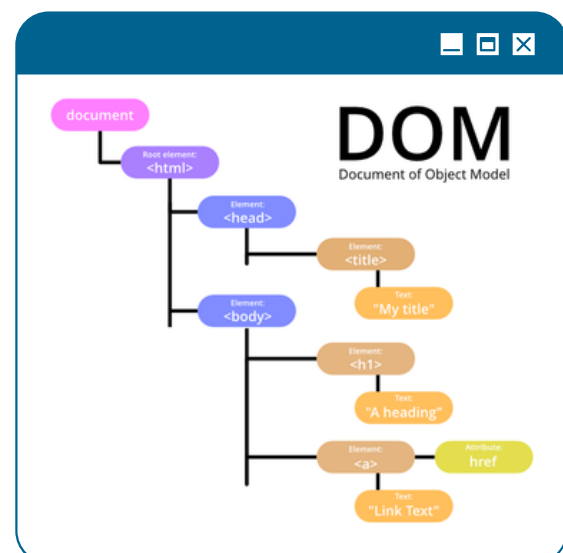
- Introduction
- Overview
- Data Tier
- Glimpse
- Output Cache
- Data Cache
- Async
- Release Builds
- Disabling Session
- Client Tier
- Chrome Developer Tools Tip
- Cheat Sheet
- Summary

Building a Feature End-to-End Systematically

- Introduction
- Understanding the Problem
- Domain Modelling
- Building the Simplest API
- Adding the Details
- Edge Cases
- Building the Front-end
- Adding Auto-completion
- Updating the DOM
- Improving the Look and Feel
- Filtering Records
- Submitting the Form
- Displaying Toast Notifications
- Implementing Client-side Validation
- Code Review
- Summary

Deployment

- Introduction
- Deploying the Application
- Deploying the Database
- Build Configurations
- Application Settings
- Securing Configuration Settings
- Custom Error Pages
- Logging Unhandled Exceptions
- Final Touch



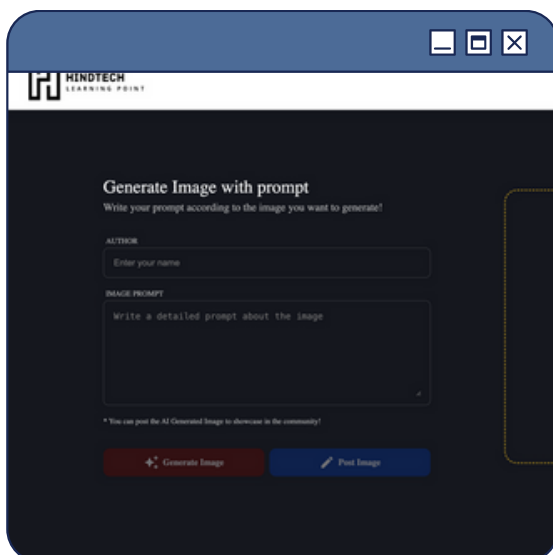
PROJECT DEVELOPMENT

1. **AI Chat & Gen**
2. **Tech News Blog with Admin**

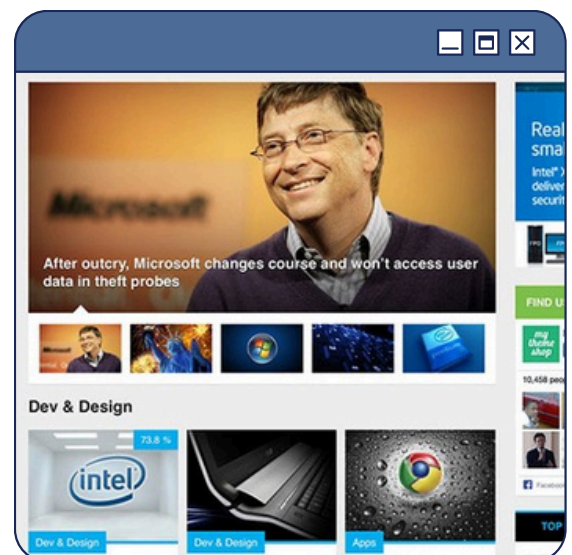
DURATION : 180 DAYS

OUTCOME:

- **Understanding HTML:** Gain foundational knowledge about HTML, including its importance, structure, and how it integrates with CSS and JavaScript.
- **HTML Document Structure:** Comprehend the basic structure of an HTML document, including the `<!DOCTYPE html>` declaration and the role of head and body tags.
- **CSS Introduction:** Learn the basics of CSS, including syntax, types of CSS (inline, internal, external), and various selectors.
- **Bootstrap Forms and Inputs:** Gain skills to design and customize forms using Bootstrap, including form control states and input sizing.
- **Responsive Design:** Gain knowledge of responsive design using media queries, viewport units, and responsive units like em and rem.
- **JavaScript Introduction:** Understand the basics of JavaScript, its benefits, and write the first JavaScript program.



PROJECT : 1



PROJECT : 2



Hindtech Learning Point

A Trusted IT Institute Brand In Lucknow

Admission Open For

Summer Training

CSE/IT

Diploma, B.Tech/B.E, BCA, MCA

45 - 50 Days Program

Apprenticeship Training

CSE/IT

B.Tech, BCA, MCA, PGDCA, Dip.

6 to 8 Months Program

Industrial Training

CSE/IT

B.Tech, BCA, MCA, PGDCA, Dip.

3 to 5 Months Program

UI/UX Designing

CSE/IT or interested student

B.Tech, BCA, MCA, PGDCA, Dip.

45 - 50 Days Program

1
Number
ONE

Excellent Performance
by Hindtech Learning Point
in IT Training



Since 2020, 1000+ students placed in top IT firms | Parmar Plaza, BKT, Bargadi, UP 226201 | Call: 6307738600 / 7905320279 | www.hindtechlearningpoint.com



HINDTECH
IT SOLUTIONS

Training

.Net Apprenticeship Training Program

Duration

6 to 8 Months

Amount

₹21,000



www.hindtechitsolutions.com



hr@hindtechitsolutions.com



+91 7905320279



www.hindtechlearningpoint.com



hindtechlearningpointlko@gmail.com



+91 6307738600



Building No 10/703, Ground Floor, near Arvindo Park Road, Sector 10, Indira Nagar, Lucknow, Uttar Pradesh 226016

Branch 1



PARMAR PLAZA, 1st Floor, Bakshi Ka Talab, Bargadi Magath, Uttar Pradesh 226201

Branch 2

Scan Me:



Scan this QR code for details



HINDTECH
LEARNING POINT

