Parul® University

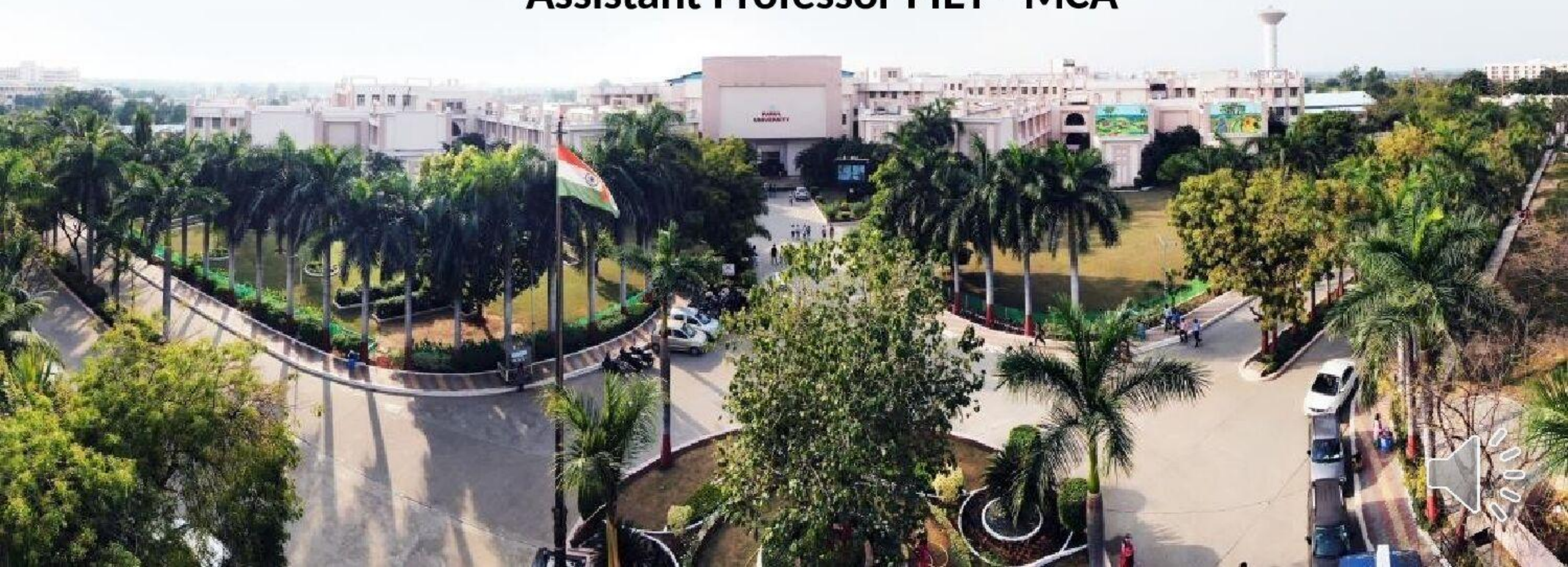
05201302

Web Application Development

MCA MSc-IT Semester 3

Prof Faruk Abdulla

Assistant Professor PIET - MCA



Parul®
University

DIGITAL LEARNING CONTENT

Unit 2: ASP.NET Server Controls

ASP.NET Server Controls are special UI components provided by the ASP.NET framework that run on the web server rather than the client browser. Unlike plain HTML elements, these controls encapsulate both the visual interface and the underlying server-side behaviour.

When a page with server controls is requested by a client, these controls execute on the server to perform tasks such as rendering HTML, maintaining state, and processing user inputs before sending the final HTML to the browser.

Parul®
University

DIGITAL LEARNING CONTENT

Unit 2: ASP.NET Server Controls

Server controls are components that run on the server and generate HTML markup to be sent to the client browser. Unlike basic HTML controls, server controls provide built-in functionality such as state management, event handling, and easier data binding

They simplify web application development by abstracting HTML, JavaScript, and HTTP protocols. Developers can focus on application logic and UI rather than low-level web details.

Parul®
University

DIGITAL LEARNING CONTENT

Unit 2: ASP.NET Server Controls

- Server controls are special ASP.NET components that run on the server and render HTML to browsers.
- They provide a higher-level abstraction over raw HTML elements.
- Automatically handle client-server communication, state management, and event handling.
- Enable developers to build dynamic, interactive web pages easily.

Parul®
University

DIGITAL LEARNING CONTENT

Unit 2: ASP.NET Server Controls

•Server controls bridge the gap between traditional web programming (where HTML is static and developers manually handle client-server communication) and modern interactive web applications that require dynamic content, complex input validation, and event-driven interactions — all while simplifying the developer's job.

Parul®
University

DIGITAL LEARNING CONTENT

Why use Server Controls?

- Automate HTML rendering
- Manage state automatically between postbacks
- Handle server-side events
- Provide easy data-binding capabilities

Parul®
University

DIGITAL LEARNING CONTENT

Characteristics of Server Controls

- **Run on Server:** They execute server-side, so you can use full .NET capabilities like data access, authentication, and complex logic.
- **Event-Driven Model:** Server controls raise events such as Click or SelectedIndexChanged, allowing developers to write familiar event handlers (similar to desktop apps).
- **State Management:** Unlike HTML, server controls automatically maintain their values across postbacks (page reloads) using ViewState or other techniques.
- **Rich Functionality:** Many built-in controls come with advanced features like data binding, templating, and validation out of the box.
- **Extensible:** Developers can create custom server controls to encapsulate reusable UI and logic.

Parul®
University

Types of Server Controls

- **Standard Controls:** Buttons, TextBoxes, Labels, CheckBoxes, RadioButtons, DropDownLists, ListBoxes, etc.
- **Data Controls:** GridView, DetailsView, FormView, Repeater controls for displaying and manipulating data.
- **Validation Controls:** RequiredFieldValidator, RangeValidator, RegularExpressionValidator, CustomValidator, ValidationSummary.
- **Navigation Controls**: Menu, TreeView, SiteMapPath.
- **Login Controls:** Login, LoginView, LoginStatus, etc.

Life Without Server Controls (For Comparison)

If you only used plain HTML and scripts, you'd need to:

- Manually handle form submission and data extraction on the server.
- Write JavaScript for client-side validation and UI updates.
- Manage state (form values) manually across page reloads.
- Write extensive boilerplate code for common UI features.

Server controls remove much of this burden and allow focus on business logic instead of plumbing.

Parul®
University

DIGITAL LEARNING CONTENT

Overview of Control Categories

Control Type	Description	Use Case
Standard Controls	Basic input and display controls	Buttons, TextBoxes, Labels
Validation Controls	Validate user input	RequiredField, Range, Regex
Data Controls	Display and manipulate data	GridView, ListView, Repeater
Navigation Controls	Site navigation aids	Menu, SiteMapPath
Web Server Controls	Custom or advanced controls	Calendar, AdRotator

Parul®
University

DIGITAL LEARNING CONTENT

Standard Controls

Standard controls form the building blocks for web forms and allow users to interact with the application. These controls abstract HTML input elements but add capabilities like server-side events and state persistence.

Parul®
University

DIGITAL LEARNING CONTENT

Standard Controls

Standard controls form the building blocks for web forms and allow users to interact with the application. These controls abstract HTML input elements but add capabilities like server-side events and state persistence.

Parul®
University

DIGITAL LEARNING CONTENT

Standard Controls

Control	Description	Example & Code Snippet
Label	Displays text or messages	<asp:Label ID="lblMsg" runat="server" Text="Welcome"></asp:Label>
TextBox	User input	<asp:TextBox ID="txtName" runat="server"></asp:TextBox>
Button	User action triggers event	<asp:Button ID="btnSubmit" runat="server" Text="Submit" OnClick="btnSubmit_Click" />
CheckBox	Boolean choice (checked/unchecked)	<asp:CheckBox ID="chkAgree" runat="server" Text="I Agree"/>
RadioButton	Select one option from group	<asp:RadioButton ID="rbMale" runat="server" GroupName="Gender" Text="Male"/>
DropDownList	Select from dropdown list	<asp:DropDownList ID="ddlCity" runat="server">...</asp:DropDownList>

Parul®
University

DIGITAL LEARNING CONTENT

How Server Controls Render HTML

Server controls generate standard HTML elements dynamically.

For example, an <asp:Button> control renders as an <input type="submit"> or <button> HTML element in the browser, but with additional JavaScript or hidden fields to handle postbacks and events.

This means the developer writes code at a higher abstraction level without worrying about the exact HTML details, making development faster and less error-prone.

Parul®
University

DIGITAL LEARNING CONTENT

How Server Controls Render HTML

Server controls generate standard HTML elements dynamically.

For example, an <asp:Button> control renders as an <input type="submit"> or <button> HTML element in the browser, but with additional JavaScript or hidden fields to handle postbacks and events.

This means the developer writes code at a higher abstraction level without worrying about the exact HTML details, making development faster and less error-prone.

Parul®
University

DIGITAL LEARNING CONTENT

Server Controls

Standard server controls are built-in controls provided by ASP.NET to create web forms and handle input/output. These controls reside on the server and interact with the client browser by rendering HTML dynamically.

Parul®
University

DIGITAL LEARNING CONTENT

TextBox Control

The TextBox control allows the user to enter text input.

TextModes:

- **SingleLine** – For basic input (e.g., names).
- **MultiLine** – For comments or descriptions.
- **Password** – Masks input for security.

Properties: Text, TextMode, MaxLength, ReadOnly, Columns, Rows

Example

```
<asp:TextBox ID="txtName" runat="server" TextMode="SingleLine" />
```

Parul®
University

DIGITAL LEARNING CONTENT

Label Control

Used to display static or dynamic text on the web page.

Properties: Text, ForeColor, Font, Visible, CssClass

Example

```
<asp:Label ID="lblWelcome" runat="server" Text="Welcome!" />
```

Parul®
University

DIGITAL LEARNING CONTENT

Label Control

Used to display static or dynamic text on the web page.

Properties: Text, ForeColor, Font, Visible, CssClass

Example

```
<asp:Label ID="lblWelcome" runat="server" Text="Welcome!" />
```

Parul®
University

DIGITAL LEARNING CONTENT

Validation Controls

Validation Controls Ensure user inputs meet requirements before processing.
Support client-side and server-side validation.
Types include:

RequiredFieldValidator — Field must not be empty.
RangeValidator — Value must be within a specified range.
RegularExpressionValidator — Match input against a regex pattern.
CompareValidator — Compare values of two controls.
CustomValidator — Custom code for validation logic.
ValidationSummary — Displays all validation errors in one place.

Parul®
University

DIGITAL LEARNING CONTENT

Validation Controls

Validation Controls Ensure user inputs meet requirements before processing.
Support client-side and server-side validation.
Types include:

RequiredFieldValidator — Field must not be empty.
RangeValidator — Value must be within a specified range.
RegularExpressionValidator — Match input against a regex pattern.
CompareValidator — Compare values of two controls.
CustomValidator — Custom code for validation logic.
ValidationSummary — Displays all validation errors in one place.

Parul®
University

DIGITAL LEARNING CONTENT

Why Validation Controls?

- To prevent invalid data submission.
- Reduce server errors by catching mistakes early.
- Improve user experience with immediate feedback.

Parul®
University

DIGITAL LEARNING CONTENT

Validation Controls

Control	Purpose
RequiredFieldValidator	Ensures control is not empty
RangeValidator	Validates input range (e.g., 1-100)
RegularExpressionValidator	Matches input to a pattern (email, phone)
CompareValidator	Compares values of two controls
CustomValidator	Custom validation logic
ValidationSummary	Displays all validation messages together

Parul®
University

DIGITAL LEARNING CONTENT

ASP.NET Page Lifecycle

- **Page_Init:** Initialization of controls.
- **Page_Load:** Business logic and control setup.
- **PostBack Events:** Respond to user actions like button clicks.
- **Page_PreRender:** Last chance to modify controls before rendering.
- **Page_Unload:** Cleanup code.

Understanding lifecycle helps in managing state and event handling correctly.