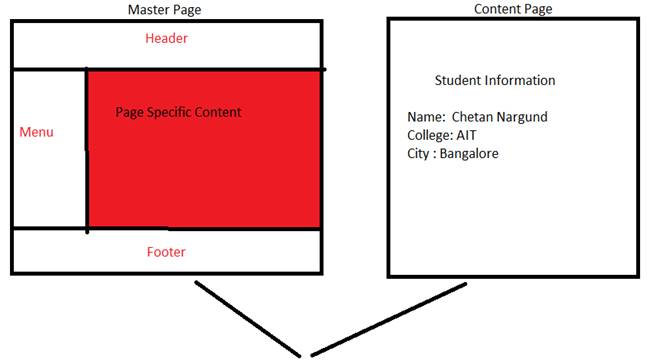
* **Chapter 5: MASTER PAGE AND THEME**
* **What is Master Page?**
* Master Page is the feature that enable you to **define common structure** and interface mark up element for your website including header or footer.
* The master page can be shared by any of the pages in your website which are known as content pages and removes the need to duplicate code for the share element within your website.
* **Advantages:**
  + Centralized common functionality
  + Make easy to create one set of code and control and apply to all set of pages
  + Provides the object model to customize the master page to individual content page
* **Steps to create Master Page:**
  + Go to add new item
  + Select Master Page
  + Then name the master page
  + We will now add the menu bar on this master page which will be common to all pages
  + Once we have added the menu bar we can have the content pages to use the master page.
  + When we add the content pages we have to select the option of master page



* **Requirement of Master page in Asp.Net:**
* ASP.NET master pages allow you to create a consistent layout for the pages in your application.
* A single master page defines the look and feel and standard behavior that you want for all of the pages (or a group of pages) in your application.
* You can then create individual content pages that contain the content you want to display.
* When users request the content pages, they merge with the master page to produce output that combines the layout of the master page with the content from the content page.
* When in the case you require to give same layout to all the pages at that master page is required which defines look and feel and standard behavior.
* They allow you to centralize the common functionality of your pages so that you can make updates in just one place.
* They make it easy to create one set of controls and code and apply the results to a set of pages. For example, you can use controls on the master page to create a menu that applies to all pages.
* It provides the object model to customize the master page to individual content page.
* **Themes:**
* A theme is a collection of property settings that allow you to define the look of pages and controls, and then apply the look consistently across pages in a Web application, across an entire Web application, or across all Web applications on a server.
* Themes are made up of a set of elements: skins, cascading style sheets (CSS), images, and other resources. At a minimum, a theme will contain skins. Themes are defined in special directories in your Web site or on your Web server.
* **Skins**
* A skin file has the file name extension .skin and contains property settings for individual controls such as Button, Label, TextBox, or Calendar controls. Control skin settings are like the control markup itself, but contain only the properties you want to set as part of the theme.
* For example, the following is a control skin for a Button control:
* <asp:button runat="server" BackColor="lightblue" ForeColor="black" />
* You create .skin files in the Theme folder. A .skin file can contain one or more control skins for one or more control types. You can define skins in a separate file for each control or define all the skins for a theme in a single file.