Q. 1	**Complete the following statement.** **_____ Can follow a Test Driven Development approach.**
Ans:	ASP.NET MVC
Q. 2	**Which part of MVC is responsible for rendering UI(HTML) ?**
Ans:	View
Q. 3	**What is used in MVC to perform validations like, Client Validations and Remote Validation?**
Ans:	Data Annotations.
Q. 4	**_____ responds to user input, talks to the Model, and it decides which view to render.**
Ans:	Controller
Q. 5	**Which methods under controller are accessible via URL?**
Ans:	Public methods (also known as actions)
Q. 6	**What is the use of Content folder created under MVC Application?**
Ans:	It is used to store static files like CSS, images, etc.
Q. 7	**List out any 4 important features introduced in ASP.NET MVC-5?**
Ans:	ASP.NET Identity One ASP.NET Bootstrap Attribute Routing
Q. 8	**What is the method which is used to place the content of the controller action's primary view?**
Ans:	RenderBody method
Q. 9	**Which are 2 type of views engines in ASP.NET MVC which lets us to embed server-based code(C# and VB.NET) into application?**
Ans:	ASPX view engine Razor view engine
Q. 10	**What is the correct Razor syntax used for multiline comment?**

Ans:	@*
	   Line one
	   Line two
	   Line N
	*@

Q. 11	**What is file in MVC which is similar to Master Pages from WebForms & where it is located in application?**
Ans:	Layout.cshtml is file which works as same as master page which is located under the Shared folder in MVC application.
Q. 12	**What is the purpose of @RenderSection() ?**
Ans:	It is used to render the content of a named section of content page inside the master page.
Q. 13	**Which are 2 different ways by which we can render the partial view within another view?**

Ans:	@Html.Partial("pvSample")
	
	@{
	   Html.RenderPartial("pvSample");
	}

Q. 14	**Complete the following statement.** **Every request that comes to your application is handled by the _____.**
Ans:	Controller

Q. 15	**State whether the following statement is true or false.**
	    1) **ASP.NET MVC implements the front View variant of the MVC pattern.**
	    2) **Controller sits in front of everything except the routing subsystem**

Ans:	Statements 1 is false & statement 2 is true
Q. 16	**Who does the job to match the incoming request and maps the request to a controller's action?**
Ans:	Routing System
Q. 17	**Is it possible to use routing system by ASP.Net Web Forms as well?**
Ans:	Yes, Possible. Because the routing system classes are in the System.Web. Assembly and not in System.Web.Mvc.
Q. 18	**Where & when the routing table is created?**
Ans:	A route table is created in the application's Global.asax file during the Application Start event.
Q. 19	**Where we used to define action methods?**
Ans:	We define action methods inside controller class.
Q. 20	**List down the benefits of MVC.**
Ans:	Multiple view support Separation of Concerns Testability Lightweight
Q. 21	**List out different return types of a controller action method?**
Ans:	ViewResult (View): Returns a webpage from an action method. PartialviewResult (Partialview): Sends a part of a view which will be rendered in other view. RedirectResult (Redirect): Redirect to any other controller & action method. RedirectToRouteResult: Used when we want to redirect to any other action method. ContentResult (Content): Used to return HTTP content type like text/plain. jsonResult (json): Used when we want to return a JSON message. javascriptResult (javascript): Used to return JavaScript code that will run in browser. FileResult (File): Used to send binary output in response. EmptyResult: Used to return nothing (void) in the result.
Q. 22	**What are Action Filters in MVC?**
Ans:	Action Filters are additional attributes that can be applied to either a controller section or the entire controller to modify the way in which action is executed. ASP.NET MVC provides the following action filters: **Output Cache**: Caches the output of a controller action for a specified amount of time. **Handle Error**: Handles errors raised when a controller action executes. **Authorize**: Enables you to restrict access to a particular user or role.
Q. 23	**What are the three segments for routing?**
Ans:	ControllerName, ActionMethodName & Parammeter.

Q. 24	What is the important difference between ViewData and ViewBag?
Ans:	ViewData derived from ViewDataDictionary class vs ViewBag is a dynamic property that takes advantage of the new dynamic features in C# 4.0. ViewData requires typecasting for complex data type vs ViewBag doesn't require typecasting for complex data type. ViewData uses array notation vs ViewBag uses object notation.
Q. 25	What is Partial View in MVC?
Ans:	Partial view is a reusable view (like a user control) which can be embedded inside other view.
Q. 26	What are HTML helpers in MVC?
Ans:	With MVC, HTML helpers are much like traditional ASP.NET Web Form controls. Just like web form controls in ASP.NET, HTML helpers are used to modify HTML. But HTML helpers are more lightweight. Unlike Web Form controls, an HTML helper does not have an event model and a view state. In most cases, an HTML helper is just a method that returns a string.
Q. 27	Explain attribute based routing in MVC?

Ans:	ASP.NET MVC 5.0 we have a new attribute route. By using the "Route" attribute we can define the URL structure. For example in the below code we have decorated the "GotoAbout" action with the route attribute. The route attribute says that the "GotoAbout" can be invoked using the URL structure "Users/about".
	
	 `public class HomeController: Controller`
	 `{`
	   `[Route("Users/about")]`
	   `publicActionResultGotoAbout()`
	   `{`
	     `return View();`
	   `}`
	 `}`

Q. 28	Explain the concept of MVC Scaffolding?
Ans:	ASP.NET Scaffolding is a code generation framework for ASP.NET Web applications. Visual Studio 2013 includes pre-installed code generators for MVC and Web API projects. Scaffolding consists of page templates, entity page templates, field page templates, and filter templates. These templates are called Scaffold templates and allow you to quickly build a functional data-driven Website

Q. 29	**What is Bundling and Minification in MVC?**
Ans:	Bundling and minification are two new techniques introduced to improve request load time. It improves load time by reducing the number of requests to the server and reducing the size of requested assets (such as CSS and JavaScript). Bundling: It lets us combine multiple JavaScript (.js) files or multiple cascading style sheet (.css) files so that they can be downloaded as a unit, rather than making individual HTTP requests. Minification: It squeezes out whitespace and performs other types of compression to make the downloaded files as small as possible. At runtime, the process identifies the user agent, for example IE, Mozilla, etc. and then removes whatever is specific to Mozilla when the request comes from IE.
Q. 30	**What is different Data Annotation used to Validate Model in MVC? Explain few important with an example.**
Ans:	Required: Specify a property as required. Ex. [Required(ErrorMessage="CustomerName is mandatory")] RegularExpression: Specifies the regular expression to validate the value of the property. Ex. [RegularExpression("[a-z]", ErrorMessage = "Invalid character")] Range: Specifies the Range of values between which the property values are checked. Ex. [Range(1000,10000,ErrorMessage="Range should be between 1k & 10k")] StringLength: Specifies the Min & Max length for a string property. Ex. [StringLength(50, MinimumLength = 5, ErrorMessage = "Minimum char is 5 and maximum char is 10")] MaxLength: Specifies the Max length for the property value. Ex. [MaxLength(10,ErrorMessage="Customer Code is exceeding")] MinLength: It is used to check for minimum length. Ex. [MinLength(5, ErrorMessage = "Customer Code is too small")]