**Accessing Data (24%)**

* Choose data access technologies
  + Choose a technology (ADO.NET, Entity Framework, WCF Data Services) based on application requirements
* Implement caching
  + Cache static data, apply cache policy (including expirations)
    - <http://msdn.microsoft.com/en-us/library/ff477235.aspx>
  + Use CacheDependency to refresh cache data
    - <http://www.dotnetcurry.com/ShowArticle.aspx?ID=331>
  + Query notifications
    - <http://msdn.microsoft.com/en-us/library/t9x04ed2(v=vs.80).aspx>
    - <http://msdn.microsoft.com/en-us/library/system.web.caching.sqlcachedependency.aspx>
    - <http://msdn.microsoft.com/en-us/library/9dz445ks(v=vs.110).aspx>
    - <http://msdn.microsoft.com/en-US/library/wd2x83zk(v=vs.80).aspx>
* Implement transactions
  + Manage transactions by using the API from System.Transactions namespace
    - <http://msdn.microsoft.com/en-us/library/a90c30fy.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms172152.aspx>
  + Implement distributed transactions
    - <http://www.c-sharpcorner.com/UploadFile/mosessaur/TransactionScope04142006103850AM/TransactionScope.aspx>
    - <http://msdn.microsoft.com/en-us/library/system.transactions.transactionscopeoption.aspx>
  + Specify transaction isolation level
    - <http://msdn.microsoft.com/en-us/library/system.transactions.isolationlevel.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms173763.aspx>
* Implement data storage in Windows Azure
  + Access data storage in Windows Azure
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/blob-storage/>
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/table-services/>
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/queue-service/>
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/sql-database/>
  + Choose data storage mechanism in Windows Azure (blobs, tables, queues, SQL Database)
    - <http://social.technet.microsoft.com/wiki/contents/articles/1674.data-storage-offerings-on-the-windows-azure-platform.aspx>
  + Distribute data by using the Content delivery network (CDN)
    - <http://www.windowsazure.com/en-us/develop/net/common-tasks/cdn/>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/ff919703.aspx>
  + Handle exceptions by using retries (SQL Database)
    - <http://social.technet.microsoft.com/wiki/contents/articles/4235.retry-logic-for-transient-failures-in-windows-azure-sql-database.aspx>
    - <http://msdn.microsoft.com/en-us/library/microsoft.practices.transientfaulthandling.retrypolicy_members.aspx>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/microsoft.windowsazure.storage.retrypolicies>
  + Manage Windows Azure Caching
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/cache/>
* Create and implement a WCF Data Services service
  + <http://msdn.microsoft.com/en-us/data/odata.aspx>
  + <http://msdn.microsoft.com/en-us/library/cc668794.aspx>
  + <http://msdn.microsoft.com/en-us/library/cc668792.aspx>
  + <http://msdn.microsoft.com/en-us/library/dd728286.aspx>
  + Address resources
    - <http://msdn.microsoft.com/en-us/library/dd728283.aspx>
  + Implement filtering/Create a query expression
    - <http://msdn.microsoft.com/en-us/library/dd673933.aspx>
    - <http://www.odata.org/documentation/uri-conventions>
  + Access payload formats (including JSON)
    - <http://msdn.microsoft.com/en-us/library/dd728282.aspx#sectionSection0>
  + Use data service interceptors and service operators
    - <http://msdn.microsoft.com/en-us/library/dd744842.aspx>
    - <http://msdn.microsoft.com/en-us/library/dd744837.aspx>
    - <http://msdn.microsoft.com/en-us/library/cc668788.aspx>
    - <http://msdn.microsoft.com/en-us/library/dd744841.aspx>
* Manipulate XML data structures
  + Read, filter, create, modify XML data structures
  + Manipulate XML data by using XMLReader, XMLWriter, XMLDocument, XPath, LINQ to XML
    - <http://www.tizag.com/xmlTutorial/xpathtutorial.php>
  + Transform XML by using XSLT transformations
    - <http://www.tizag.com/xmlTutorial/xslttutorial.php>
    - <http://www.w3schools.com/xsl/xsl_w3celementref.asp>

**Querying and Manipulating Data by Using the Entity Framework (20%)**

* Query and manipulate data by using the Entity Framework.
  + Query, update, and delete data by using DbContext
  + Build a query that uses deferred execution
    - <http://msdn.microsoft.com/en-us/library/bb738633.aspx>
  + Implement lazy loading and eager loading
    - <http://msdn.microsoft.com/en-ca/data/jj574232>
  + Create and run compiled queries
    - <http://msdn.microsoft.com/en-us/library/bb896297.aspx>
  + Query data by using Entity SQL
    - <http://msdn.microsoft.com/en-us/library/bb387145.aspx>
    - <http://msdn.microsoft.com/en-us/library/bb738573.aspx>
* Query and manipulate data by using Data Provider for Entity Framework
  + Query and manipulate data by using Connection, DataReader, Command from the System.Data.EntityClient namespace
    - <http://msdn.microsoft.com/en-us/library/bb738533.aspx>
    - <http://msdn.microsoft.com/en-us/library/bb738561.aspx>
  + Perform synchronous and asynchronous operations
    - <http://msdn.microsoft.com/en-us/library/ca56w9se.aspx>
    - <http://msdn.microsoft.com/en-us/library/1a674khd.aspx>
    - <http://msdn.microsoft.com/en-us/library/x76a3b72.aspx>
    - <http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlcommand.endexecutenonquery.aspx>
    - <http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlcommand.endexecutereader.aspx>
    - <http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlcommand.endexecutexmlreader.aspx>
  + Manage transactions (API)
* Query data by using LINQ to Entities.
  + Query data by using LINQ operators (for example, project, skip, aggregate, filter, and join)
  + Log queries
    - <http://msdn.microsoft.com/en-us/magazine/gg490349.aspx>
  + Implement query boundaries (IQueryable vs. IEnumerable)
  + <http://stackoverflow.com/questions/2876616/returning-ienumerablet-vs-iqueryablet>
* Query and manipulate data by using ADO.NET
  + Query and manipulate data by using Connection, DataReader, Command, DataAdapter, DataSet
    - <http://www.codeproject.com/Articles/361579/A-Beginners-Tutorial-for-Understanding-ADO-NET>
    - <http://www.agilechai.com/content/archive/2007/06/09/how-to-use-sqldatareader-plus-source-code.aspx>
  + Perform synchronous and asynchronous operations
    - <http://msdn.microsoft.com/en-us/library/zw97wx20.aspx>
    - <http://msdn.microsoft.com/en-us/library/ca56w9se.aspx>
    - <http://msdn.microsoft.com/en-us/library/1a674khd.aspx>
    - <http://msdn.microsoft.com/en-us/library/x76a3b72.aspx>
  + Manage transactions (API)
* Create an Entity Framework data model. **– Entity Framework and Data Models**
  + Structure the data model using Table per type, table per class, table per hierarchy
    - <http://weblogs.asp.net/manavi/archive/2010/12/24/inheritance-mapping-strategies-with-entity-framework-code-first-ctp5-part-1-table-per-hierarchy-tph.aspx>
    - <http://weblogs.asp.net/manavi/archive/2010/12/28/inheritance-mapping-strategies-with-entity-framework-code-first-ctp5-part-2-table-per-type-tpt.aspx>
    - <http://weblogs.asp.net/manavi/archive/2011/01/03/inheritance-mapping-strategies-with-entity-framework-code-first-ctp5-part-3-table-per-concrete-type-tpc-and-choosing-strategy-guidelines.aspx>
  + Choose and implement an approach to manage a data model (code first vs. model first vs. database first)
    - <http://msdn.microsoft.com/en-ca/data/jj193542>
    - <http://msdn.microsoft.com/en-ca/data/jj200620>
    - <http://msdn.microsoft.com/en-ca/data/jj205424>
    - <http://msdn.microsoft.com/en-ca/data/jj206878>
  + Implement POCO objects
  + Describe a data model by using conceptual schema definitions, storage schema definition, and mapping language (CSDL, SSDL, MSL)
    - <http://msdn.microsoft.com/en-us/data/jj652004>
    - <http://msdn.microsoft.com/en-us/data/jj652016>
    - <http://msdn.microsoft.com/en-us/data/jj652027>

**Designing and Implementing WCF Services (19%)**

* Create a WCF service
  + <http://msdn.microsoft.com/en-us/magazine/cc163447.aspx>
  + Create contracts (service, data, message, callback, and fault)
    - <http://msdn.microsoft.com/en-us/library/ms733070.aspx>
    - <http://msdn.microsoft.com/en-us/library/ee960161.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms733127.aspx>
    - <http://stackoverflow.com/questions/4731987/datacontract-and-servicecontract-difference>
    - <http://msdn.microsoft.com/en-us/library/ee960168.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms730255.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms731064.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms752208.aspx>
  + Implement message inspectors
    - <http://msdn.microsoft.com/en-us/library/aa717047.aspx>
    - <http://blogs.msdn.com/b/carlosfigueira/archive/2011/04/19/wcf-extensibility-message-inspectors.aspx>
  + Implement asynchronous operations in the service
    - <http://msdn.microsoft.com/en-us/library/ms734701.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms731177.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms730059.aspx>
* Configure WCF services by using configuration settings
  + Configure service behaviors
    - <http://msdn.microsoft.com/en-us/library/ms730137.aspx>
    - <http://msdn.microsoft.com/en-us/magazine/cc163302.aspx>
  + Configure service endpoints
    - <http://msdn.microsoft.com/en-us/library/ms734786.aspx>
  + Configure bindings
    - <http://msdn.microsoft.com/en-us/magazine/cc163394.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms734662.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms733051.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms731144.aspx>
  + Specify a service contract
  + Expose service metadata (XSDs, WSDL and metadata exchange endpoint)
    - <http://msdn.microsoft.com/en-us/library/ms734765.aspx>
* Configure WCF services by using the API
  + Configure service behaviors
    - <http://msdn.microsoft.com/en-us/library/ms730137.aspx>
    - <http://msdn.microsoft.com/en-us/library/aa702788.aspx>
  + Configure service endpoints
    - <http://msdn.microsoft.com/en-us/library/ms731080.aspx>
  + Configure bindings
    - <http://msdn.microsoft.com/en-us/library/ms731833.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms731862.aspx>
  + Specify a service contract
  + Expose service metadata (XSDs, WSDL and metadata exchange endpoint)
    - <http://msdn.microsoft.com/en-us/library/aa738489.aspx>
  + WCF routing and discovery features
    - <http://msdn.microsoft.com/en-us/library/ee517422.aspx>
    - <http://weblogs.asp.net/gsusx/archive/2009/02/13/using-ws-discovery-in-wcf-4-0.aspx>
    - <http://msdn.microsoft.com/en-us/library/dd456791.aspx>
* Secure a WCF service
  + Implement message level security
    - <http://msdn.microsoft.com/en-us/library/ms733137.aspx>
    - <http://msdn.microsoft.com/en-us/library/ff648863.aspx>
  + Implement transport level security
    - <http://msdn.microsoft.com/en-us/library/ms729700.aspx>
  + Implement certificates
    - <http://msdn.microsoft.com/en-us/library/ms731899.aspx>
    - <http://msdn.microsoft.com/en-us/library/ff648360.aspx>
    - <http://msdn.microsoft.com/en-us/library/ff650785.aspx>
* Consume WCF services
  + Generate proxies by using SvcUtil;
  + Generate proxies by creating a service reference;
  + Create and implement channel factories
    - <http://msdn.microsoft.com/en-us/library/ms734681.aspx>
* Version a WCF service
  + This objective may include but is not limited to: Version different types of contracts (message, service, data)
    - <http://stackoverflow.com/a/7560192/1068266>
    - <http://msdn.microsoft.com/en-us/library/ms731138.aspx>
    - <http://msdn.microsoft.com/en-us/library/ee816862.aspx>
  + Configure address, binding, and routing service versioning
    - <http://msdn.microsoft.com/en-us/library/ms731060.aspx>
* Create and configure a WCF service on Windows Azure
  + Create and configure bindings for WCF services (Azure SDK-- extensions to WCF)
    - <http://msdn.microsoft.com/en-us/library/windowsazure/hh410102.aspx>
  + Relay bindings to Azure using service bus endpoints
    - <http://msdn.microsoft.com/en-us/library/windowsazure/hh966775.aspx>
  + Integrate with the Azure service bus relay
    - <http://msdn.microsoft.com/en-us/library/windowsazure/hh367519.aspx>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/jj860549.aspx>
* Implement messaging patterns
  + Implement one way, request/reply, streaming, and duplex communication
    - <http://msdn.microsoft.com/en-us/library/ee960162.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms733742.aspx>
    - <http://msdn.microsoft.com/en-us/library/ms789010.aspx>
  + Implement Windows Azure Service Bus and Windows Azure Queues
    - <http://www.windowsazure.com/en-us/manage/services/service-bus/>
    - <http://www.windowsazure.com/en-us/develop/net/how-to-guides/service-bus-queues/>
* Host and manage services
  + Manage services concurrency (single, multiple, reentrant)
    - <http://www.codeproject.com/Articles/89858/WCF-Concurrency-Single-Multiple-and-Reentrant-and>
  + Create service hosts
    - <http://msdn.microsoft.com/en-us/library/ms731758.aspx>
  + Choose a hosting mechanism
    - <http://msdn.microsoft.com/en-us/library/ms730158.aspx>
  + Choose an instancing mode (per call, per session, singleton)
  + Activate and manage a service by using AppFabric
    - <http://www.dotnetcurry.com/ShowArticle.aspx?ID=771>
  + Implement transactional services
    - <http://msdn.microsoft.com/en-us/library/ff384250.aspx>
  + Host services in an Windows Azure worker role
    - <http://code.msdn.microsoft.com/windowsazure/CSAzureWCFWorkerRole-38b4e51d>
  + **Creating and Consuming Web API-based services (18%)**
* Design a Web API
  + Define HTTP resources with HTTP actions
  + Plan appropriate URI space, and map URI space using routing
    - <http://www.asp.net/web-api/overview/web-api-routing-and-actions/routing-in-aspnet-web-api>
  + Choose appropriate HTTP method (get, put, post, delete) to meet requirements
  + Choose appropriate format (Web API formats) for responses to meet requirements
    - <http://www.asp.net/web-api/overview/formats-and-model-binding/json-and-xml-serialization>
  + Plan when to make HTTP actions asynchronous
    - <http://blogs.msdn.com/b/henrikn/archive/2012/02/24/async-actions-in-asp-net-web-api.aspx>
* Implement a Web API
  + Accept data in JSON format (in JavaScript, in an AJAX callback)
  + Use content negotiation to deliver different data formats to clients
    - <http://www.asp.net/web-api/overview/formats-and-model-binding/content-negotiation>
  + Define actions and parameters to handle data binding
  + Use HttpMessageHandler to process client requests and server responses
    - <http://www.asp.net/web-api/overview/working-with-http/http-message-handlers>
  + Implement dependency injection, along with the dependency resolver, to create more flexible applications
    - <http://www.asp.net/web-api/overview/extensibility/using-the-web-api-dependency-resolver>
  + Implement action filters and exception filters to manage controller execution
    - <http://www.asp.net/web-api/overview/web-api-routing-and-actions/exception-handling>
  + Implement asynchronous and synchronous actions
    - <http://www.asp.net/mvc/tutorials/mvc-4/using-asynchronous-methods-in-aspnet-mvc-4>
  + Implement streaming actions
    - <http://www.strathweb.com/2013/01/asynchronously-streaming-video-with-asp-net-web-api/>
* Secure a Web API
  + Implement HTTPBasic authentication over SSL
    - <http://www.asp.net/web-api/overview/security/basic-authentication>
    - <http://www.asp.net/web-api/overview/security/working-with-ssl-in-web-api>
  + Implement Windows Auth
    - <http://www.asp.net/web-api/overview/security/integrated-windows-authentication>
  + Enable cross-domain requests
    - <http://blogs.msdn.com/b/carlosfigueira/archive/2012/02/20/implementing-cors-support-in-asp-net-web-apis.aspx>
  + Prevent cross-site request forgery (XSRF)
    - <http://www.asp.net/web-api/overview/security/preventing-cross-site-request-forgery-%28csrf%29-attacks>
  + Implement, and extend, authorization filters to control access to the application
    - <http://www.asp.net/web-api/overview/security/authentication-and-authorization-in-aspnet-web-api>
* Host and manage Web API
  + Self-host a Web API in your own process (a Windows service)
    - <http://www.asp.net/web-api/overview/hosting-aspnet-web-api/self-host-a-web-api>
  + Host Web API in an ASP.NET app
  + Host services in a Windows Azure worker role
    - <http://www.dotnetglobe.com/2012/04/hosting-aspnet-web-api-on-windows-azure.html>
  + Restricting message size
    - <http://stackoverflow.com/q/9453738/1068266>
  + Configure the host server for streaming
    - <http://www.strathweb.com/2012/09/dealing-with-large-files-in-asp-net-web-api/>
* Consume Web API web services
  + Consume Web API services by using HttpClient synchronously and asynchronously
    - <http://blogs.msdn.com/b/henrikn/archive/2012/02/16/httpclient-is-here.aspx>
  + Send and receive requests in different formats (JSON/HTML/etc.)
    - <http://stackoverflow.com/q/10679214/1068266>
    - <http://msdn.microsoft.com/en-us/library/hh944339%28v=vs.108%29.aspx>
    - <http://stackoverflow.com/q/6117101/1068266>

**Deploying Web Applications and Services (19%)**

* Design a deployment strategy
  + Create an IIS install package
    - <http://weblogs.asp.net/scottgu/archive/2010/09/13/automating-deployment-with-microsoft-web-deploy.aspx>
    - <http://msdn.microsoft.com/en-us/library/dd465323.aspx>
  + Deploy to web farms
    - <http://weblogs.asp.net/scottgu/archive/2010/09/08/introducing-the-microsoft-web-farm-framework.aspx>
    - <http://msdn.microsoft.com/en-us/library/ff731049%28v=azure.10%29.aspx>
    - <http://raquila.com/software/ms-deploy-basics/>
  + Deploy a web application by using XCopy
    - <http://help.infragistics.com/Help/NetAdvantage/WinForms/2012.2/CLR4.0/html/Win_Using_XCOPY_Deployment.html>
    - <http://msdn.microsoft.com/en-us/library/aa302347.aspx>
  + Automate a deployment from TFS or Build Server
    - <http://codingcraft.wordpress.com/2012/02/18/automated-deployment-with-tfs/>
    - <http://msdn.microsoft.com/en-us/library/vstudio/ms181709.aspx>
* Choose a deployment strategy for a Windows Azure web application
  + Perform an in-place upgrade and VIP swap
    - <http://www.windowsazure.com/en-us/develop/net/common-tasks/staging-deployment/?bcsi-ac-bbaf765720ef3335=20190C6B00000503aHYllMrACWM2RwoWOHeWxZgS+FG7BgAAAwUAAJMKZQAIBwAAAgEAAKXSEgA>
  + Configure an upgrade domain
    - <http://msdn.microsoft.com/en-us/library/windowsazure/hh472157.aspx>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/ee758711.aspx>
  + Create and configure input and internal endpoints
    - <http://msdn.microsoft.com/en-us/library/windowsazure/hh180158.aspx>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/gg433020.aspx>
    - <http://msdn.microsoft.com/en-us/library/windowsazure/gg432980.aspx>
  + Specify operating system configuration
    - <http://msdn.microsoft.com/en-us/library/windowsazure/ee758710.aspx>
    - <http://www.windowsazure.com/en-us/manage/services/cloud-services/how-to-configure-a-cloud-service/>
* Configure a web application for deployment
  + Switch from production/release mode to debug mode
    - <http://msdn.microsoft.com/en-us/library/e8z01xdh%28v=vs.80%29.aspx>
  + Use SetParameters to set up an IIS app pool, set permissions and passwords
    - <http://msdn.microsoft.com/en-us/library/ff398068.aspx>
  + Configure WCF endpoints, bindings, and behaviors
  + Transform web.config by using XSLT (for example, across development, test, and production/release environments)
    - <http://msdn.microsoft.com/en-us/library/dd465326(v=vs.100).aspx>
  + Configure Azure configuration settings
* Manage packages by using NuGet
  + Create and configure a NuGet package
    - <http://www.hanselman.com/blog/CreatingANuGetPackageIn7EasyStepsPlusUsingNuGetToIntegrateASPNETMVC3IntoExistingWebFormsApplications.aspx>
  + Install and update an existing NuGet package
    - <http://docs.nuget.org/docs/start-here/installing-nuget>
  + Connect to a local repository cache for NuGet, set up your own package repository
    - <http://docs.nuget.org/docs/creating-packages/hosting-your-own-nuget-feeds>
* Create, configure, and publish a web package
  + Create an IIS InstallPackage
    - <http://www.iis.net/learn/publish/using-web-deploy/building-a-web-deploy-package-from-visual-studio-2010>
  + Configure the build process to output a web package
    - <http://blogs.msdn.com/b/webdev/archive/2009/02/24/web-packaging-creating-web-packages-using-msbuild.aspx>
  + Apply pre- and post- condition actions to ensure that transformations are correctly applied
    - <http://msdn.microsoft.com/en-us/library/ke5z92ks.aspx>
  + Include appropriate assets (web content, certificates)
    - <http://sedodream.com/2010/05/01/WebDeploymentToolMSDeployBuildPackageIncludingExtraFilesOrExcludingSpecificFiles.aspx>
* Share assemblies between multiple applications and servers
  + Prepare the environment for use of assemblies across multiple servers (interning)
    - <http://blogs.technet.com/b/sateesh-arveti/archive/2011/11/30/look-at-sharing-common-assemblies-in-asp-net-4-5.aspx>
  + Sign assemblies by using a strong name
    - <http://msdn.microsoft.com/en-us/library/xc31ft41.aspx>
    - <http://msdn.microsoft.com/en-us/library/xwb8f617.aspx>
  + Deploy assemblies to the global assembly cache
    - <http://msdn.microsoft.com/en-us/library/dkkx7f79.aspx>
    - <http://msdn.microsoft.com/en-us/library/ex0ss12c.aspx>
  + Implement assembly versioning
    - <http://msdn.microsoft.com/en-us/library/51ket42z(v=vs.110).aspx>
  + Create an assembly manifest
    - <http://msdn.microsoft.com/en-us/library/1w45z383.aspx>
    - <http://msdn.microsoft.com/en-us/library/4w8c1y2s.aspx>
  + Configure assembly binding redirects (for example, from MVC2 to MVC3)
    - <http://msdn.microsoft.com/en-us/library/2fc472t2.aspx>