1. **Software Requirements :**
   * + - 1. Visual Studio 2015 or above
         2. Azure Account
2. **Topic:** Azure Storage – Table
3. **Description:** To create a Windows (desktop) Application called “AzureTableApp” to enable users to store data in a tabular format on the Azure Cloud.
4. **Reference links :**

<https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-how-to-use-tables>

1. **Hands on assignment**
   1. **Step -1 : Designing the UI**
      1. Project Template : Windows Forms Application

|  |
| --- |
|  |

* 1. **Step -2 : Making the Azure storage API available.**
     1. Go to Package Manager console of Visual Studio.
        1. PS>Install-package windowsazure.storage
  2. **Step - 3 : Setting up the initial configuration.**

|  |
| --- |
| Namespaces to use  using Microsoft.WindowsAzure.Storage;  using Microsoft.WindowsAzure.Storage.Table;  using Microsoft.WindowsAzure.Storage.Auth;  string storageAccountName = "";  string accountAccountKey = ""; //primary key  StorageCredentials storageCredentials = new StorageCredentials(storageAccountName, accountAccountKey);  CloudStorageAccount account = new CloudStorageAccount(storageCredentials, useHttps: true);  CloudTableClient cloudTableClient = account.CreateCloudTableClient(); |

**d. Step - 4 : Creating Azure Table.**

|  |
| --- |
| public class Product : TableEntity  {  public int ProdID { get; set; }  public string ProdName { get; set; }  public int Price { get; set; }  }  string tableName = textBox1.Text;  CloudTable azureTable=cloudTableClient.GetTableReference(tableName);  azureTable.CreateIfNotExists();  MessageBox.Show("Azure table is created"); |

1. **Step - 5 : Adding data into Azure Table.**

|  |
| --- |
| var prodEntity = new Product  {  ProdID=int.Parse(textBox2.Text),  ProdName=textBox3.Text,  Category = textBox4.Text,  };  prodEntity.RowKey = prodEntity.ProdID + " " + ProductName;  prodEntity.PartitionKey = prodEntity.Category;  TableOperation tableOperation = TableOperation.Insert(prodEntity);  azureTable.Execute(tableOperation); |

1. **Step – 6 : Listing Table Data.**

|  |
| --- |
| TableQuery<Product> query = new TableQuery<Product>();  var prodEntites = azureTable.ExecuteQuery(query);  foreach (var entity in prodEntites)  {  listBox1.Items.Add(entity.ProdID +" "+entity.ProdName +" "+entity.Category);  } |

------\*\*\*------