**И.П. Клементьев**

**ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ**

**Лабораторная работа номер 1.**

**Создание первого Windows Azure приложения**

***Цель работы***

Целью лабораторной работы является практическое освоение создания приложений Windows Azure.

***Аппаратура и программные инструменты, необходимые для лабораторной работы***

Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.

Доступ к сети Интернет.

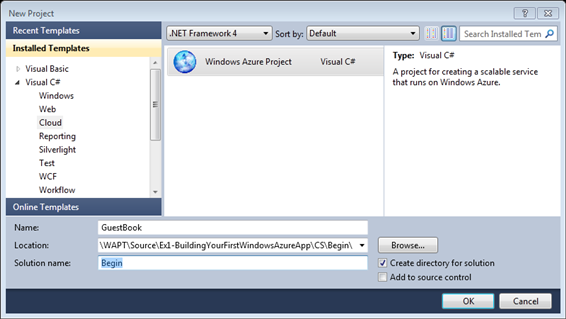
Наличие аккаунта Windows Azure.

***Продолжительность лабораторной работы***

2 академических часа

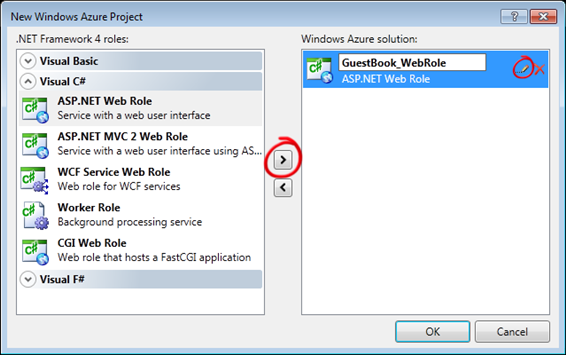
***1. Создание проекта в Visual Studio.***

1. Откройте меню **Пуск** | **Все программы**| **Microsoft Visual Studio 2010** | **Microsoft Visual Studio 2010**.
2. В меню File выберите New и затем **Project**.
3. В диалоговом окне **New Project** развернитеузел **Visual C#** и в списке **Installed Templates** выберите **Cloud**.
4. В списке **Templates** выберите **Windows AzureCloud Service**. Введите Name «**GuestBook**», имя solution «**Begin**». Затем выберите расположение внутри папки **Ex1-BuildingYourFirstWindowsAzureApp**. Убедитесь что опция **Create directory for solution** выбрана и нажмите **OK** чтобы создать проект.



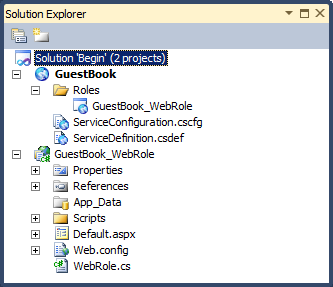
Создание нового проектаWindows Azure Cloud Service

1. В диалоге **New Cloud Service Project** разверните узел  **Visual C#** и выберите **ASP.NET Web Role**. Переместите выбранную роль с переместит в проект. Выберите роль в проекте, нажмите на изображение «карандаш» и введите имя **GuestBook\_WebRole**. Нажмите **OK** для завершения.



Добавление ролей в проект.

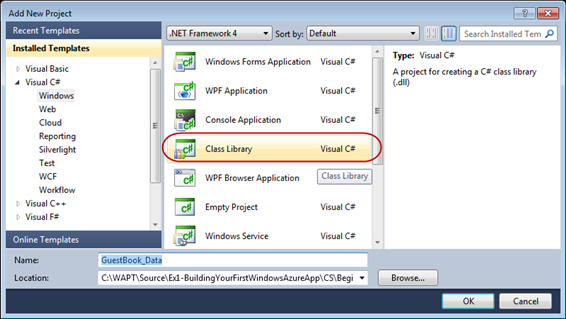
1. Обратите внимание на структуру проекта в **Solution Explorer**.



Структура проекта в Solution Explorer.

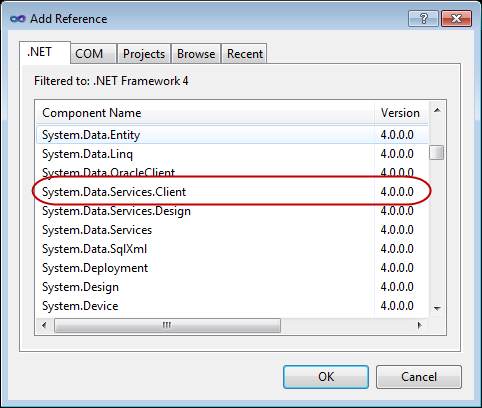
***2. Создание модели данных для элементов в Table Storage.***

1. В **Solution Explorer** нажмитеправойкнопкоймышипо **Begin,** выберите **Add | New Project**.
2. В диалоге **Add New Project,** разверните узел **Visual C#** в списке **Installed Templates**, выберите категорию **Windows** ивыделите **Class Library** в списке шаблонов. Убедитесь что выбрат **.NET Framework 3.5**. Введите имя **GuestBook\_Data** и нажмите **OK**.

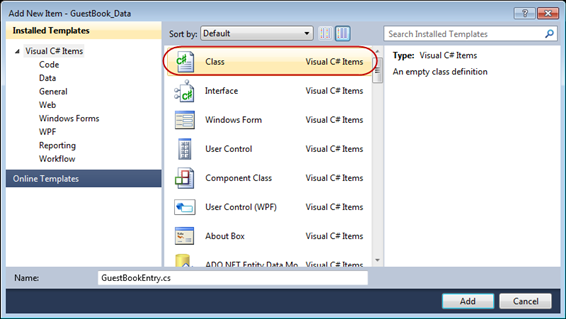


Создание библиотеки классов.

1. Удалите файл класса по умолчанию. Нажмите правой кнопкой по **Class1.cs** и выберите **Delete**. Нажмите **OK**.
2. Добавьте ссылку на библиотеку .NET для ADO.NET в проект **GuestBook\_Data**. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_Data**, выберите **Add Reference**, затем выберите закладку **.NET**, выделите компонент **System.Data.Service.Client** и нажмите **OK**.
3. Выполните пункт 4, добавив ссылку библиотеку **Microsoft.WindowsAzure.StorageClient**



1. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookEntry.cs** и нажмите **Add**.



1. Откройте файл **GuestBookEntry.cs**, добавьте в начало файла

*using Microsoft.WindowsAzure.StorageClient;*

1. Измените объявление класса**GuestBookEntry**

*public class GuestBookEntry :*

*Microsoft.WindowsAzure.StorageClient.TableServiceEntity*

*{*

*}*

1. Добавьте конструктор по умолчанию

*public GuestBookEntry()*

*{*

*PartitionKey = DateTime.UtcNow.ToString("MMddyyyy");*

*// Row key allows sorting, so we make sure the rows come back in time order.*

*RowKey = string.Format("{0:10}\_{1}", DateTime.MaxValue.Ticks - DateTime.Now.Ticks, Guid.NewGuid());*

*}*

1. Добавьте свойства

*public string Message { get; set; }*

*public string GuestName { get; set; }*

*public string PhotoUrl { get; set; }*

*public string ThumbnailUrl { get; set; }*

1. Сохраните файл **GuestBookEntry.cs**.
2. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookDataContext.cs** и нажмите **Add**.
3. Откройте файл **GuestBookDataContext.cs**, добавьте в начало файла

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.StorageClient;*

1. Измените объявление класса**GuestBookDataContext** и добавьте конструктор

*public class GuestBookDataContext : TableServiceContext*

*{*

*public GuestBookDataContext(string baseAddress, StorageCredentials credentials)*

*: base(baseAddress, credentials)*

*{ }*

*}*

1. Добавьте свойство

*public class GuestBookDataContext : TableServiceContext*

*{*

*...*

*public IQueryable<GuestBookEntry> GuestBookEntry*

*{*

*get*

*{*

*return this.CreateQuery<GuestBookEntry>("GuestBookEntry");*

*}*

*}*

*}*

1. Нажмите правой кнопкой мыши по **GuestBook\_Data** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **GuestBookDataContext.cs** и нажмите **Add**.
2. Откройте файл **GuestBookEntryDataSource.cs**, добавьте в начало файла

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.StorageClient;*

1. Далее измените класс

*public class GuestBookEntryDataSource*

*{*

*private static CloudStorageAccount storageAccount;*

*private GuestBookDataContext context;*

*}*

1. Добавьте конструктор

*public class GuestBookEntryDataSource*

*{*

*private static CloudStorageAccount storageAccount;*

*private GuestBookDataContext context;*

*static GuestBookEntryDataSource()*

*{*

*storageAccount = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");*

*CloudTableClient.CreateTablesFromModel(*

*typeof(GuestBookDataContext),*

*storageAccount.TableEndpoint.AbsoluteUri,*

*storageAccount.Credentials);*

*}*

*}*

1. Добавьте констурктор для класса **GuestBookDataEntrySource**

*public GuestBookEntryDataSource()*

*{*

*this.context = new GuestBookDataContext(storageAccount.TableEndpoint.AbsoluteUri, storageAccount.Credentials);*

*this.context.RetryPolicy = RetryPolicies.Retry(3, TimeSpan.FromSeconds(1));*

*}*

1. Добавьте методы

*public IEnumerable<GuestBookEntry> Select()*

*{*

*var results = from g in this.context.GuestBookEntry*

*where g.PartitionKey == DateTime.UtcNow.ToString("MMddyyyy")*

*select g;*

*return results;*

*}*

*public void UpdateImageThumbnail(string partitionKey, string rowKey, string thumbUrl)*

*{*

*var results = from g in this.context.GuestBookEntry*

*where g.PartitionKey == partitionKey && g.RowKey == rowKey*

*select g;*

*var entry = results.FirstOrDefault<GuestBookEntry>();*

*entry.ThumbnailUrl = thumbUrl;*

*this.context.UpdateObject(entry);*

*this.context.SaveChanges();*

*}*

1. Сохраните файл **GuestBookEntryDataSource.cs**

***3. Создание Веб роли для отображения гостевой книги.***

1. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add Reference**, затем выберите закладку **.NET**, выделите компонент **System.Data.Service.Client** и нажмите **OK**.
2. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add Reference,** затем выберите закладку **Project**, выделите **GuestBook\_Data** и нажмите **OK**.
3. Нажмите правой кнопкой по **Default.aspx** и выберите **Delete**. Нажмите **OK**.
4. В **Solution Explorer** нажмите правой кнопкой по проекту **GuestBook\_WebRole**, выберите **Add**, выделите **Existing Item.**
5. В диалоге **Add Existing Item** выберите директорию **\Source\Ex1-BuildingYourFirstWindowsAzureApp\CS\Assets**, выберите **Add.**
6. В **Solution Explorer** нажмите правой кнопкой по **Default.aspx**, выберите **View Code,** объявите следующие пространства имен

*using System.IO;*

*using System.Net;*

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.ServiceRuntime;*

*using Microsoft.WindowsAzure.StorageClient;*

*using GuestBook\_Data;*

1. В классе укажите

*private static bool storageInitialized = false;*

*private static object gate = new Object();*

*private static CloudBlobClient blobStorage;*

1. Найдите событие **SignButton\_Click** и добавьте следующий код

*protected void SignButton\_Click(object sender, EventArgs e)*

*{*

*if (FileUpload1.HasFile)*

*{*

*InitializeStorage();*

*// upload the image to blob storage*

*CloudBlobContainer container = blobStorage.GetContainerReference("guestbookpics");*

*string uniqueBlobName = string.Format("image\_{0}.jpg", Guid.NewGuid().ToString());*

*CloudBlockBlob blob = container.GetBlockBlobReference(uniqueBlobName);*

*blob.Properties.ContentType = FileUpload1.PostedFile.ContentType;*

*blob.UploadFromStream(FileUpload1.FileContent);*

*System.Diagnostics.Trace.TraceInformation("Uploaded image '{0}' to blob storage as '{1}'", FileUpload1.FileName, uniqueBlobName);*

*// create a new entry in table storage*

*GuestBookEntry entry = new GuestBookEntry() { GuestName = NameTextBox.Text, Message = MessageTextBox.Text, PhotoUrl = blob.Uri.ToString(), ThumbnailUrl = blob.Uri.ToString() };*

*GuestBookEntryDataSource ds = new GuestBookEntryDataSource();*

*ds.AddGuestBookEntry(entry);*

*System.Diagnostics.Trace.TraceInformation("Added entry {0}-{1} in table storage for guest '{2}'", entry.PartitionKey, entry.RowKey, entry.GuestName);*

*}*

*NameTextBox.Text = "";*

*MessageTextBox.Text = "";*

*DataList1.DataBind();*

*}*

1. Обновите метод**Timer1\_Tick**

*protected void Timer1\_Tick(object sender, EventArgs e)*

*{*

*DataList1.DataBind();*

*}*

1. Обновите событие **Page\_Load**

*protected void Page\_Load(object sender, EventArgs e)*

*{*

*if (!Page.IsPostBack)*

*{*

*Timer1.Enabled = true;*

*}*

*}*

1. Произведите изменения в методе **InitializeStorage**

*private void InitializeStorage()*

*{*

*if (storageInitialized)*

*{*

*return;*

*}*

*lock (gate)*

*{*

*if (storageInitialized)*

*{*

*return;*

*}*

*try*

*{*

*// read account configuration settings*

*var storageAccount = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");*

*// create blob container for images*

*blobStorage = storageAccount.CreateCloudBlobClient();*

*CloudBlobContainer container = blobStorage.GetContainerReference("guestbookpics");*

*container.CreateIfNotExist();*

*// configure container for public access*

*var permissions = container.GetPermissions();*

*permissions.PublicAccess = BlobContainerPublicAccessType.Container;*

*container.SetPermissions(permissions);*

*}*

*catch (WebException)*

*{*

*throw new WebException("Storage services initialization failure. "*

*+ "Check your storage account configuration settings. If running locally, "*

*+ "ensure that the Development Storage service is running.");*

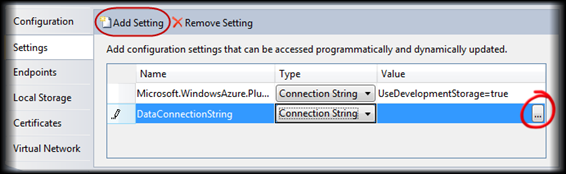
*}*

*storageInitialized = true;*

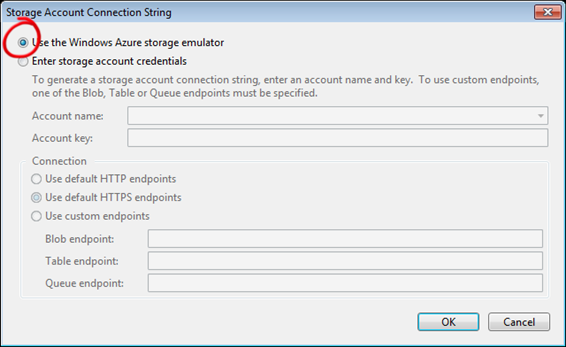
*}*

*}*

1. В **Solution Explorer** разверните узел **Roles** в проекте **GuestBook**. Нажмите два раза по **GuestBook\_WebRole**, откроется свойства данной роли, выберите закладку **Setting**. Нажмите **Add Setting**, наберите “DataConnectionString” в колонке **Name**, измените **Type** не *ConnectionString* и нажмите **Add** **Setting**.



1. В диалоге **Storage Connection String** выберите **Use development storage** и нажмите **OK.**



1. Сохраните изменения.
2. В проекте GuestBook\_WebRole, откройте файл Global.asax.cs
3. Объявите пространства имен

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.ServiceRuntime;*

1. Вставьте следующий код внутрь метода **Application\_Start**, заменим содержимое по умолчанию

*void Application\_Start(object sender, EventArgs e)*

*{*

*Microsoft.WindowsAzure.CloudStorageAccount.SetConfigurationSettingPublisher((configName, configSetter) =>*

*{*

*configSetter(RoleEnvironment.GetConfigurationSettingValue(configName));*

*});*

*}*

***4. Организация очереди рабочих элементов для обработки в фоне.***

1. В **Solution Explorer** нажмите правой кнопкой по **Default.aspx**, выберите **View Code,** объявите элемент клиента очереди

*public partial class \_Default : System.Web.UI.Page*

*{*

*private static bool storageInitialized = false;*

*private static object gate = new Object();*

*private static CloudBlobClient blobStorage;*

*private static CloudQueueClient queueStorage;*

*...*

*}*

1. Найдите метод **InitializeStorage** и вставьте следующий код внутрь данного метода

*public partial class Default : System.Web.UI.Page*

*{*

*...*

*private void InitializeStorage()*

*{*

*...*

*try*

*{*

*...*

*// configure container for public access*

*var permissions = container.GetPermissions();*

*permissions.PublicAccess = BlobContainerPublicAccessType.Container;*

*container.SetPermissions(permissions);*

*// create queue to communicate with worker role*

*queueStorage = storageAccount.CreateCloudQueueClient();*

*CloudQueue queue = queueStorage.GetQueueReference("guestthumbs");*

*queue.CreateIfNotExist();*

*}*

*catch (WebException)*

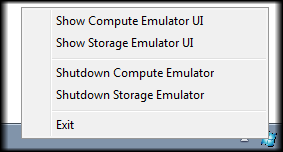
*{*

*...*

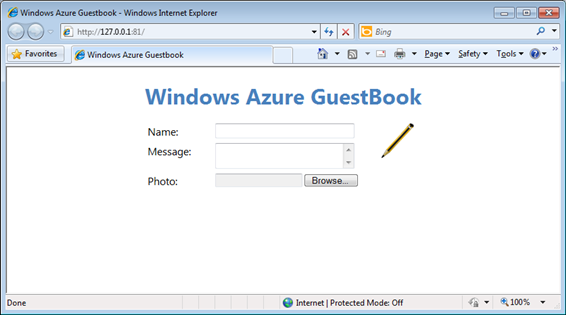
*}*

**Проверка**

1. Нажмите **F5** для запуска сервиса. Сервис запустится в development fabric. Для открытия пользовательского интерфейса необходимо нажать правой кнопкой мыши на значке в области уведомления панели задач и выбрать **Show Development Fabric UI**



1. Переключитесь на Internet Explorer для просмотра приложения GuestBook
2. Добавьте новую запись в гостевой книг



***Литература***

* 1. Windows Azure Platform Training Kit - January 2011 Update

**И.П. Клементьев**

**ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ**

**Лабораторная работа номер 2.**

**Развертывание приложения Windows Azure**

***Цель работы***

Целью лабораторной работы является практическое освоение процесса развертывания приложений Windows Azure.

***Аппаратура и программные инструменты, необходимые для лабораторной работы***

Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.

Доступ к сети Интернет.

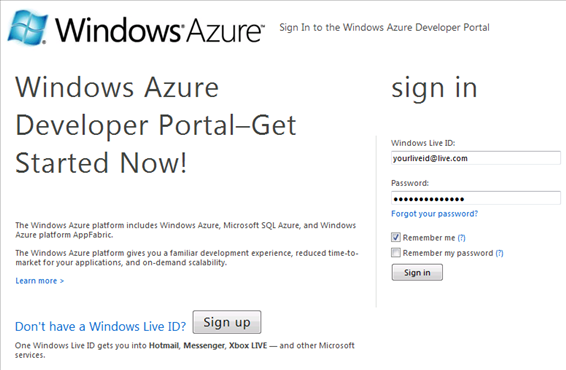
Наличие аккаунта Windows Azure.

***Продолжительность лабораторной работы***

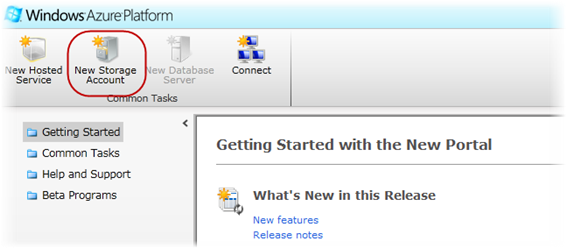
2 академических часа

***1. Создание Storage Account***

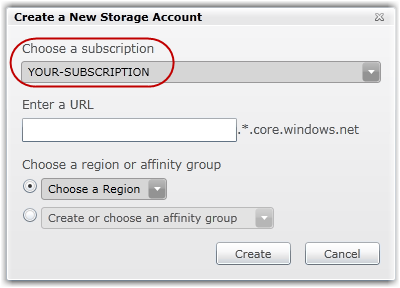
1. Откройте в веб браузере адрес [http://windows.azure.com](http://windows.azure.com/) и войдите используя Windows Live ID, ассоциированный с учетной записью Windows Azure



1. На панели инструментов Windows Azure выберите **New Storage Account.**



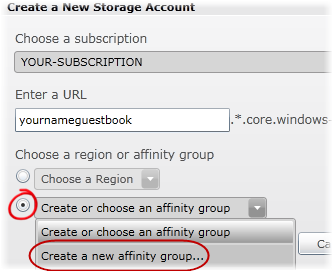
1. В диалоге **Create a New Storage Account** выберите Вашу подписку из списка.



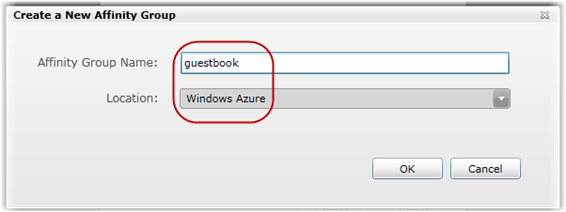
1. Введите имя для Вашего **Storage Account.**



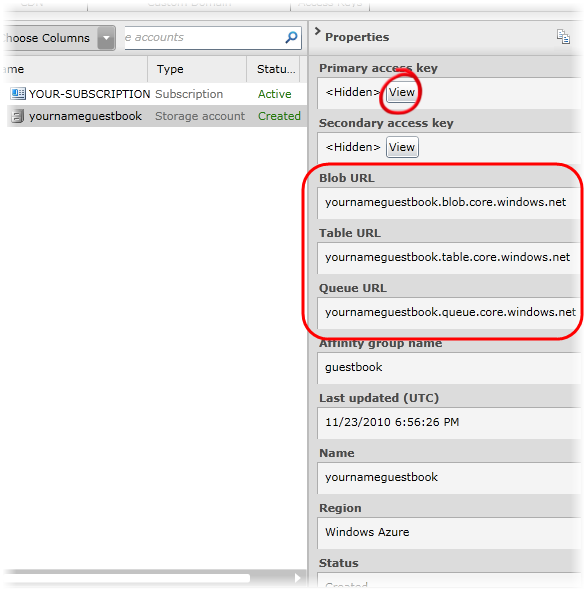
1. Выберите настройку **Create or choose an affinity group** и затем выберите из списка **Create a new affinity group**



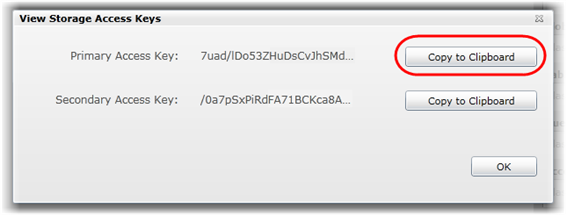
1. В диалоге **Create a New Affinity Group** введите **Affinity Group Name**, выберите размещение и нажмите **OK**.



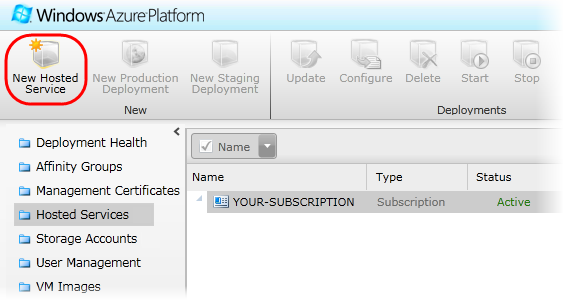
1. Вернитесь в диалог **Create a New Storage Account** и нажмите **Create** для создания нового **Storage Account.**



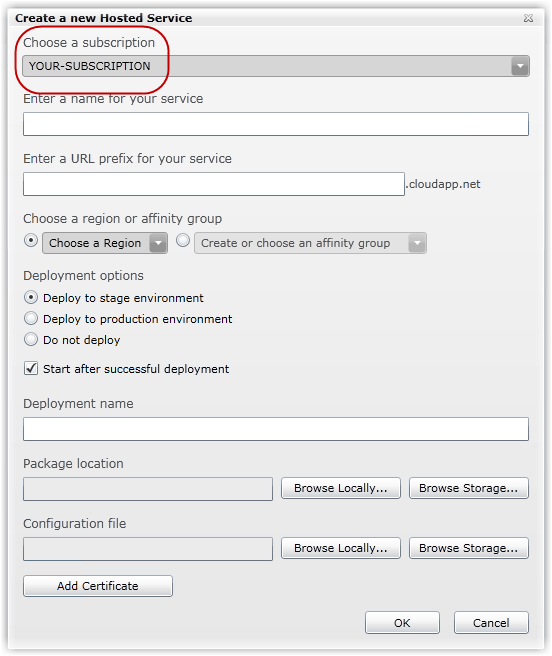
1. Нажмите кнопку **View**. В диалоге **View Storage Access Keys** нажмите кнопку **Copy to Clipboard** следующую за **Primary Access Key**.



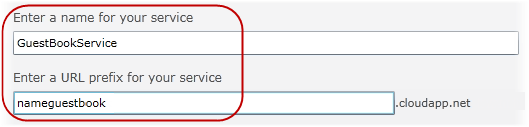
1. Нажмите **Hosted Services** на левой панели. Нажмите кнопку **New Hosted Service** на панели инструментов.



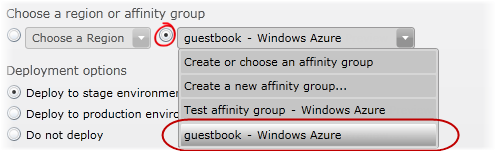
1. В диалоге **Create a new Hosted Service** выберите подписку и списка **Choose a subscription**.



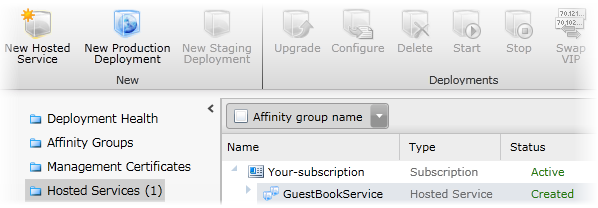
1. Введите имя сервиса **Enter a name for your service** и выберите префикс адреса в **Enter a URL prefix for your service**



1. Выберите настройку **Create or choose an affinity group** и затем выберите из списка **guestbook**



1. Выберите настройку **Do not Deploy**.
2. Нажмите **OK** для создания сервиса и дождитесь завершения процесса инициализации

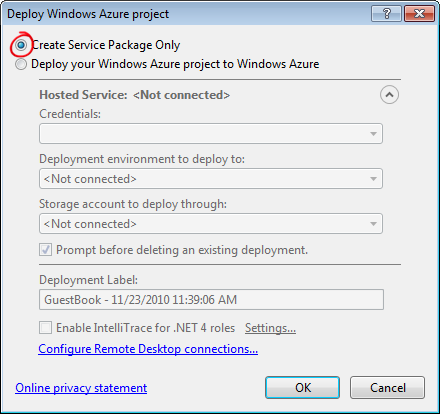


***2. Развертывание приложения на портале Windows Azure Platform***

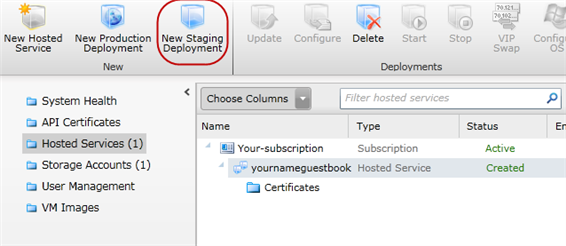
1. Откройте меню **Пуск** | **Все программы**| **Microsoft Visual Studio 2010** и запустите **Microsoft Visual Studio 2010** от имени администратора.
2. В меню **File** выберите **Open** и затем **Project/Solution**. Откройте файл **Begin.sln** проекта **Ex3-WindowsAzureDeployment**
3. Для изменения конфигурации хранилища перед развертыванием сервиса, откройте файл **ServiceConfiguration.cscfg** расположенный в сервисе **GuestBook**. Замените значение [YOUR\_ACCOUNT\_NAME] на значение **Storage Account Name**, которое Вы выбрали когда настраивали **Storage account** в Задании 1.
4. Далее замените значение [YOUR\_ACCOUNT\_KEY] на значение **Primary Access Key** которое Вы получили раннее, при создании **Storage account** в Задании 1



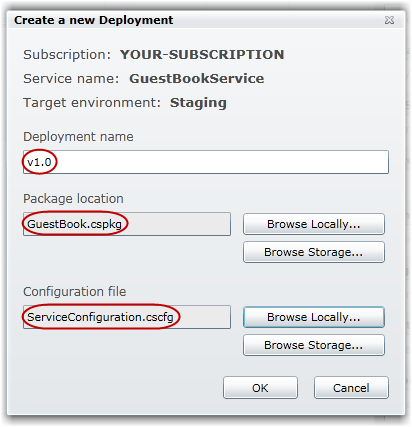
1. Необходимо создать пакет для развертывания в облаке. Для этого откройте меню правой кнопкой мыши в проекте **GuestBook** и выберите **Publish**. В диалоге **Deploy Windows Azure project**, выберите настройку **Create Service Package Only** и затем нажмите **OK**.

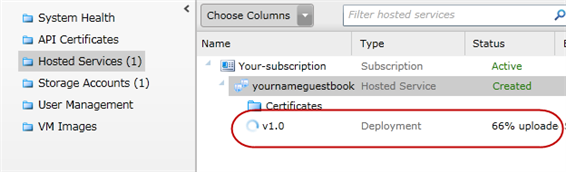


1. Переключитесь обратно в окно браузера, где открыт портал управления Windows Azure.
2. На портале выберите сервис, который Вы создали в предыдущем упражнении и нажмите **New Staging Deployment** на панели инструментов.

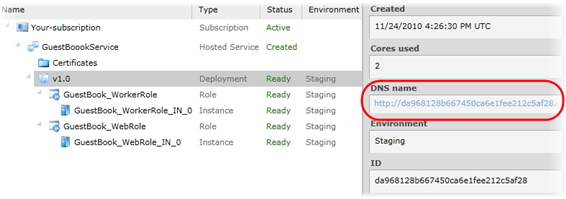


1. В диалоге **Create a new Deployment** выберите **Package location**, нажмите **Browse Locally**, выберите папку где был создан пакет в пункте 4 и затем выберите файл **GuestBook.cspkg**.
2. Далее выберите **Configuration File**, нажмите **Browse Locally** и выберите **ServiceConfiguration.cscfg** из той же папки (пункт 8).
3. Введите имя **Deployment name** и нажмите **OK.** В диалоге с предупреждением выберите **Yes**.





1. Дождитесь завершения процесса развертывания, это можно занять несколько минут.



***Литература***

* 1. Windows Azure Platform Training Kit - January 2011 Update

**И.П. Клементьев**

**ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ**

**Лабораторная работа номер 3.**

***Работа с Blob***

***Цель работы***

Целью лабораторной работы является практическое освоение процесса работы с Blob в Windows Azure.

***Аппаратура и программные инструменты, необходимые для лабораторной работы***

Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.

Доступ к сети Интернет.

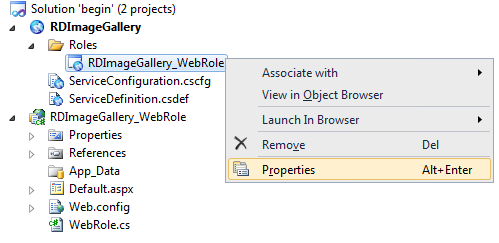
Наличие аккаунта Windows Azure.

***Продолжительность лабораторной работы***

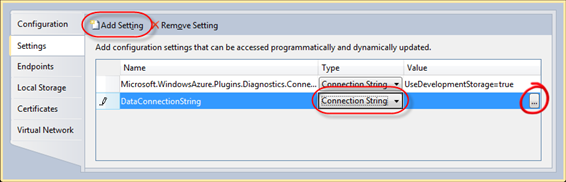
2 академических часа

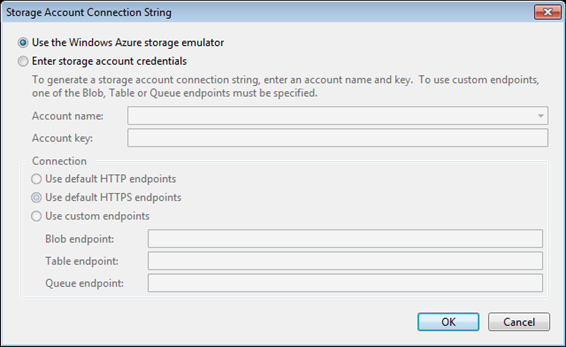
***1. Получение Blob данных из хранилища***

1. Откройте меню **Пуск** | **Все программы**| **Microsoft Visual Studio 2010** | **Microsoft Visual Studio 2010**.
2. В меню **File** выберите **Open** и затем **Project/Solution**. Откройте файл проекта **ExploringWindowsAzureStorageVS2010\Source\Ex02-WorkingWithBlobs\begin\CS\begin.sln**
3. В **Solution Explorer**, в проекте **RDImageGallery** нажмитеправойкнопкой по узлу **RDImageGallery\_WebRole** и выберите **Properties**

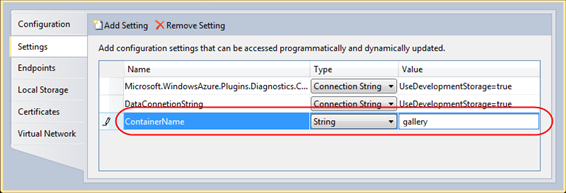


1. На закладке **Settings** создайте **ConnectionString** с именем **DataConnectionStrin***.* ***Выберите*****Use development storage**





1. **Создайте String** с именем **ContainerName** и значением **gallery**



1. В **Solution Explorer** нажмите правой кнопкой по **Default.aspx** в проекте **RDImageGallery\_WebRole**, выберите **View Code,** объявите следующие пространства имен
2. Убедитесь что в начале файла объявлены пространства имен

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.StorageClient;*

*using Microsoft.WindowsAzure.ServiceRuntime;*

1. В конец класса **\_Default** добавьте метод

*private void EnsureContainerExists()*

*{*

*var container = GetContainer();*

*container.CreateIfNotExist();*

*var permissions = container.GetPermissions();*

*permissions.PublicAccess = BlobContainerPublicAccessType.Container;*

*container.SetPermissions(permissions);*

*}*

1. В конец класса **\_Default** добавьте метод

*private CloudBlobContainer GetContainer()*

*{*

*// Get a handle on account, create a blob storage client and get container proxy*

*var account = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");*

*var client = account.CreateCloudBlobClient();*

*return client.GetContainerReference(RoleEnvironment.GetConfigurationSettingValue("ContainerName"));*

*}*

1. Добавьте следующий код в метод **Page\_Load**

*protected void Page\_Load(object sender, EventArgs e)*

*{*

*try*

*{*

*if (!IsPostBack)*

*{*

*this.EnsureContainerExists();*

*}*

*this.RefreshGallery();*

*}*

*catch (System.Net.WebException we)*

*{*

*status.Text = "Network error: " + we.Message;*

*if (we.Status == System.Net.WebExceptionStatus.ConnectFailure)*

*{*

*status.Text += "<br />Please check if the blob storage service is running at " +*

*ConfigurationManager.AppSettings["storageEndpoint"];*

*}*

*}*

*catch (StorageException se)*

*{*

*Console.WriteLine("Storage service error: " + se.Message);*

*}*

*}*

1. В конец класса **\_Default** добавьте метод

*private void RefreshGallery()*

*{*

*images.DataSource =*

*this.GetContainer().ListBlobs(new BlobRequestOptions()*

*{*

*UseFlatBlobListing = true,*

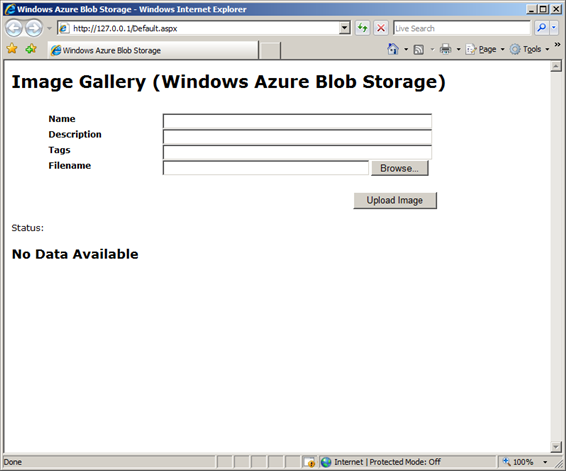
*BlobListingDetails = BlobListingDetails.All*

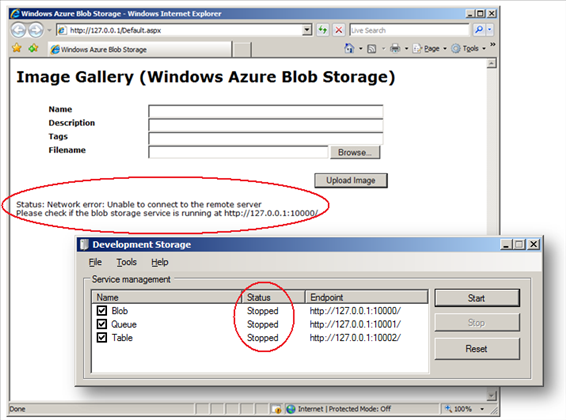
*});*

*images.DataBind();*

*}*

1. Нажмите **F5** для запустка приложения. Запустится браузер с приложением.





***2. Загрузка Blob данных в хранилище.***

1. Откройте ***Default.aspx.cs***
2. Добавьте метод в конец страницы

*private void SaveImage(string id, string name, string description, string tags, string fileName, string contentType, byte[] data)*

*{*

*// Create a blob in container and upload image bytes to it*

*var blob = this.GetContainer().GetBlobReference(name);*

*blob.Properties.ContentType = contentType;*

*// Create some metadata for this image*

*var metadata = new NameValueCollection();*

*metadata["Id"] = id;*

*metadata["Filename"] = fileName;*

*metadata["ImageName"] = String.IsNullOrEmpty(name) ? "unknown" : name;*

*metadata["Description"] = String.IsNullOrEmpty(description) ? "unknown" : description;*

*metadata["Tags"] = String.IsNullOrEmpty(tags) ? "unknown" : tags;*

*// Add and commit metadata to blob*

*blob.Metadata.Add(metadata);*

*blob.UploadByteArray(data);*

*}*

1. Измените метод **upload\_Click**

*protected void upload\_Click(object sender, EventArgs e)*

*{*

*if (imageFile.HasFile)*

*{*

*status.Text = "Inserted [" + imageFile.FileName + "] - Content Type [" + imageFile.PostedFile.ContentType + "] - Length [" + imageFile.PostedFile.ContentLength + "]";*

*this.SaveImage(*

*Guid.NewGuid().ToString(),*

*imageName.Text,*

*imageDescription.Text,*

*imageTags.Text,*

*imageFile.FileName,*

*imageFile.PostedFile.ContentType,*

*imageFile.FileBytes*

*);*

*RefreshGallery();*

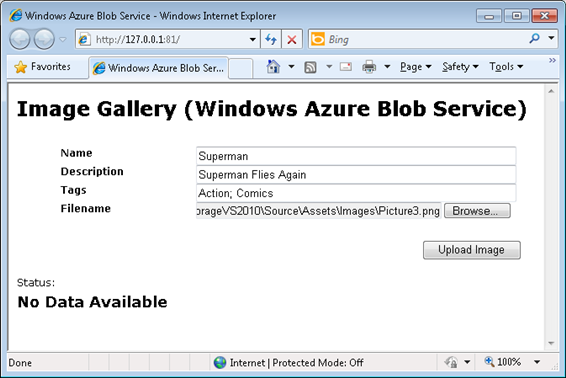
*}*

*else*

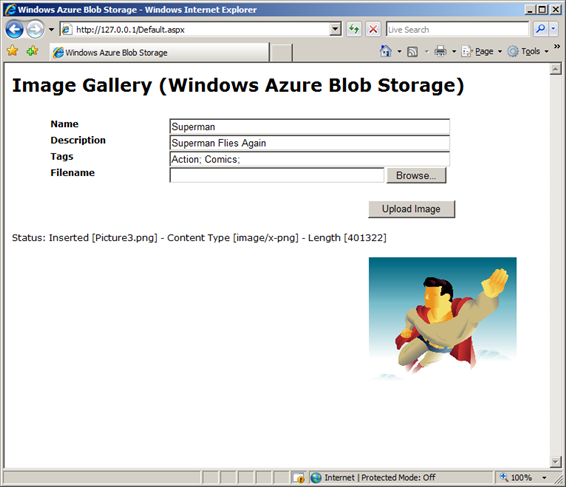
*status.Text = "No image file";*

*}*

1. Нажмите F5 для запуска приложения
2. Введете метаданные **Name**, **Description** и **Tags**. Для выбора изображения нажмите **Browse**

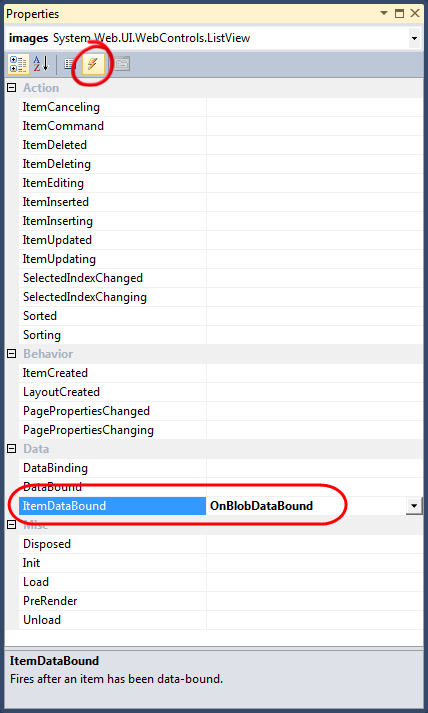


1. Нажмите **Upload Image** для публикации изображения в веб приложении



***3. Извлечение метаданных для Blob в хранилище.***

1. Откройте ***Default.aspx*** в режиме Design, выберите контрол *images***ListView** и в окне свойств нажмите кнопку **Events**



1. Найдите метод **OnBlobDataBound** и вставьте следующий код

*protected void OnBlobDataBound(object sender, ListViewItemEventArgs e)*

*{*

*if (e.Item.ItemType == ListViewItemType.DataItem)*

*{*

*var metadataRepeater = e.Item.FindControl("blobMetadata") as Repeater;*

*var blob = ((ListViewDataItem)(e.Item)).DataItem as CloudBlob;*

*// If this blob is a snapshot, rename button to "Delete Snapshot"*

*if (blob != null)*

*{*

*if(blob.SnapshotTime.HasValue)*

*{*

*var delBtn = e.Item.FindControl("deleteBlob") as LinkButton;*

*if (delBtn != null) delBtn.Text = "Delete Snapshot";*

*var snapshotBtn = e.Item.FindControl("SnapshotBlob") as LinkButton;*

*if (snapshotBtn != null) snapshotBtn.Visible = false;*

*}*

*if (metadataRepeater != null)*

*{*

*//bind to metadata*

*metadataRepeater.DataSource = from key in blob.Metadata.AllKeys*

*select new*

*{*

*Name = key,*

*Value = blob.Metadata[key]*

*};*

*metadataRepeater.DataBind();*

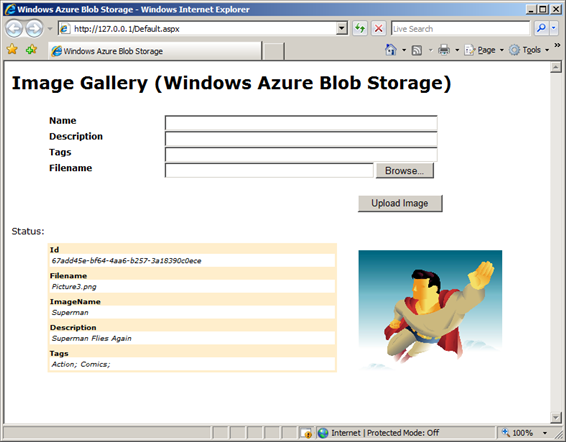
*}*

*}*

*}*

*}*

1. Нажмите **F5** для запуска приложения. Убедитесь что отображаются метаданные для изображения, загруженного ранее



***4. Удаление Blob из хранилища***

1. Откройте ***Default.aspx*** в режиме Source, найдте **ItemTemplate** для контрола **asp:ListView.** Раскомментируйте код , следующий за контролом **blobMetadata**

*<div class="item">*

*<ul style="width:40em;float:left;clear:left" >*

*<asp:Repeater ID="blobMetadata" runat="server">*

*<ItemTemplate>*

*<li><%# Eval("Name") %><span><%# Eval("Value") %></span></li>*

*</ItemTemplate>*

*</asp:Repeater>*

*<li>*

*<asp:LinkButton ID="deleteBlob"*

*OnClientClick="return confirm('Delete image?');"*

*CommandName="Delete"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Delete" oncommand="OnDeleteImage" />*

*</li>*

*</ul>*

*<img src="<%# Eval("Uri") %>" alt="<%# Eval("Uri") %>" style="float:left"/>*

*</div>*

1. Добавьте следующий код в файл **Default.aspx.cs**

*protected void OnDeleteImage(object sender, CommandEventArgs e)*

*{*

*try*

*{*

*if (e.CommandName == "Delete")*

*{*

*var blobUri = (string)e.CommandArgument;*

*var blob = this.GetContainer().GetBlobReference(blobUri);*

*blob.DeleteIfExists();*

*RefreshGallery();*

*}*

*}*

*catch (StorageClientException se)*

*{*

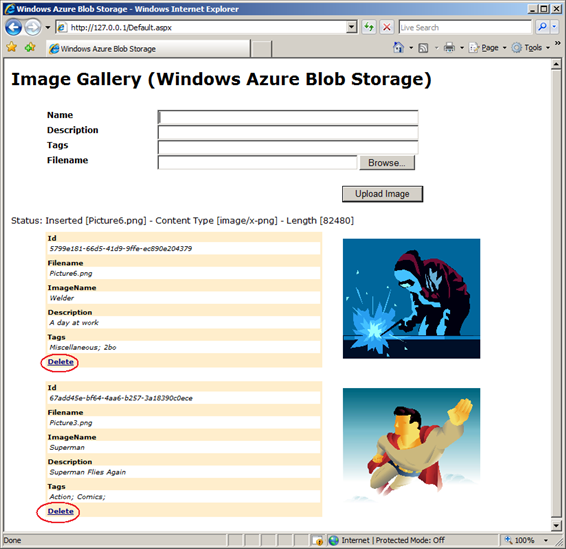
*status.Text = "Storage client error: " + se.Message;*

*}*

*catch (Exception) { }*

*}*

1. Нажмите **F5** для запуска приложения
2. Добвьте еще несколько изображений и нажмите **Delete** на любом из изображений



***5. Копирование Blob***

1. Откройте ***Default.aspx*** в режиме Source, найдте **ItemTemplate** для контрола **asp:ListView.** Раскомментируйте следующий код

*<div class="item">*

*<ul style="width:40em;float:left;clear:left" >*

*<asp:Repeater ID="blobMetadata" runat="server">*

*<ItemTemplate>*

*<li><%# Eval("Name") %><span><%# Eval("Value") %></span></li>*

*</ItemTemplate>*

*</asp:Repeater>*

*<li>*

*<asp:LinkButton ID="deleteBlob"*

*OnClientClick="return confirm('Delete image?');"*

*CommandName="Delete"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Delete" oncommand="OnDeleteImage" />*

*<asp:LinkButton ID="CopyBlob"*

*OnClientClick="return confirm('Copy image?');"*

*CommandName="Copy"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Copy" oncommand="OnCopyImage" />*

*</li>*

*</ul>*

*<img src="<%# Eval("Uri") %>" alt="<%# Eval("Uri") %>" style="float:left"/>*

*</div>*

1. Добавьте в файл **Default.aspx.cs**

*protected void OnCopyImage(object sender, CommandEventArgs e)*

*{*

*if (e.CommandName == "Copy")*

*{*

*// Prepare an Id for the copied blob*

*var newId = Guid.NewGuid();*

*// Get source blob*

*var blobUri = (string)e.CommandArgument;*

*var srcBlob = this.GetContainer().GetBlobReference(blobUri);*

*// Create new blob*

*var newBlob = this.GetContainer().GetBlobReference(newId.ToString());*

*// Copy content from source blob*

*newBlob.CopyFromBlob(srcBlob);*

*// Explicitly get metadata for new blob*

*newBlob.FetchAttributes(new BlobRequestOptions{BlobListingDetails = BlobListingDetails.Metadata});*

*// Change metadata on the new blob to reflect this is a copy via UI*

*newBlob.Metadata["ImageName"] = "Copy of \"" + newBlob.Metadata["ImageName"] + "\"";*

*newBlob.Metadata["Id"] = newId.ToString();*

*newBlob.SetMetadata();*

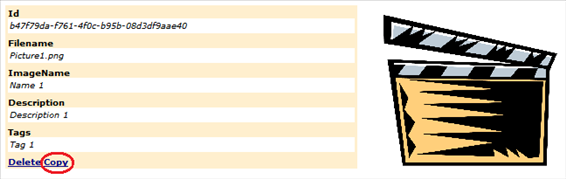
*// Render all blobs*

*RefreshGallery();*

*}*

*}*

1. Нажмите **F5** для запуска приложения
2. Добвьте еще несколько изображений и нажмите **Copy** на любом из изображений



***6. получение снимков Blob***

1. Откройте ***Default.aspx*** в режиме Source, найдте **ItemTemplate** для контрола **asp:ListView.** Раскомментируйте следующий код

*<div class="item">*

*<ul style="width:40em;float:left;clear:left" >*

*<asp:Repeater ID="blobMetadata" runat="server">*

*<ItemTemplate>*

*<li><%# Eval("Name") %><span><%# Eval("Value") %></span></li>*

*</ItemTemplate>*

*</asp:Repeater>*

*<li>*

*<asp:LinkButton ID="deleteBlob"*

*OnClientClick="return confirm('Delete image?');"*

*CommandName="Delete"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Delete" oncommand="OnDeleteImage" />*

*<asp:LinkButton ID="CopyBlob"*

*OnClientClick="return confirm('Copy image?');"*

*CommandName="Copy"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Copy" oncommand="OnCopyImage" />*

*<asp:LinkButton ID="SnapshotBlob"*

*OnClientClick="return confirm('Snapshot image?');"*

*CommandName="Snapshot"*

*CommandArgument='<%# Eval("Uri")%>'*

*runat="server" Text="Snapshot" oncommand="OnSnapshotImage" />*

*</li>*

*</ul>*

*<img src="<%# Eval("Uri") %>" alt="<%# Eval("Uri") %>" style="float:left"/>*

*</div>*

1. В файл **Default.aspx.cs** добавьте

*protected void OnSnapshotImage(object sender, CommandEventArgs e)*

*{*

***if (e.CommandName == "Snapshot")***

***{***

***// Get source blob***

***var blobUri = (string) e.CommandArgument;***

***var srcBlob = this.GetContainer().GetBlobReference(blobUri);***

***// Create a snapshot***

***var snapshot = srcBlob.CreateSnapshot();***

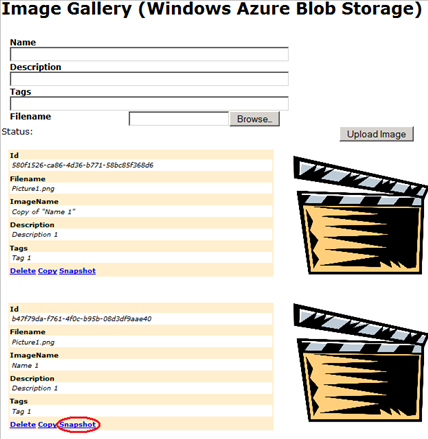
***status.Text = "A snapshot has been taken for image blob:" + srcBlob.Uri + " at " + snapshot.SnapshotTime;***

***RefreshGallery();***

***}***

*}*

1. Нажмите **F5** для запуска приложения
2. Нажмите **Snapshot** на любом из изображений



***Литература***

* 1. Windows Azure Platform Training Kit - January 2011 Update

**И.П. Клементьев**

**ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ**

**Лабораторная работа номер 3.**

***Работа с Tables***

***Цель работы***

Целью лабораторной работы является практическое освоение процесса работы с Tables в Windows Azure.

***Аппаратура и программные инструменты, необходимые для лабораторной работы***

Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.

Доступ к сети Интернет.

Наличие аккаунта Windows Azure.

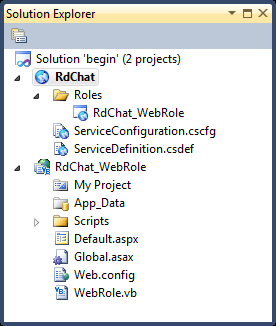
***Продолжительность лабораторной работы***

2 академических часа

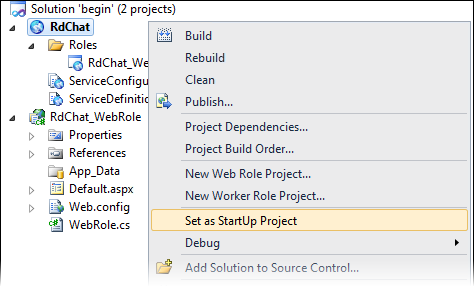
***1. Настройка Storage Account Settings***

1. Откройте меню **Пуск** | **Все программы**| **Microsoft Visual Studio 2010** | **Microsoft Visual Studio 2010**.
2. В меню **File** выберите **Open** и затем **Project/Solution**. Откройте файл проекта

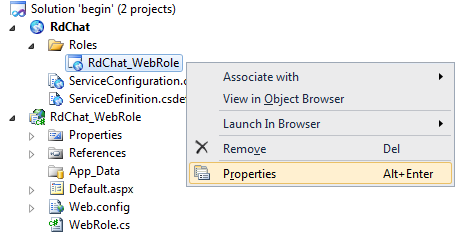
**ExploringWindowsAzureStorageVS2010\Source\Ex01-WorkingWithTables\begin\C#\begin.sln**



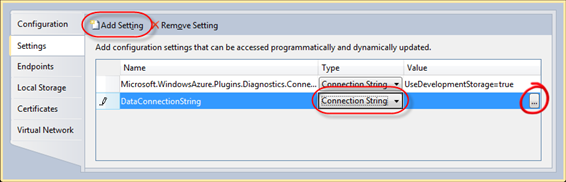
1. Убедитесь что проект **RdChat\_WebRole** запускается по умолчанию.



1. В **Solution Explorer**, в проекте **RdChat** нажмитеправойкнопкой по узлу **RdChat\_WebRole** и выберите **Properties**



1. На закладке **Settings** создайте **ConnectionString** с именем **DataConnectionString***.* Выберите**Use development storage**



1. Сохраните изменения

***2. Создание классов для модели Model the Table Schema***

1. В **Solution Explorer** нажмите правой кнопкой по проекту **RdChat\_WebRole**, выберите **Add Reference**, затем выберите закладку **.NET**, выделите компонент **System.Data.Service.Client** и нажмите **OK**.
2. Нажмите правой кнопкой мыши по **RdChat\_WebRole** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **Message.cs** и нажмите **Add**.
3. Обновите класс

*public class Message :*

*Microsoft.WindowsAzure.StorageClient.TableServiceEntity*

*{*

*}*

1. Добавьте конструктор

*public Message()*

*{*

*PartitionKey = "a";*

*RowKey = string.Format("{0:10}\_{1}", DateTime.MaxValue.Ticks - DateTime.Now.Ticks, Guid.NewGuid());*

*}*

1. Добавьте два свойства

*public string Name { get; set; }*

*public string Body { get; set; }*

1. Сохраните изменения в **Message.cs**
2. Нажмите правой кнопкой мыши по **RdChat\_WebRole** в **Solution Explorer**, выберите Add, затем **Class**. В диалоге **Add New Item** введите имя **MessageDataServiceContext.cs** и нажмите **Add**.
3. Добавьте пространства имен в начало файла

*using Microsoft.WindowsAzure.StorageClient;*

*using Microsoft.WindowsAzure;*

1. Замените объявление класса

*namespace RdChat\_WebRole*

*{*

*public class MessageDataServiceContext : TableServiceContext*

*{*

*public MessageDataServiceContext(string baseAddress, StorageCredentials credentials)*

*: base(baseAddress, credentials)*

*{*

*}*

*}*

*}*

1. Добавьте свойство

*public IQueryable<Message> Messages*

*{*

*get*

*{*

*return this.CreateQuery<Message>("Messages");*

*}*

*}*

1. Добавьте метод

*public void AddMessage(string name, string body)*

*{*

*this.AddObject("Messages", new Message { Name = name, Body = body });*

*this.SaveChanges();*

*}*

1. В меню **Build** выберите **Build Solution**.

***3. Создание пользовательского интерфейса Chat***

1. В файле **Global.asax.cs** найдите медо **Application\_Start** и вставьте следующий код.

*protected void Application\_Start()*

*{*

*...*

*/// Create data table from MessageDataServiceContext*

*/// It is recommended the data tables should be only created once. It is typically done as a*

*/// provisioning step and rarely in application code.*

*var account = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");*

*// dynamically create the tables*

*CloudTableClient.CreateTablesFromModel(typeof(MessageDataServiceContext),*

*account.TableEndpoint.AbsoluteUri, account.Credentials);*

*}*

1. Убедитесь что в файле **Global.asax.cs** объявлены пространства имен

*using Microsoft.WindowsAzure;*

*using Microsoft.WindowsAzure.ServiceRuntime;*

*using Microsoft.WindowsAzure.StorageClient;*

1. Разверните узел **RDChat\_WebRole** в **Solution Explorer**, откройте меню правой кнопкой мыши на **Default.aspx** и выберите **View Code**. У бедитесь что следующие пространства имен есть в файле **Default.aspx.cs** / **Default.aspx.vb**

*using System.Data.Services.Client;*

*using Microsoft.WindowsAzure;*

1. В файле **Default.aspx.cs** найдитесобытие **SubmitButton\_Click** и вставьте следующий код

*protected void SubmitButton\_Click(object sender, EventArgs e)*

*{*

*var statusMessage = String.Empty;*

*try*

*{*

*var account = CloudStorageAccount.FromConfigurationSetting("DataConnectionString");*

*var context = new MessageDataServiceContext(account.TableEndpoint.ToString(), account.Credentials);*

*context.AddMessage(this.nameBox.Text, this.messageBox.Text);*

*this.messageList.DataSource = context.Messages;*

*this.messageList.DataBind();*

*}*

*catch (DataServiceRequestException ex)*

*{*

*statusMessage = "Unable to connect to the table storage server. Please check that the service is running.<br>"*

*+ ex.Message;*

*}*

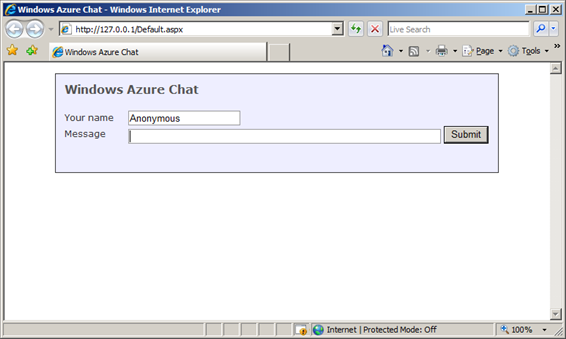
*status.Text = statusMessage;*

*}*

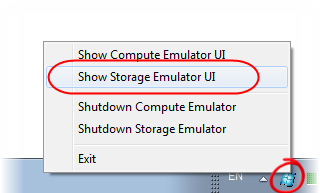
1. Сохраните все изменения. В меню **Build** выберите **Build Solution**.

Проверка:

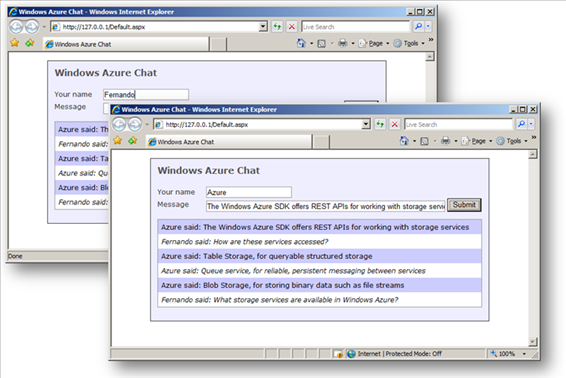
1. Для запуска приложения нажмите F5. Приложение откроется в веб браузере



1. В области уведомлений панели задач нажмите правой кнопкой по значку и выберите **Show Development Storage UI**



1. Наберите сообщение и нажмите кнопку **Submit**. Через несколько секунд страница обновится и появится сообщение.



***Литература***

* 1. Windows Azure Platform Training Kit - January 2011 Update