# Лабораторная работа 1. Установка и настройка Hyper-V

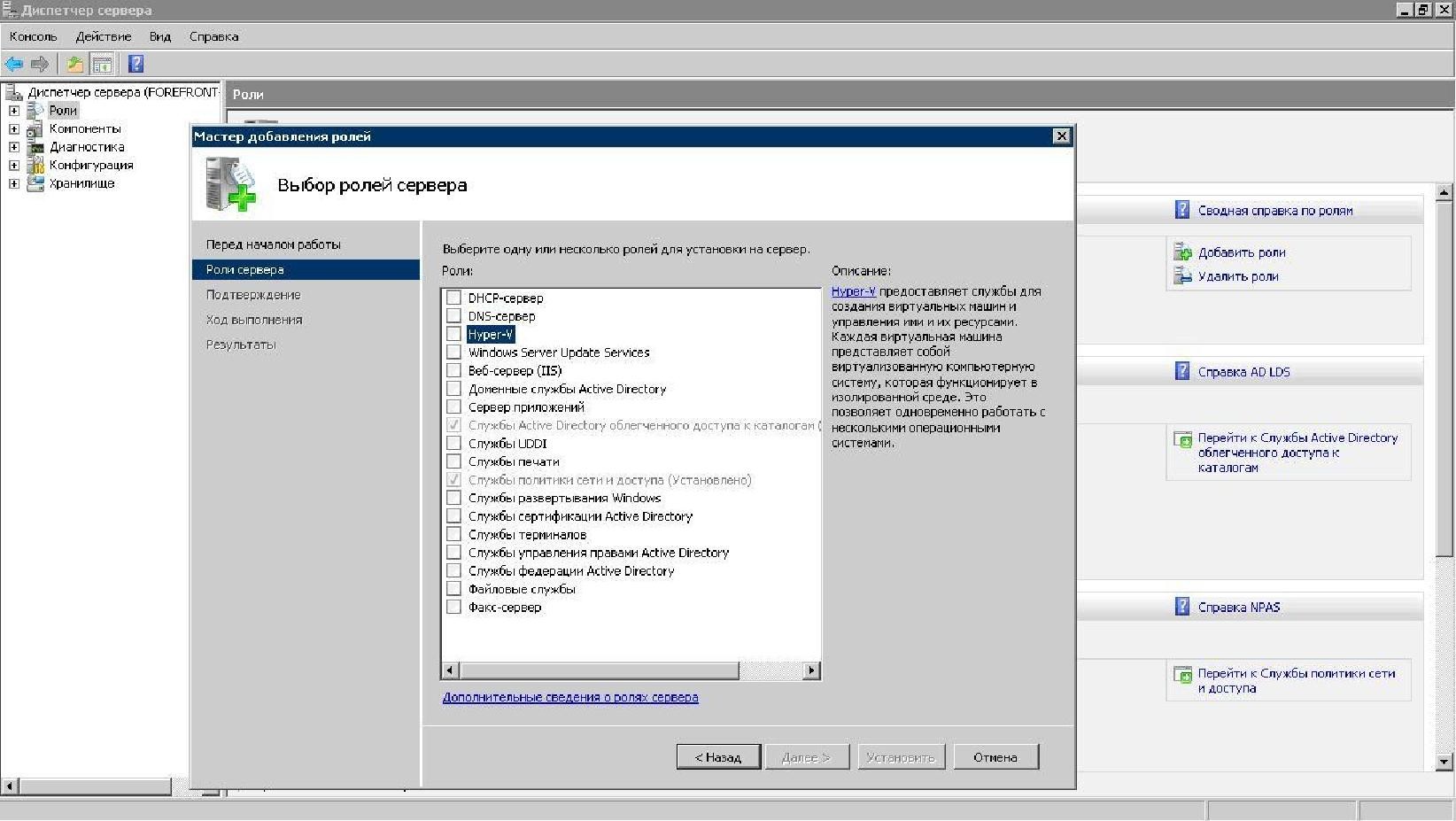
Целью лабораторной работы является практическое освоение технологий виртуализации Hyper-V.

# Аппаратура и программные инструменты, необходимые для лабораторной работы

Компьютер или сервер, поддерживающий виртуализацию, операционная система Microsoft Windows Server 2008 (R2)

**Продолжительность лабораторной работы** 6 академических часов

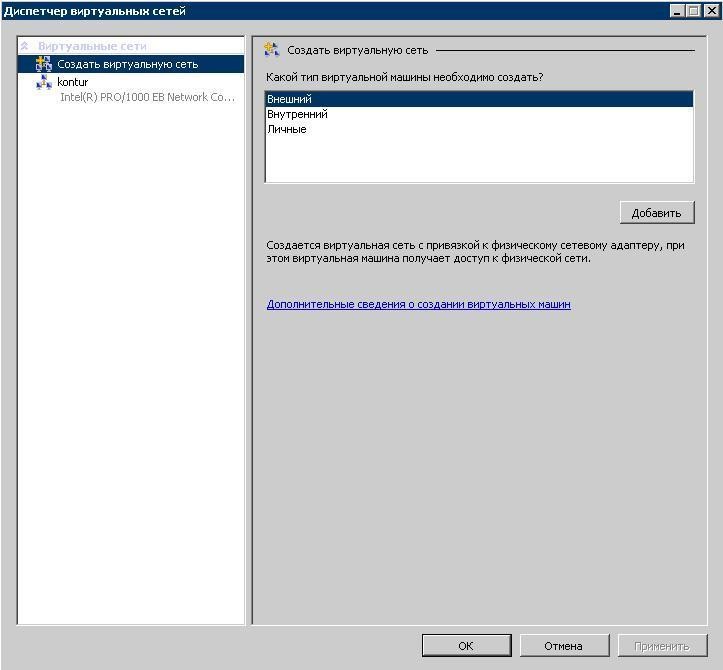
1. **Установите роль Hyper-V на сервере Windows 2008 (или выше).** Включите компьютер, дождитесь загрузки. Войдите в систему, используя учетную запись администратора. **Откройте Пуск – Диспетчер сервера – Роли – Добавить роли**. В открывшемся окне мастера установки выберите роль сервера Hyper-V и следуя инструкциям мастера произведите установку.



Установка роли Hyper-V

# Произведите сетевые настройки.

После успешной установки роли Hyper-V, произведите настройку виртуальной сети. Откройте **Пуск – Диспетчер сервера – Роли – Hyper-V**. В панели **Действия** выберите **Диспетчер виртуальной сети**. Создайте **Внешнюю** и **Внутреннюю** виртуальные сети.



Диспетчер виртуальной сети

# Создание виртуальных машин Откройте Пуск – Диспетчер сервера – Роли – Hyper-V.

* 1. Создайте виртуальную машину с фиксированным виртуальным жестким диском.
  2. Создайте виртуальную машину с динамически расширяющимся виртуальным жестким диском.
  3. Создайте виртуальную машину для гостевой операционной системы Windows 7.
  4. Измените количество оперативной памяти, процессоров, тип сетевого подключения в конфигурации виртуальной машины.
  5. Установите Integration Services для гостевой операционной системы
  6. Создайте снимок виртуальной машины.
  7. Произведите изменения в гостевой операционной системе.
  8. Отмените изменения, используйте возврат к предыдущему снимку.
  9. Выполните экспорт виртуальной машины.
  10. Измените конфигурацию виртуальной машины, измените размер диска виртуальной машины.

# Обзор System Center Virtual Machine Manager

* 1. Запустите Configuration Analyzer для тестирования системы.
  2. Установите SCVMM сервер.
  3. Установите консоль администратора SCVMM.
  4. Добавьте пользователей в группу IT Admin Support
  5. Создайте новую группу узлов в VMM
  6. Добавьте необходимые сервера Hyper-V в VMM.
  7. Измените конфигурацию виртуальной машины.
  8. Создайте шаблон виртуальной машины.
  9. Создайте несколько экземпляров виртуальных машины из полученного шаблона.
  10. Обзор библиотеки VMM.
  11. Произведите конвертирование физического сервера в виртуальное окружение.

# Лабораторная работа 2. Установка и настройка VMWare Workstation

Целью лабораторной работы является практическое освоение технологий виртуализации VMWare на примере VMWare Workstation.

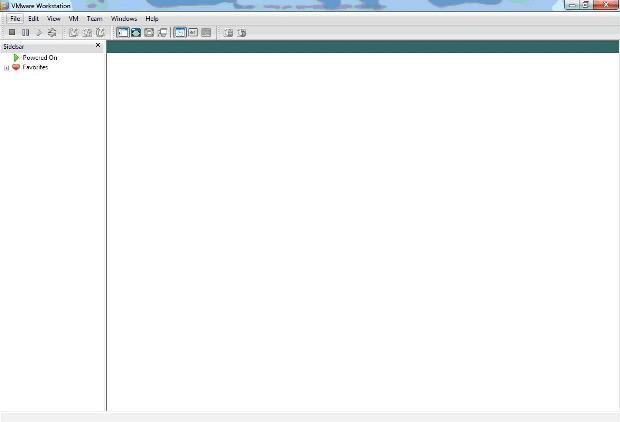
# Аппаратура и программные инструменты, необходимые для лабораторной работы

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

**Продолжительность лабораторной работы** 2 академических часа

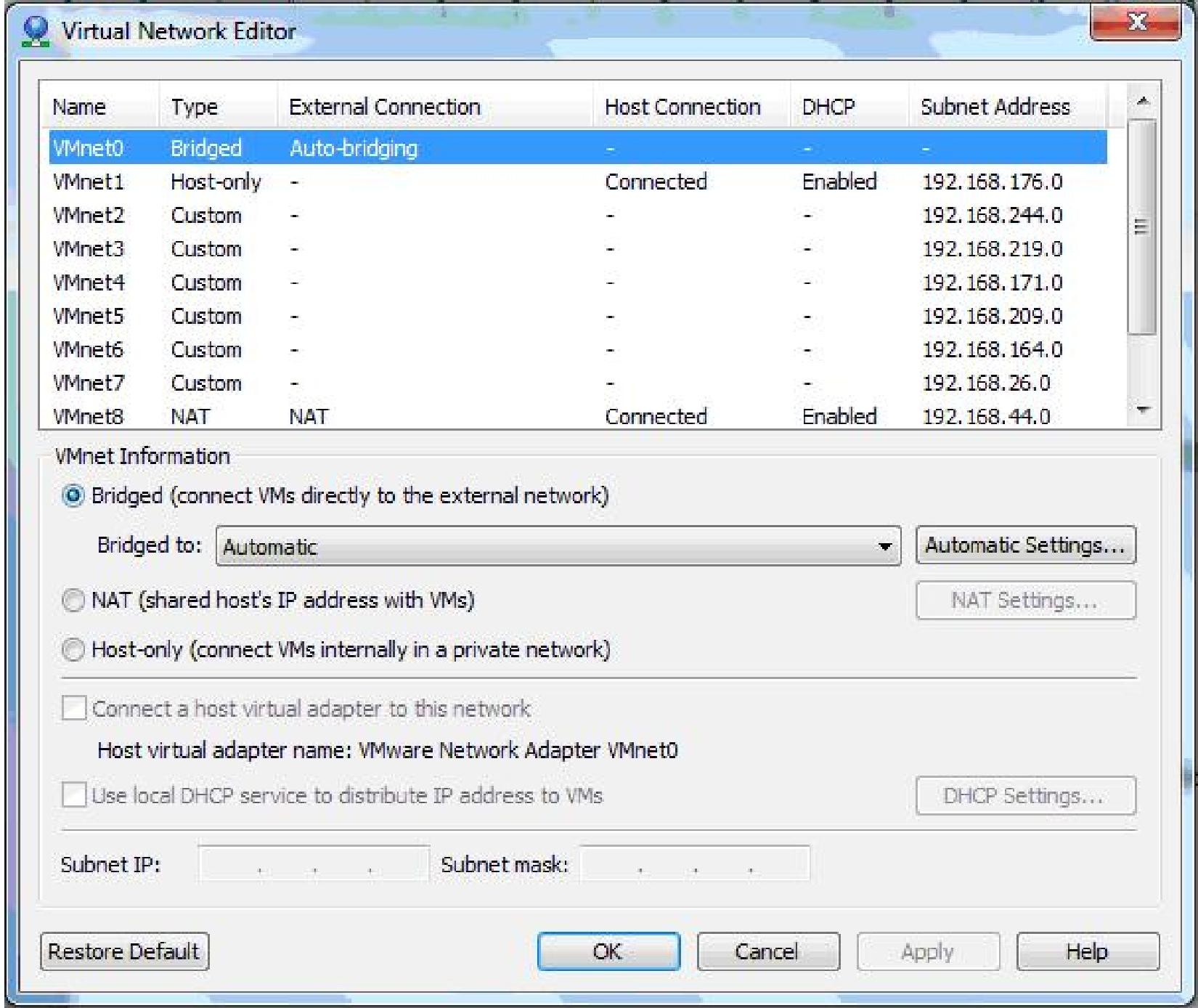
# Установите VMWare Workstation

Используя установочный дистрибутив VMWare Workstation, установите продукт на компьютер. Запустите программу, открыв Пуск – Все программы – VMWare – VMWare Workstation.



# Произведите сетевые настройки.

Откройте **Пуск – Все программы – VMWare – Virual Network Editor**. Произведите конфигурацию виртуальной сети.



# Создайте виртуальную машину для гостевой операционной системы Windows 7

В меню **File – New –Virtual Machine** создайте новую виртуальную машину. Установите операционную систему Windows 7 в виртуальной машине.

# Установите VMWare Tools

После установки операционной системы Windows 7, установите инструменты VMWare в меню **VM – Install VMWare Tools**

# Создайте снимок виртуальной машины.

Создайте снимок виртуальной машины в меню **VM – Snapshot – Take Snapshot**.

1. **Произведите изменения в гостевой операционной системе**. Выполните произвольные изменения в виртуальной машине,

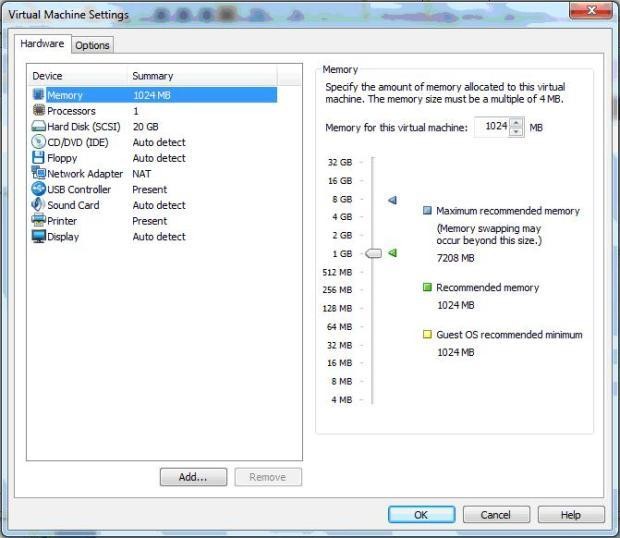
скопируйте на рабочий стол несколько ярлыков, создайте несколько папок.

# Отмените изменения, использовав возврат к предыдущему снимку.

Выполните возврат к предыдущему снимку виртуальной машины в меню **VM – Snapshot – Revert to Snapshot** и выберите предыдущий снимок.

# Измените конфигурацию виртуально машины.

Произведите изменение конфигурации виртуальной машины в меню **VM – Settings**. Увеличьте количество оперативной памяти, количество процессоров. Увеличьте размер жесткого диска. Создайте дополнительный жесткий диск.



## Лабораторная работа 3. Создание первого Windows Azure приложения

Целью лабораторной работы является практическое освоение создания приложений Windows Azure.

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

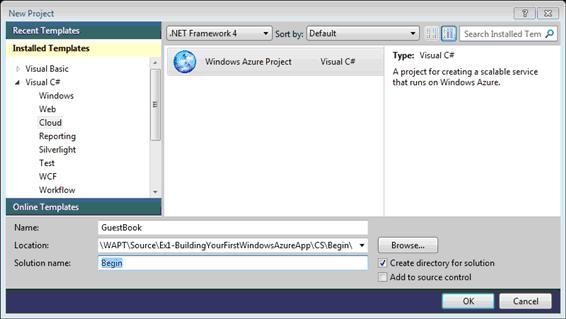
1. Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.
2. Доступ к сети Интернет.
3. Наличие аккаунта Windows Azure.

**Продолжительность лабораторной работы** 2 академических часа

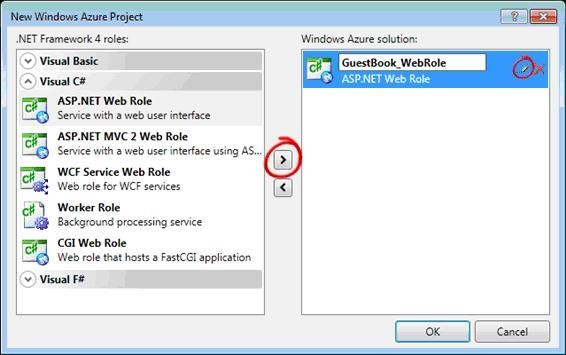
## Создание проекта в Visual Studio

### Откройте меню Пуск | Все программы| Microsoft Visual Studio 2010 | Microsoft Visual Studio 2010.

1. В меню File выберите New и затем **Project**.
2. В диалоговом окне **New Project** разверните узел **Visual C#** и в списке **Installed Templates** выберите **Cloud**.
3. 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.

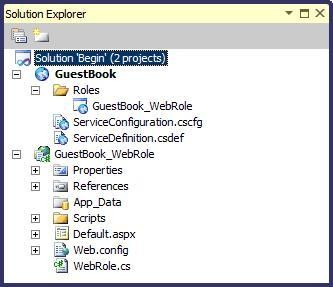


1. В диалоге **New Cloud Service Project** разверните узел **Visual C#** и выберите **ASP.NET Web Role**. Переместите выбранную роль в проект. Выберите роль в проекте, нажмите на изображение "карандаш" и введите имя **GuestBook\_WebRole**. Нажмите **OK** для завершения.



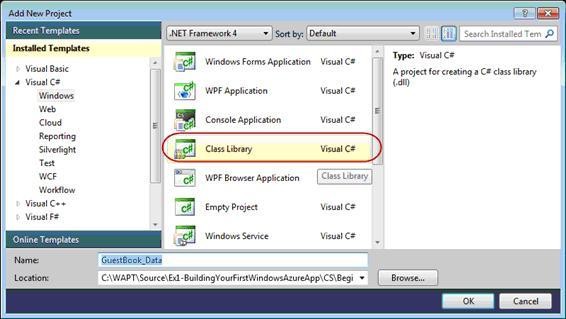
Добавление ролей в проект

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



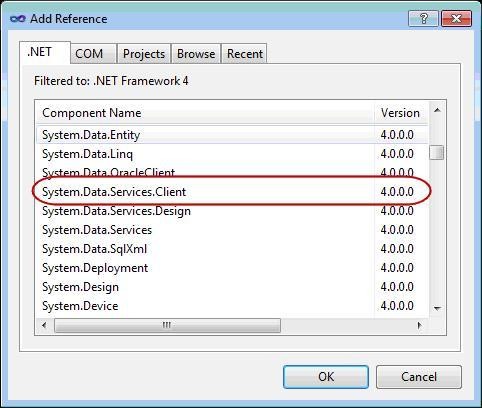
## Создание модели данных для элементов в Table Storage

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

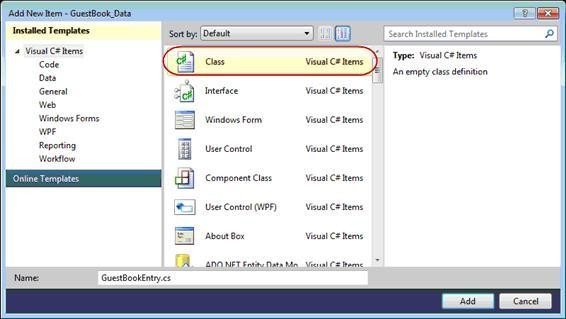


Создание библиотеки классов

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* ;
2. 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 конструктор 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**

## Лабораторная работа 4. Развертывание приложения Windows Azure

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

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

1. Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.
2. Доступ к сети Интернет.
3. Наличие аккаунта Windows Azure.

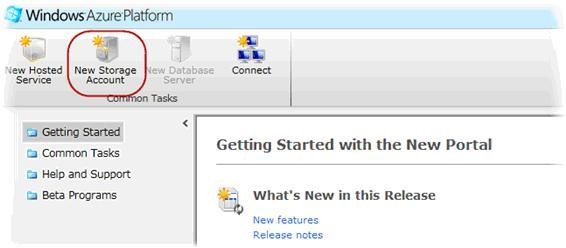
**Продолжительность лабораторной работы** 2 академических часа

## 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.



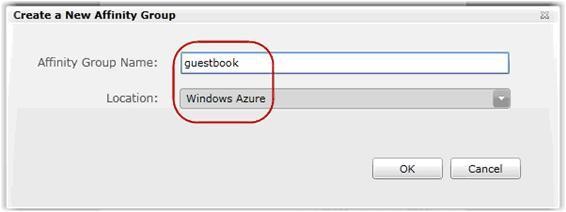
1. Введите имя для Вашего **Storage Account**.



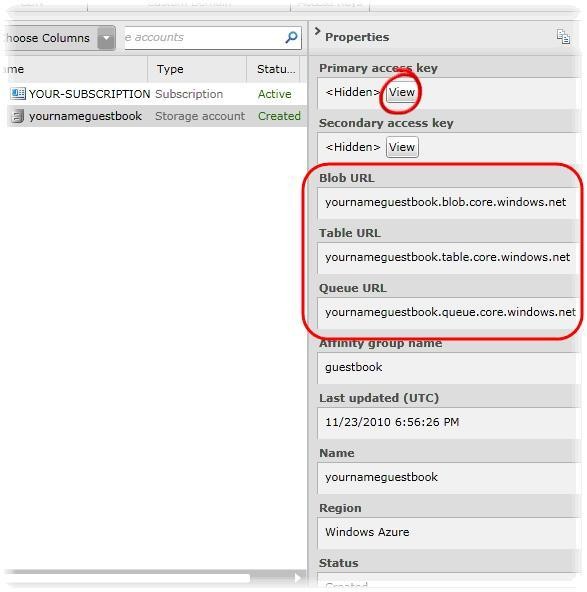
1. Выберите настройку **Create or choose an affinity group** и затем выберите из списка **Create a new affinity group**



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



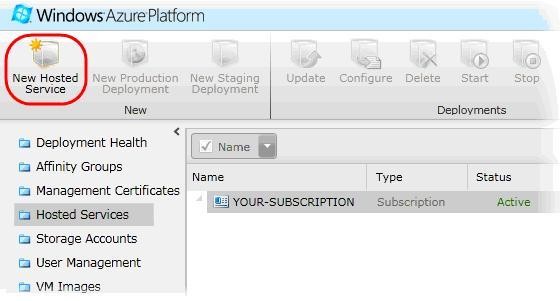
1. Вернитесь в диалог **Create a New Storage Account** и нажмите **Create** для создания нового **Storage Account**.



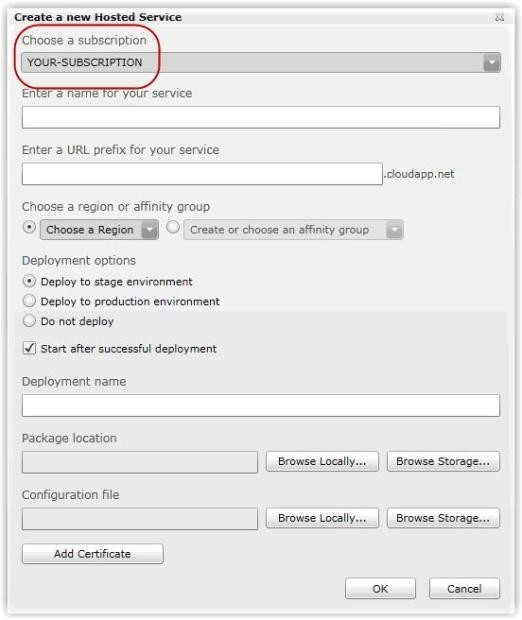
1. Нажмите кнопку **View**. В диалоге **View Storage Access Keys** нажмите кнопку **Copy to Clipboard** следующую за **Primary Access Key**.



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

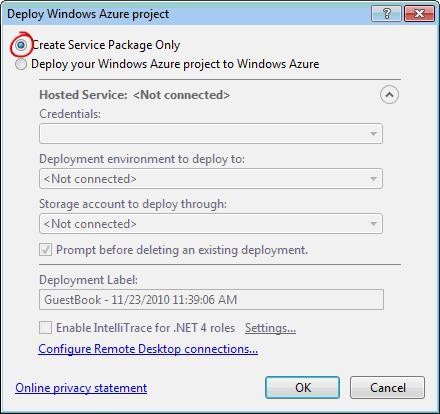


## Развертывание приложения на портале Windows Azure Platform

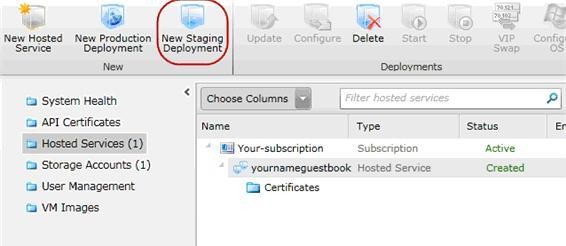
1. Откройте меню **Пуск | Все программы| Microsoft Visual Studio 2010** и запустите **Microsoft Visual Studio 2010** от имени администратора.
2. В меню **File** выберите **Open** и затем **Project/Solution**. Откройте файл **Begin.sln** проекта **Ex3-WindowsAzureDeployment**
3. Для изменения конфигурации 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**.

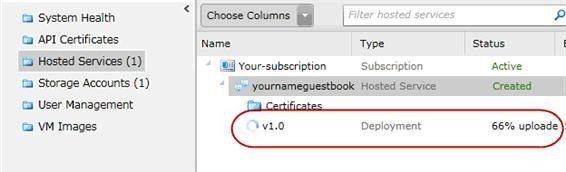


1. Переключитесь обратно в окно браузера, где открыт портал управления **Windows Azure**.
2. На портале выберите сервис, который Вы создали в предыдущем упражнении и нажмите **New Staging Deployment** на панели инструментов.



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



## Лабораторная работа 5. Работа с Blob

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

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

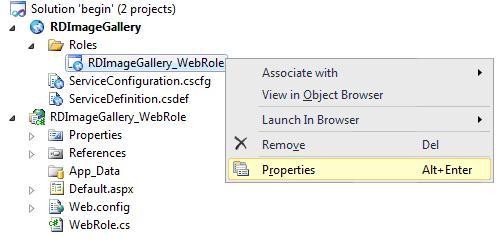
1. Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.
2. Доступ к сети Интернет.
3. Наличие аккаунта Windows Azure.

**Продолжительность лабораторной работы** 2 академических часа

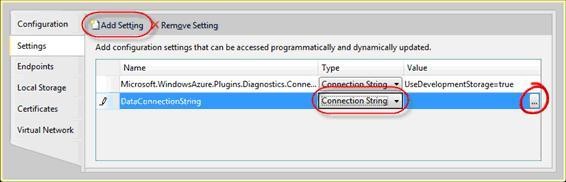
## Получение Blob данных из хранилища

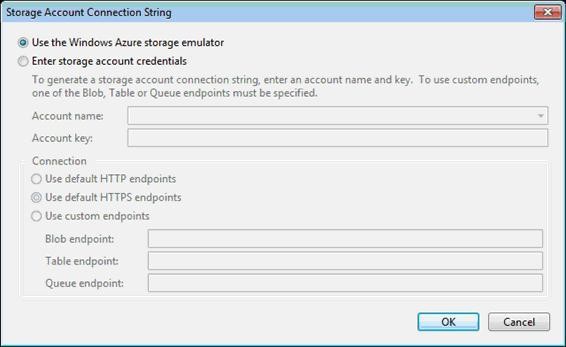
### Откройте меню Пуск | Все программы| Microsoft Visual Studio 2010 | Microsoft Visual Studio 2010.

1. В меню **File** выберите **Open** и затем **Project/Solution**. Откройте файл проекта **ExploringWindowsAzureStorageVS2010\Source\Ex02- WorkingWithBlobs\begin\CS\begin.sln**
2. В **Solution Explorer**, в проекте **RDImageGallery** нажмите правой кнопкой по узлу **RDImageGallery\_WebRole** и выберите **Properties**

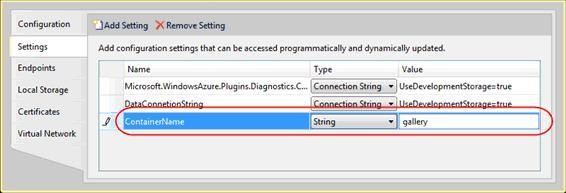


1. На закладке **Settings** создайте **ConnectionString** с именем **DataConnectionStrin**. Выберите **Use development storage**





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.GetConfigurationSettingValu e("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()

{

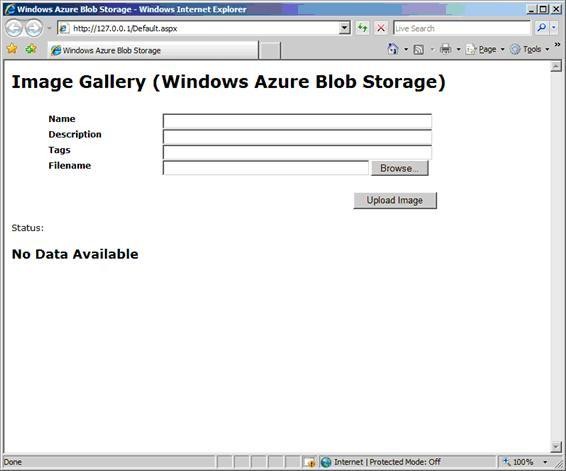
BlobListingDetails.All images.DataBind();

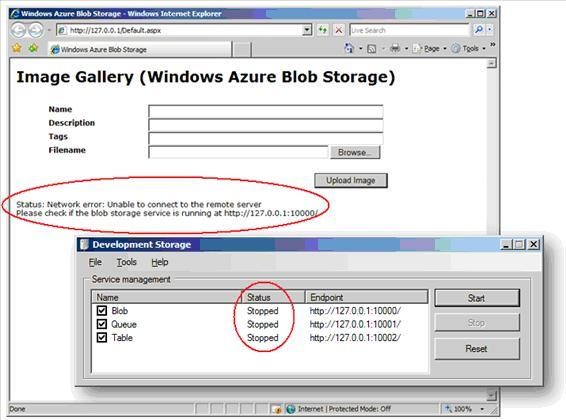
}

UseFlatBlobListing = true, BlobListingDetails =

});

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





## 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();

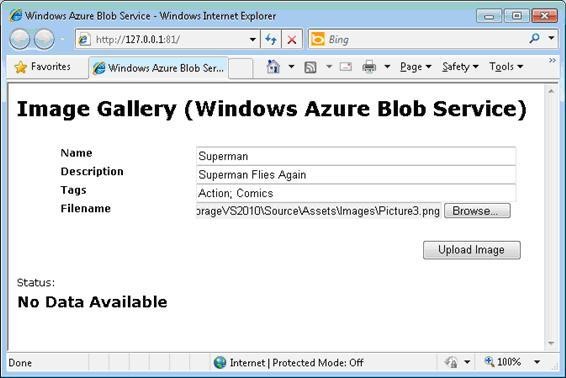
}

else

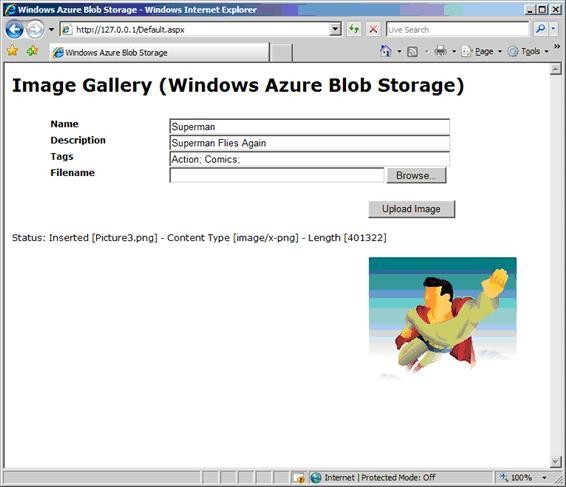
status.Text = "No image file";

}

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

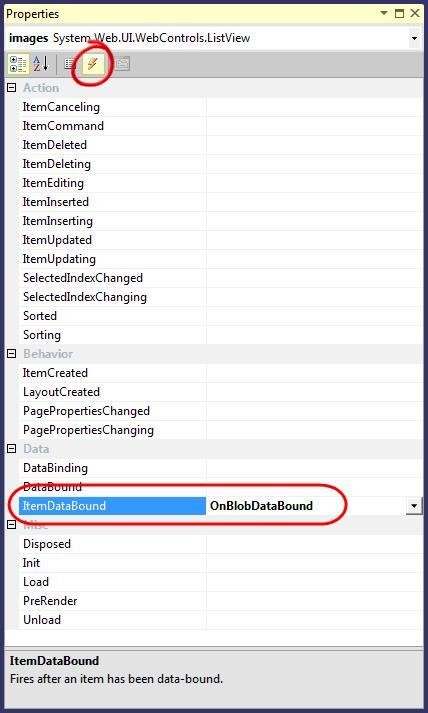


1. Нажмите **Upload Image** для публикации изображения в веб приложении



## Извлечение метаданных для Blob в хранилище

1. Откройте **Default.aspx** в режиме **Design**, выберите контрол **imagesListView** 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";

LinkButton;

}

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

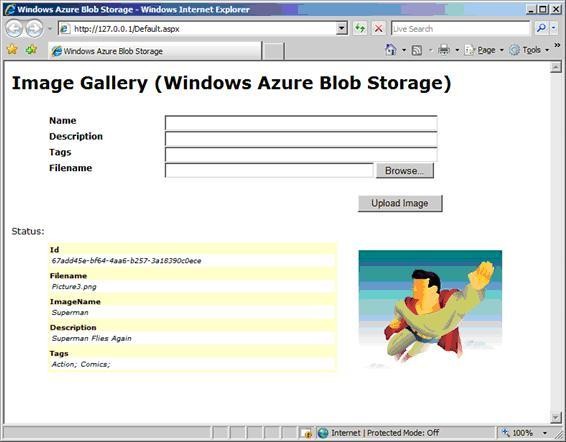
}

}

}

}

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



## Лабораторная работа 6. Работа с Tables

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

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

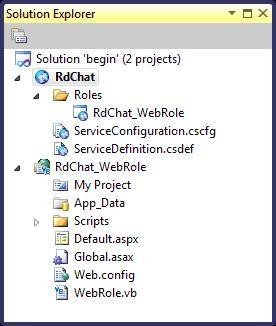
1. Настольный или портативный компьютер, поддерживающий виртуализацию, операционная система Microsoft Windows XP, Vista, Windows 7.
2. Доступ к сети Интернет.
3. Наличие аккаунта Windows Azure.

**Продолжительность лабораторной работы** 2 академических часа

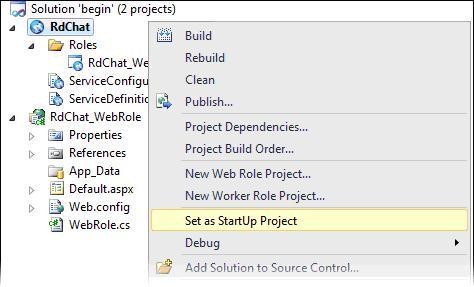
## Настройка Storage Account Settings

### Откройте меню Пуск | Все программы| Microsoft Visual Studio 2010 | Microsoft Visual Studio 2010.

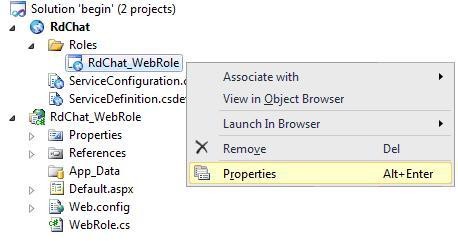
1. В меню **File** выберите **Open** и затем Project/Solution. Откройте файл проекта ExploringWindowsAzureStorageVS2010\Source\Ex01- WorkingWithTables\begin\C#\begin.sln



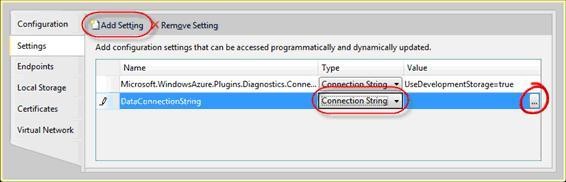
1. Убедитесь что проект **RdChat\_WebRole** запускается по умолчанию.



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



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



1. Сохраните изменения

## . 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**.

## Лабораторная работа 7. Работа в Windows Live

1. Откройте в Internet Explorer страницу <http://live.com/>
2. Пройдите аутентификацию использую Live ID, при необходимости зарегистрируйтесь
3. В верхнем меню наведите выберите **Hotmail**
4. Создайте учетную запись **Hotmail**
5. Подключите имеющийся у Вас почтовый аккаунт
6. В верхнем меню наведите курсор на **Hotmail** и выберите **Calendar**
7. Укажите часовой пояс
8. Создайте новый календарь в меню **New | Calendar**
9. Создайте новое событие в меню **New | Event**
10. В верхнем меню наведите курсор на **Messenger**, выберите **Contacts**
11. Создайте новый контакт в меню **New**
12. В меню Office создайте документы Word, Excel, PowerPoint
13. Дважды выполните изменения в файле.
14. Откройте историю версий, выполните возврат к предыдущей версии документа.
15. Измените настройки общего доступа к файлу.
16. В меню **Photos | Your albums** создайте новый альбом, нажав **Create album**
17. Загрузите несколько фотографий в созданный альбом
18. Откройте общий доступ к альбому в меню **Photos | Share photos**
19. В меню **Windows Live** выберите **SkyDrive**
20. Создайте учетную запись в **SkyDrive**
21. Создайте новую папку в **SkyDrive** через меню **New**
22. Загрузите несколько файлов в созданную папку
23. Откройте общий доступ к папке, выбрав ее и в меню **Share** выберите **Edit Permissions**
24. В меню **Profile** выберите **connect**
25. При наличие, подключите имеющиеся учетные записи **Facebook, YouTube, MySpace, LinkedIn**

## Лабораторная работа 8. Работа в Office Live

1. Откройте в Internet Explorer страницу <http://workspace.officelive.com/>
2. Нажмите **Вход** (справа вверху), используйте **Live ID** для входа.
3. Выберите **Новая рабочая область** для создания новой рабочей области
4. Перейдите в созданную рабочую область
5. В меню **Создать** выберите **Примечание**, заполните необходимую информацию
6. В меню **Создать** выберите **Список задач**, заполните необходимую информацию
7. Добавьте новую строку и новый столбец
8. В меню **Создать** выберите **Список контактов**, заполните необходимую информацию
9. Добавьте несколько контактов
10. В меню **Создать** выберите **Список событий**, заполните необходимую информацию
11. Добавьте несколько событий
12. Подключите список в Microsoft Outlook, нажав в меню на кнопку **Подключение к Outlook**
13. Произведите экспорт в Microsoft Excel нажав в меню на кнопку **Экспорт в Excel**
14. Измените настройки общего доступа в меню **Общий доступ**
15. Создайте несколько комментариев.