Лабораторная #2.

Введение в работу с Azure App services.

В этой работе мы познакомимся с основами разработки PaaS приложений. Реализуем собственное ASP.Net приложение и развернём его в Azure.

Что конкретно мы сделаем:

* Удостоверимся в том, что приложение работает локально и познакомимся с его функционалом
* Создадим Web Application на Azure Portal
* Развернём наше приложение в облаке
* Убедимся в том, что оно доступно извне

Примечание!

В данной работе, а также в ряде последующих, нам потребуется использовать уникальные имена для сервисов в Azure (для Web App, Storage Account, …). В некоторых случаях инструменты предлагают сгенерировать эти имена автоматически, но увы – не всегда. Поэтому, заранее придумайте такое уникальное имя, например по схеме:

AzureWorkshop<ВашеИмя><ДатаРождения/ИлиПростоНаборЦифр>

# Задание 01. Ознакомление с приложением

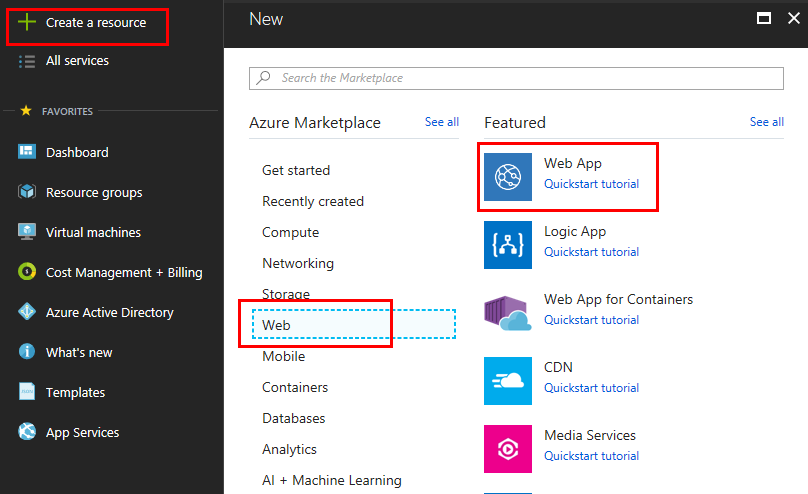
В этом задании мы ознакомимся с шаблоном приложения ASP.Net MVC, которое мы будем переиспользовать и модифицировать в последующих задачах.

1. Перейдите в директорию *01.AppService*
2. Откройте файл *Epam.AzureWorkShop.Labs.sln*. У вас откроется окно Visual Studio.
3. Нажмите Ctrl+F5 для того, чтобы запустить приложение и убедиться в том, что оно работает.
4. Попробуйте добавить заметки с изображениями.

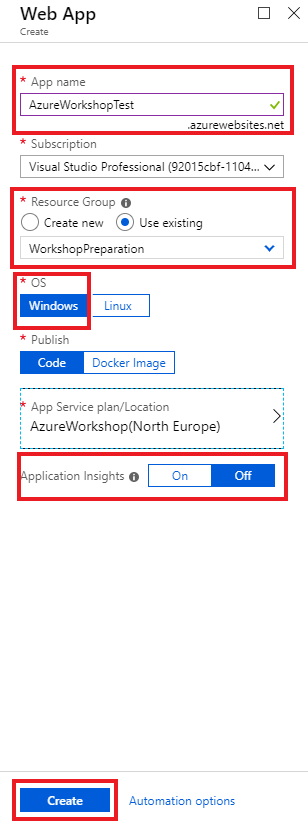
# Задание 02. Создание Web приложения на Azure Portale

Теперь нам нужно создать сервис для размещения нашего приложения в Azure. Это можно делать прямо из Visual Studio, при настройке параметров деплоя. Однако мы воспользуемся настройкой через Azure Portal, т.к. этот способ представляется более общим и более конфигурируемым.

1. На Azure Portal нажмите кнопку **Create a resource,** затем выберите **Web > WebApp**

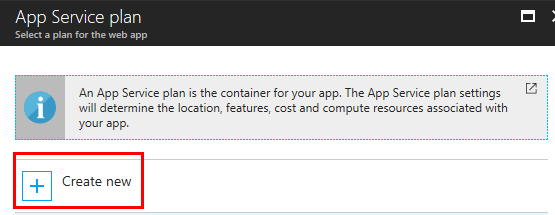


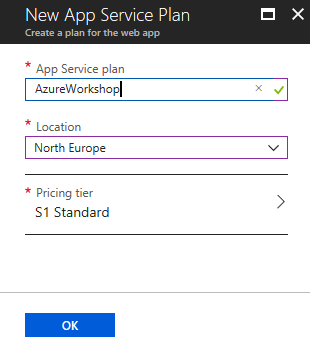
1. В окне создания нового приложения укажите следующие параметры:



|  |  |
| --- | --- |
| Настройка | Значение |
| App name | [Ранее выбранное уникальное имя] |
| Resource Group | [Название ресурсной группы созданной ранее] |
| OS | Windows |
| Application Insights | Off |

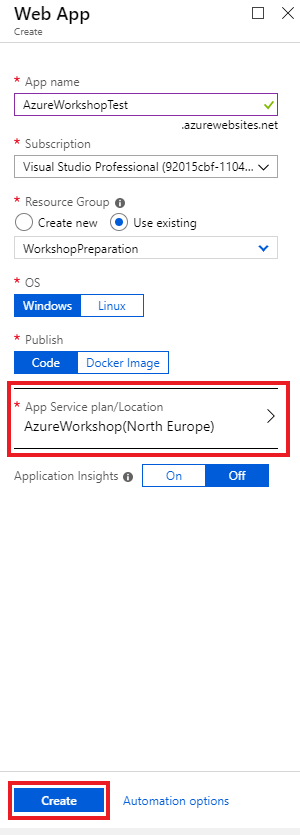
1. Нажмите на секцию «**App Service plan/location**», в открывшемся окне **Create new,** заполните параметры нового плана и нажмите **Ok**



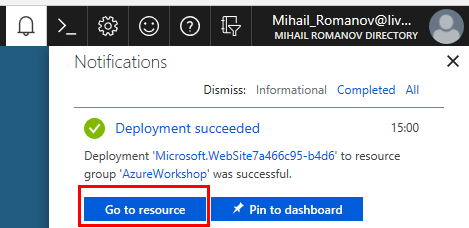


|  |  |
| --- | --- |
| Настройка | Значение |
| App Service plan | AzureWorkshop |
| Location | North Europe |
| Pricing tier | S1 Standard |

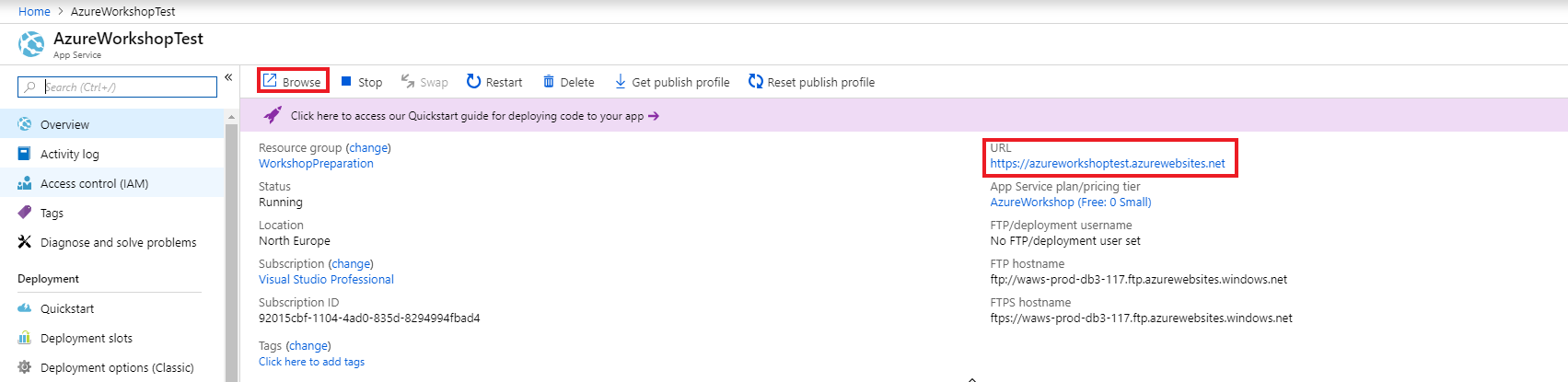
1. Вернитесь к окну создания приложения. Убедитесь, что только что созданный план выбран и нажмите **Create**



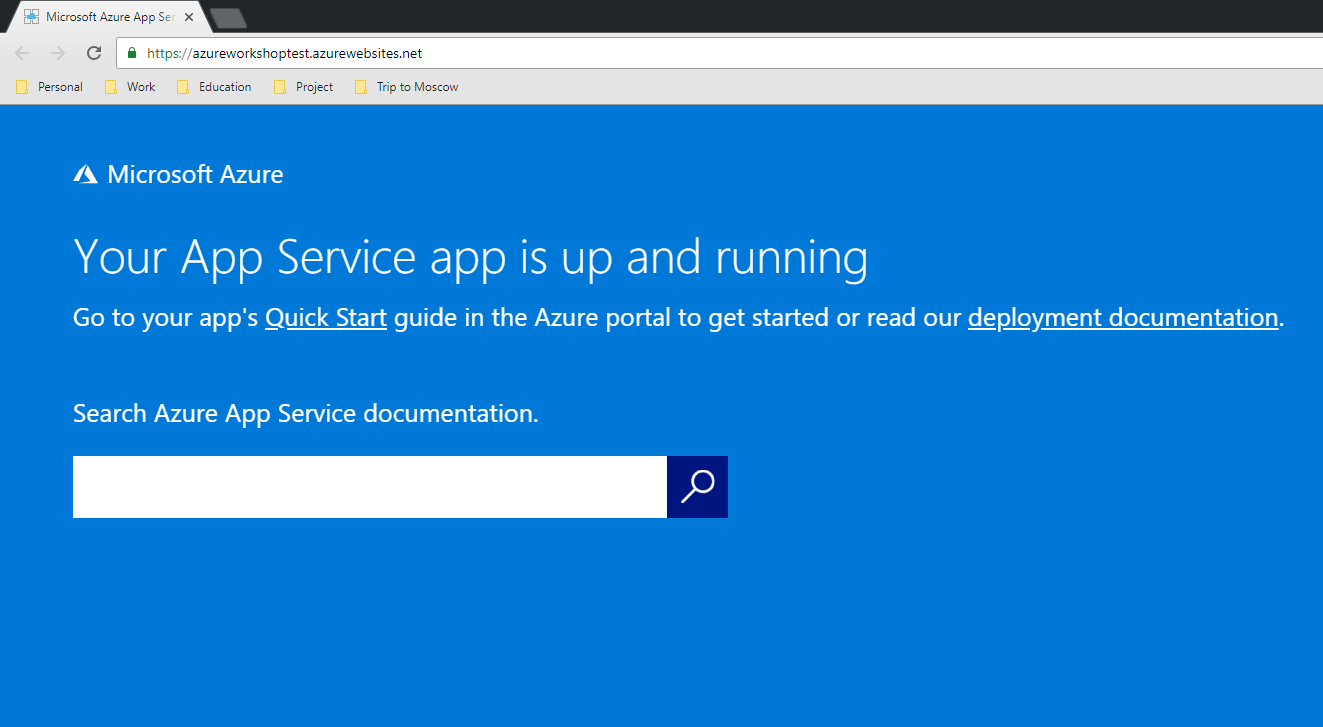
1. Дождитесь появления в панели нотификаций уведомления об окончании деплоя и нажмите кнопку **Go to resource**



1. В окне параметров приложения нажмите кнопку закрепления ярлыка на рабочем столе (в правом верхнем углу ), а затем ссылку в параметре **URL** или на кнопку **Browse**.



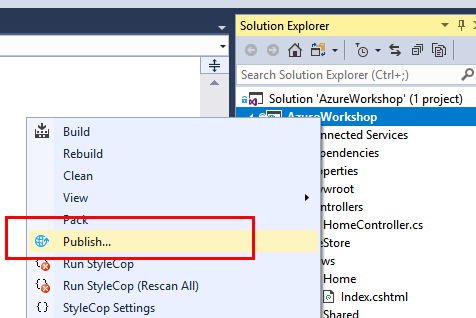
1. В открывшемся окне или вкладке браузера вы увидите стандартную заставку созданного приложения



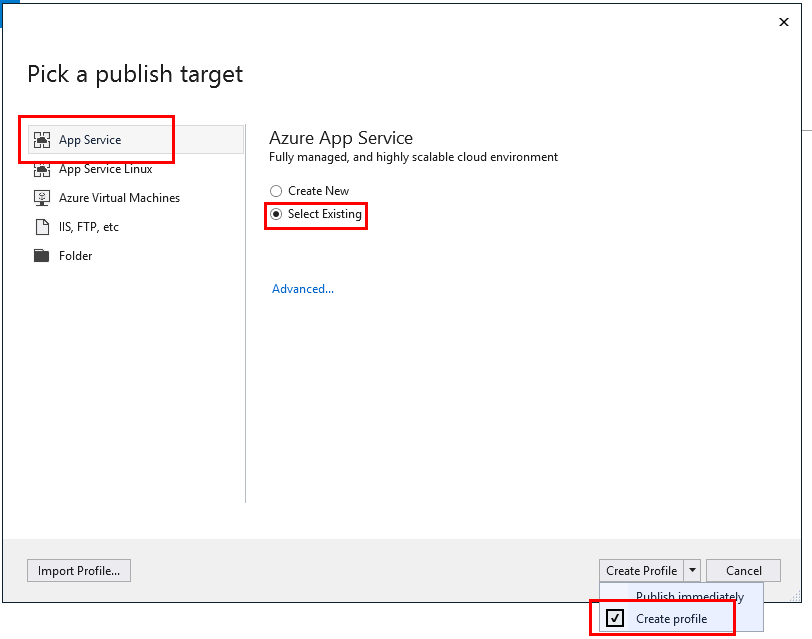
# Task 03. Deploy Web application to Azure

Следующим шагом мы настроим деплой нашего приложения в облако. Azure App Services поддерживает несколько способов деплоя, но из всех мы остановимся на варианте MS Deploy как на одном из наиболее простых с точки зрения настройки, а также как «из коробки» поддерживаемый в Visual Studio

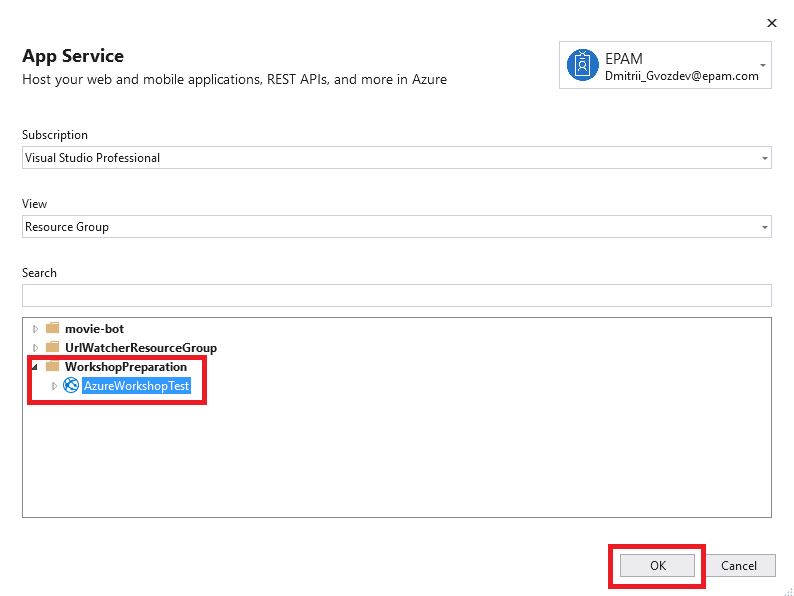
1. Откройте Visual Studio с вашим приложением и в контекстном меню проекта выберите пункт **Publish**



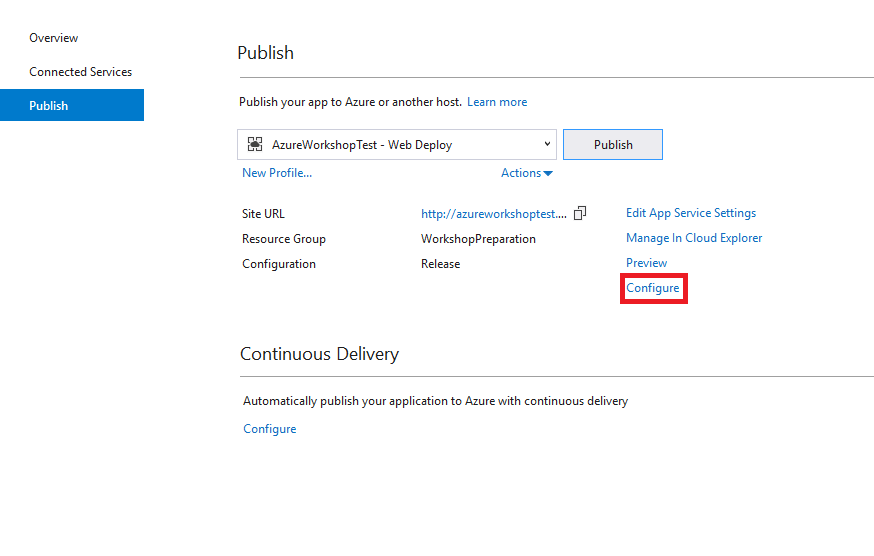
1. В диалоге настройки публикации выберите **App Service,** **Select Existing** и нажмите **Create Profile**



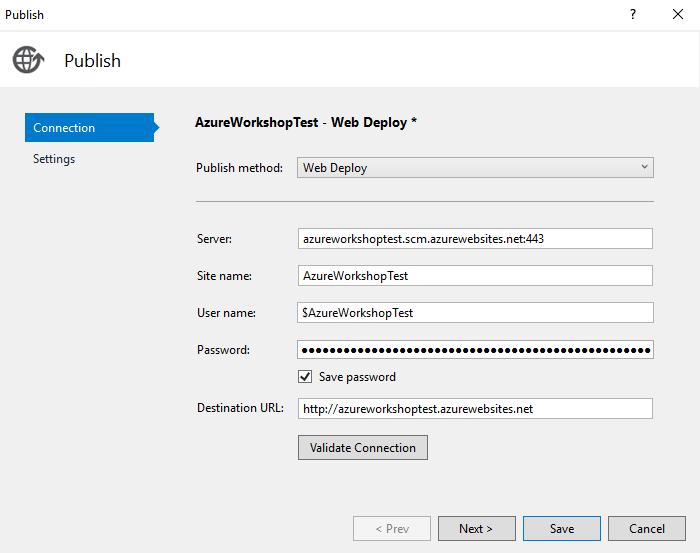
1. В диалоге выбора цели укажите созданный ранее App Service и нажмите Ok



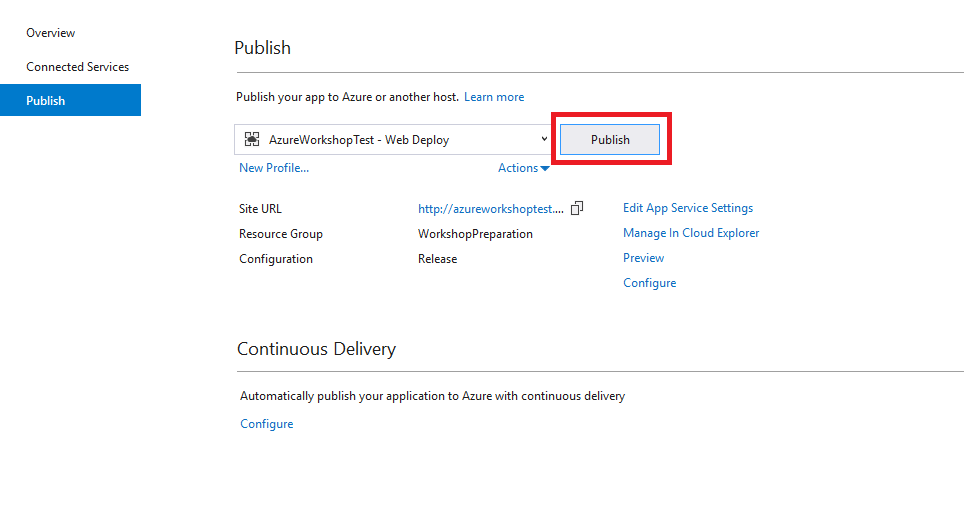
1. В открывшемся окне профиля публикации нажмите **Configure**



1. Изучите настройки деплоя установленные по умолчанию. Особого внимания заслуживает закладка Settings, на которой, в частности указывается:
   * в какой конфигурации (Debug/Release) происходит деплой приложения
   * параметры .Net Core publishing (framework depended/self-contained, платформа компиляции, версия)
   * будут ли во время деплоя удаляться «лишние» файлы (те, которых нет в пакете)



1. Закройте окно настроек и в окне профиля нажмите **Publish**



1. Дождитесь окончания деплоя и открытия окна браузера. Убедитесь, что приложение запускается без ошибок и возвращает форму загрузки файлов

