## 注册：

账号：simonangelfong@gmail.com

信用卡：8955

# 入门

<https://azure.microsoft.com/zh-cn/get-started/?v=17.39>

## 入门Video：

1. 界面介绍
2. 创建、管理Windiows VM
3. 创建并部署web apps

应用VS，将web app部署在az

使用命令栏+github，部署web app

1. 创建sql数据库

可以在vm上安装sql server

可以使用az sql database service

创建az sql database service

防火墙与白名单来让sql server management访问

## 创建Windows虚拟机

### 创建途径：

* portal
* PowerShell
* Azure CLI

### Create Windows virtual machine in the Azure Portal

#### Sign in to Azure

#### Create virtual machine in portal

protal => virtual machines => Create virtual machine

##### Basics Setup

**Project details:**

Subscription: 免费试用

Resource group : iResourceGroup\_Demo

**Instance details:**

Virtual machine name: iVM-Demo

Region : East US

Size: Standard B1s - 1 vcpu, 1 GiB memory (US$10.22/month)

**Administrator account:**

Username: simonangelfong

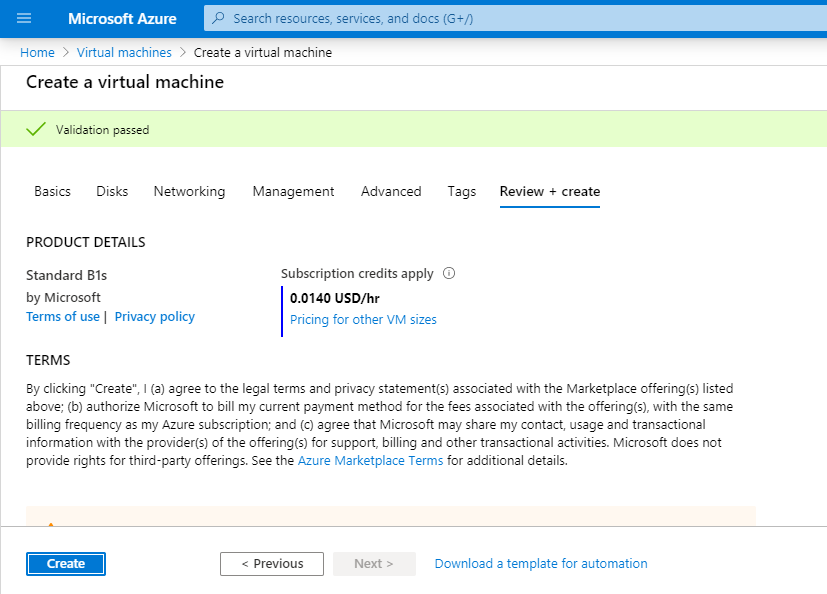
Password: SimonFong!23

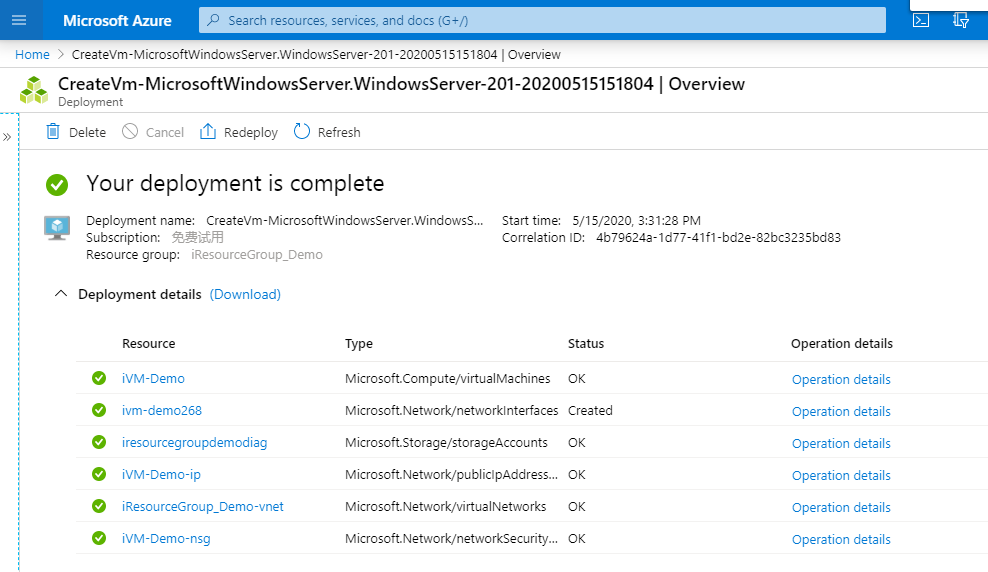
**Inbound port rules：**

Public inbound ports : Allow selected ports

Select inbound ports : HTTP(80), RDP(3389)

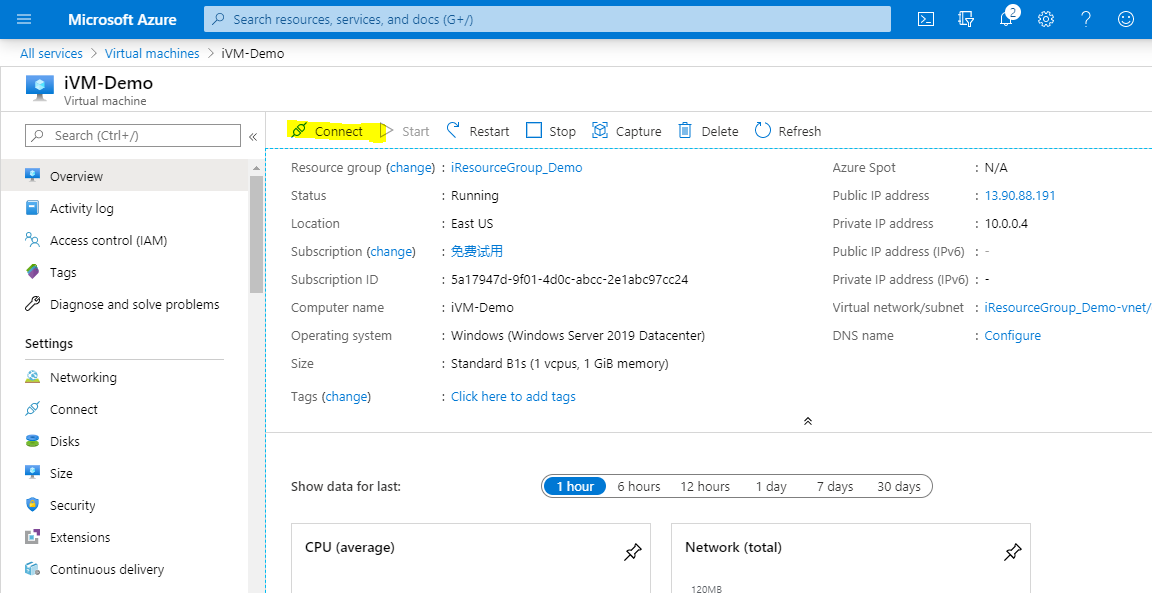
**"Review + create"(CLICK THE BUTTON)**



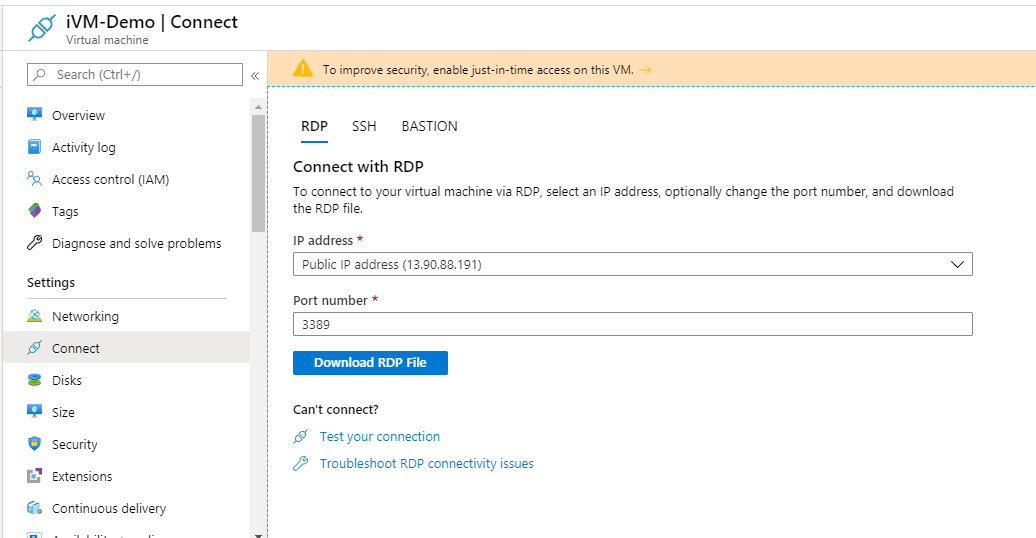


#### Create connection to virtual machine

1. Click "Connect" Button, Choose "RDP" option.

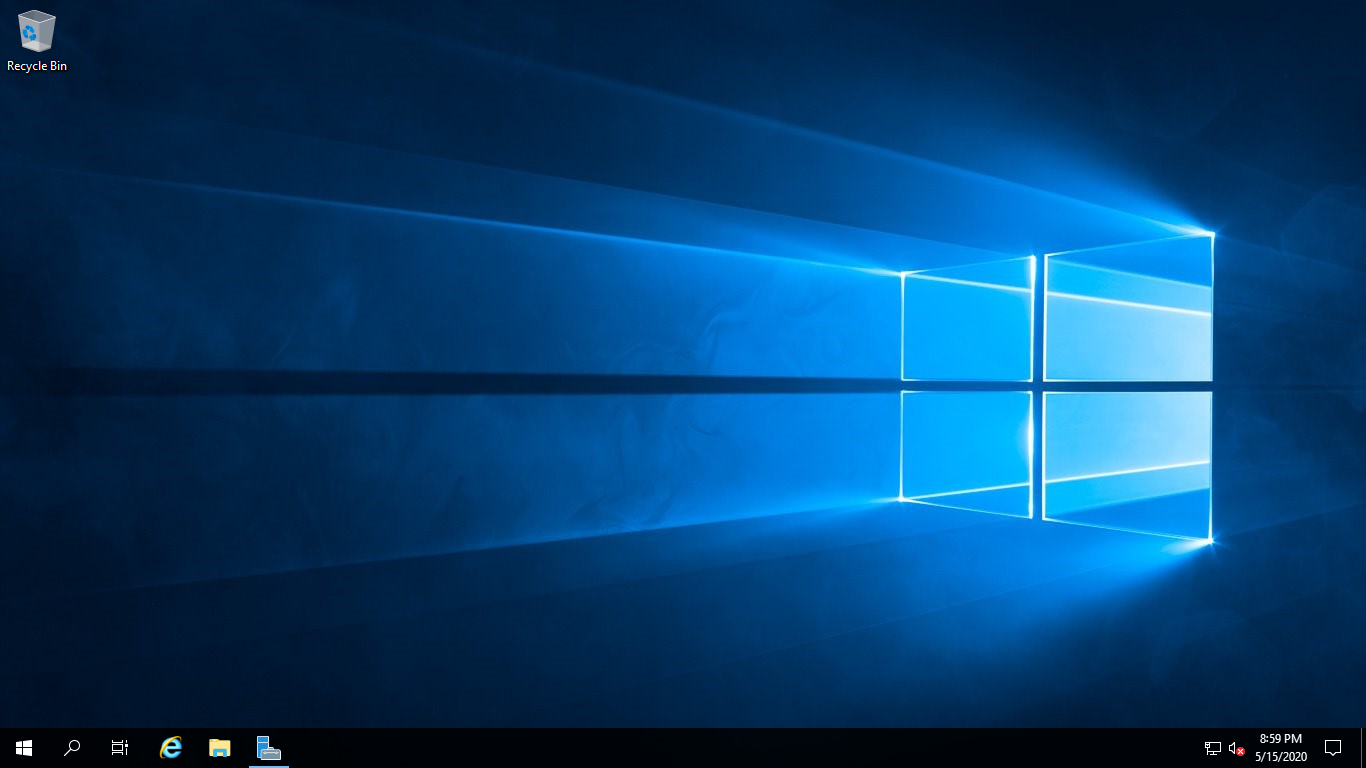


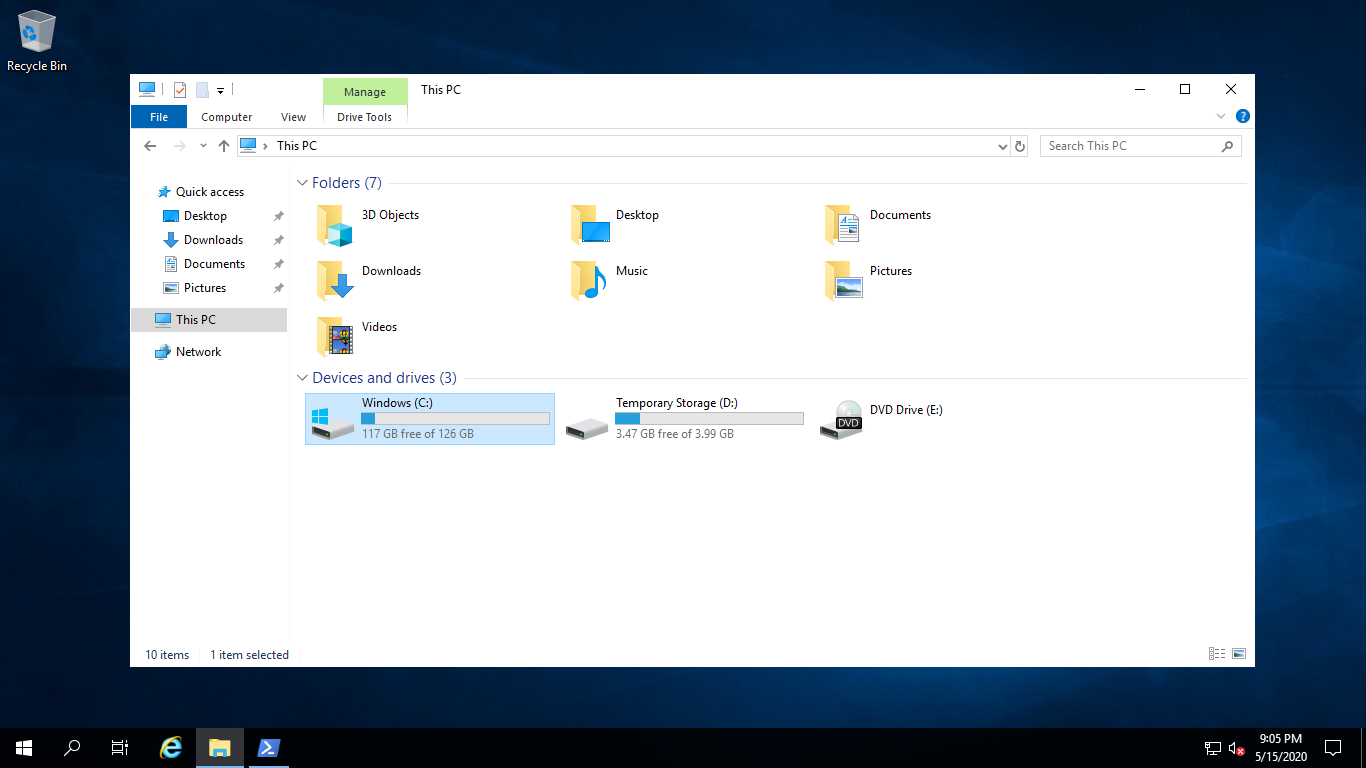
1. Download RDP File



1. Open RDP file on a local machine, input username and password.

Type username as localhost\<username>.



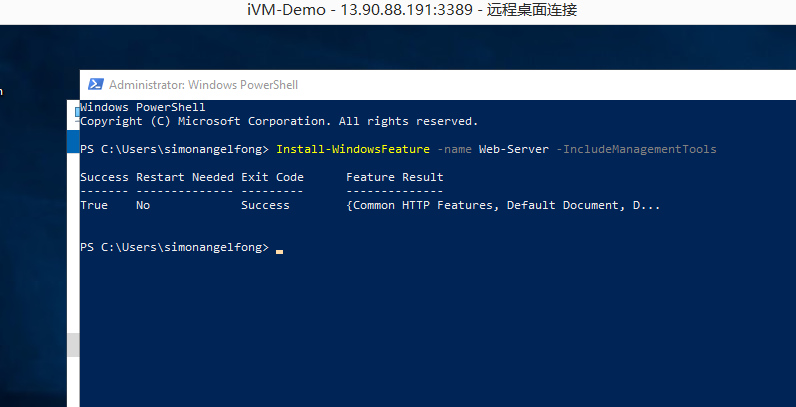


#### Install web server in VM

To see VM in action, install IIS web server.

On VM's PowerShell:

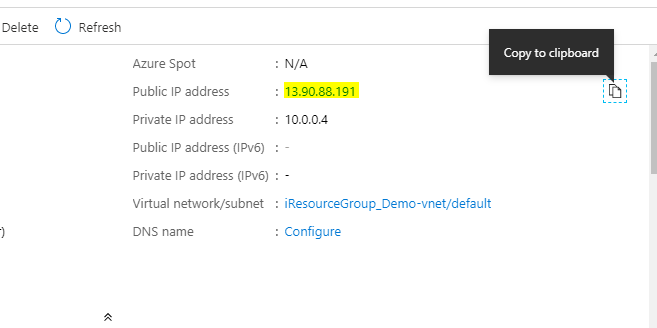
Install-WindowsFeature -name Web-Server -IncludeManagementTools



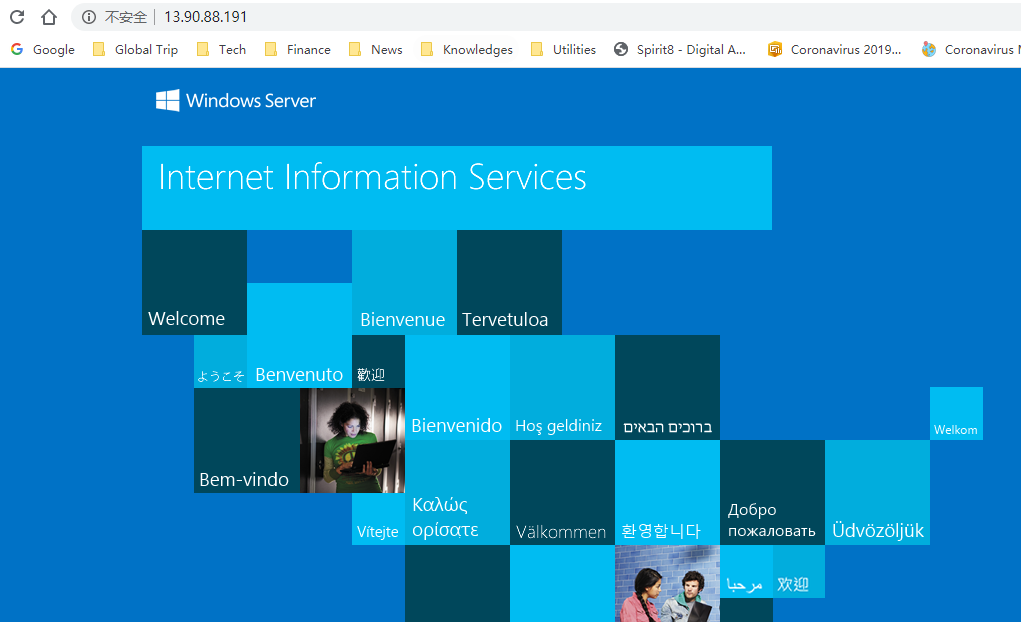
Then close RDP.

#### View the IIS welcome page

In the portal, copy the ip address:



Paste it into a browser tab:



### Furhter about Virtual Machine

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/tutorial-manage-vm>

## 创建web应用

Azure App Service is

* HTTP-Based service
* for hosting web application, REST API, moblie back end

先使用VS创建web app，再通过VS部署在Azure上。

### Create an ASP.NET web app(Visual Studio)

1. Create a new project = > ASP.NET Web Application (.NET Framework), => Configure your new project => MVC template => No Authentication => Create
2. Debug > Start Without Debugging

### Publish web app

1. Publish => App Service => Create profile
2. Sign in Azure
3. App Service Setting

Name: iAppServiceDemo

Resource Group: iResourceGroup\_Demo

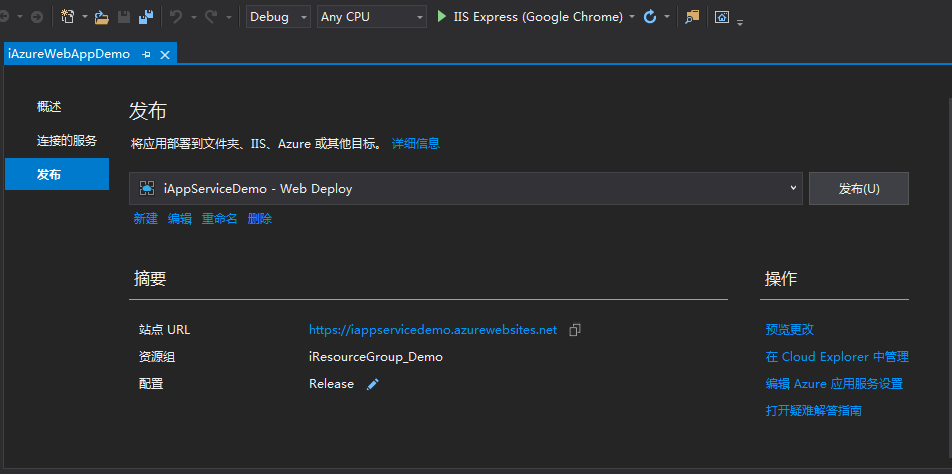
Create New Hosting Plan:

Hosting Plan: iAppServicesPlan

Location: East US

Size: Free

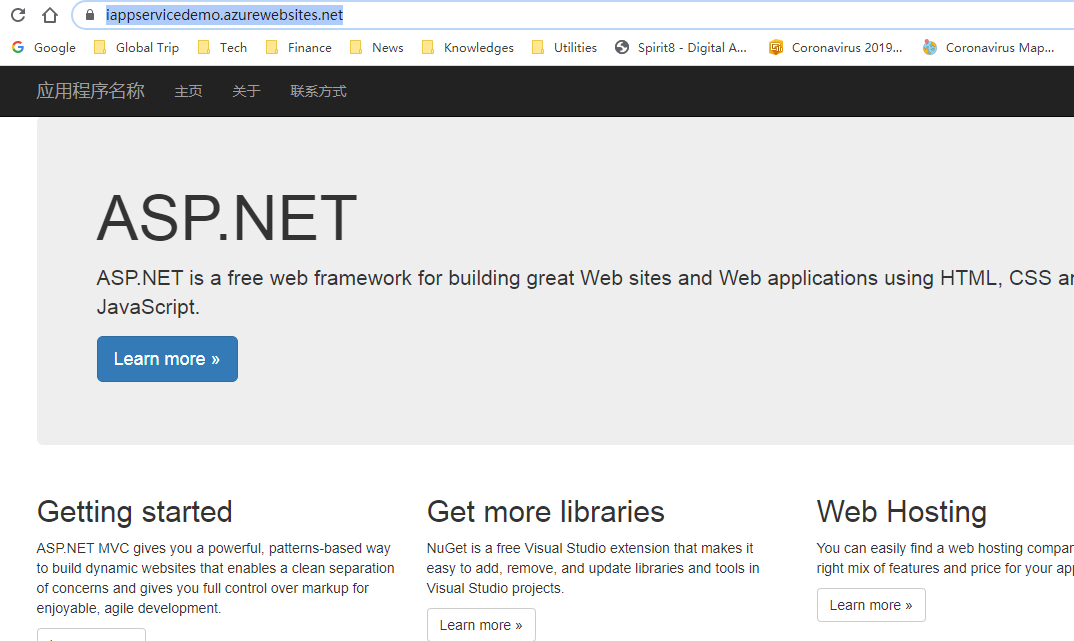
"Create"(Click button)

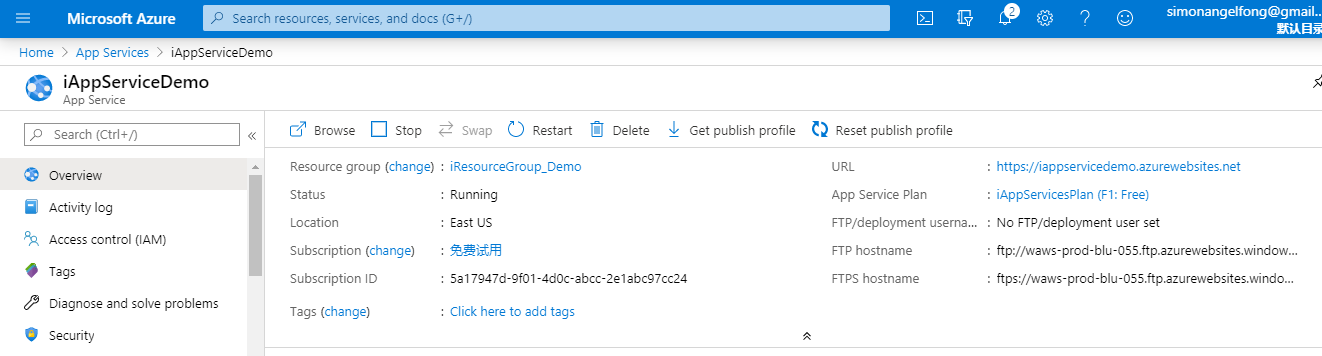


1. Publish

Bowser tab url: http://<app-name>.azurewebsites.net

<https://iappservicedemo.azurewebsites.net/>



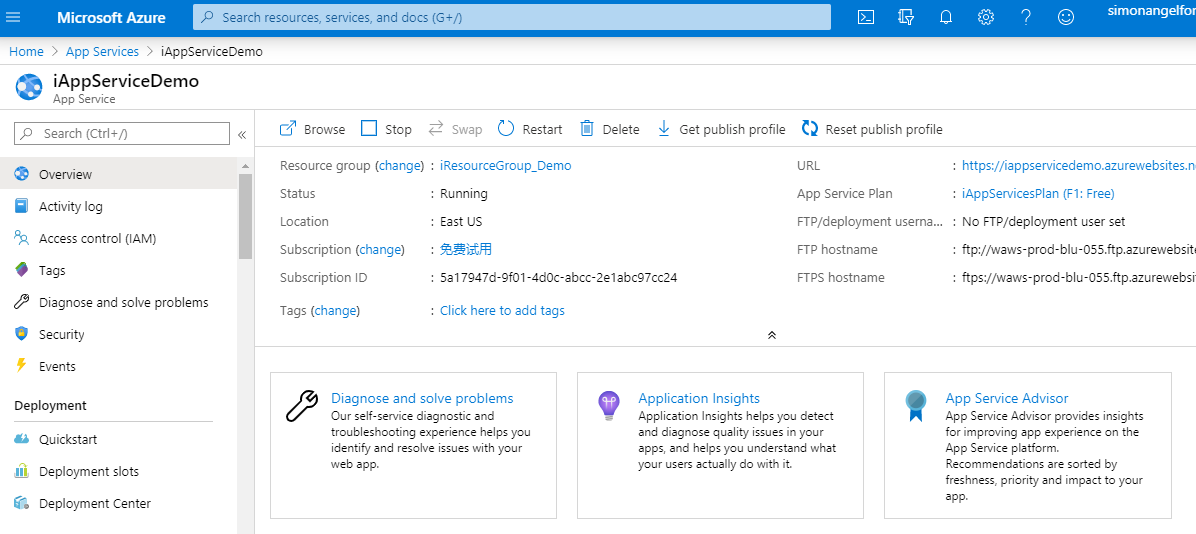


### Update the app and redeploy

1. Update app VS.
2. Redeploy.



### Manage the Azure app



## 创建Azure SQL数据库（单一数据库）

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-single-database-get-started?tabs=azure-portal>

### Create a single database

1. Portal => Azure SQL => Add

**Select SQL deployment option**

select "**SQL databases**" => "**Create**"

1. Configuration settings

**Basics Tab**

**Project details:**

Resource group : iResourceGroup\_SQLDemo

**Database details:**

Database name: iDatabaseDemo

Server: =>

Create new

Server name: isqlserver

Server admin login: azureuser

Password: Azsql!23

Location: (US) East US

**Networking Tab**

Connectivity method: Public endpoint

**Firewall rules**

Add current client IP address: Yes

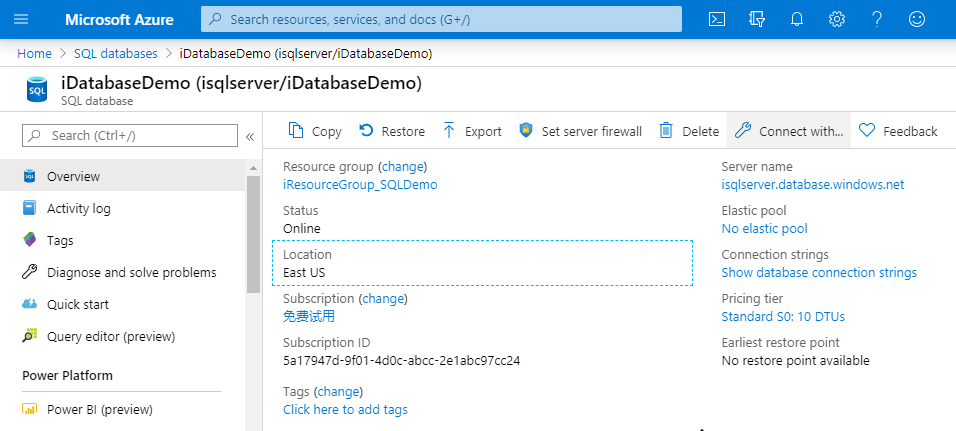
**Additional settings Tab**

**Data source**

Use existing data: Sample

" Review + create "

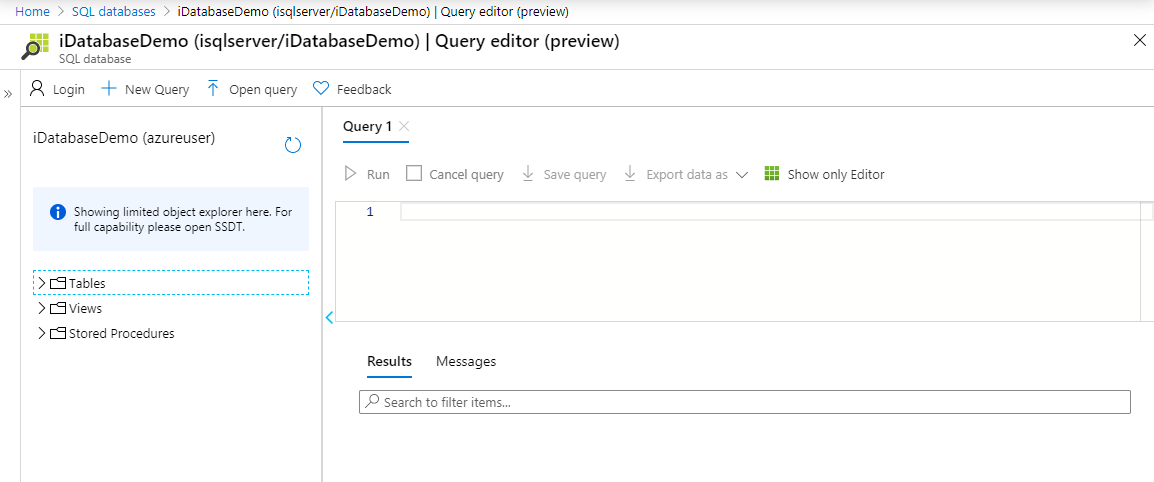
1. "Create"



### Query the database

1. portal => SQL databases => select database
2. SQL Database page => Query editor(preview)

login SQL server with username and password



In **Query editor:**

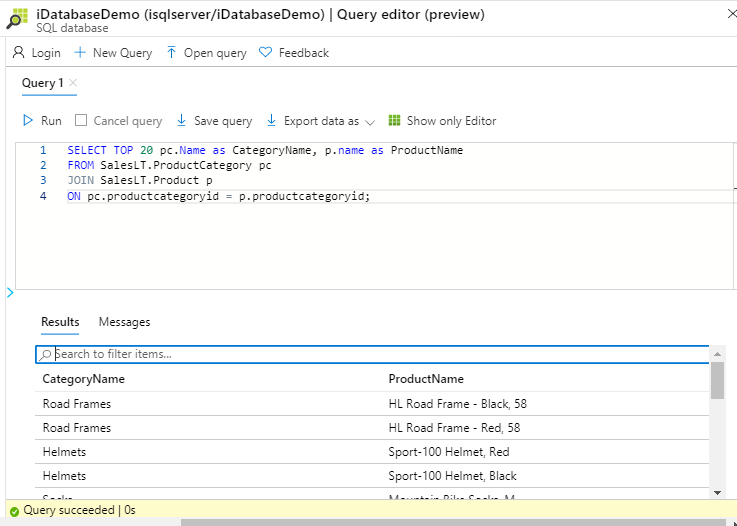
SELECT TOP 20 pc.Name as CategoryName, p.name as ProductName

FROM SalesLT.ProductCategory pc

JOIN SalesLT.Product p

ON pc.productcategoryid = p.productcategoryid;

"Run"



## 创建无服务器的应用(Azure Functions)

使用VSC创建Python functions，响应HTTP请求。

在本地创建并测试代码，然后在Azure部署。

### Configure environment（python）

Python3.7以上

VSC

Python extension for VSC

Azure Functions extension for VSC

### Create local project

1. Azure: Functions => Deploy to function app
2. Create setting:

Select folder for project

Select a language for your function project: Python

Select inerpreter: python 3.8.1

Select a template for your project's first function: HTTP trigger

Provide a function name: HttpExample

Authorization level: Anonymous

Select how you would like to open your project: Add to workspace

### Run the function locally

问题：VSCode 本地运行function需要安装azure function core tools

解决：

1、下载并安装node.js到本机，安装node.js的同时，npm会伴随安装。

2、在windows的cmd中确认node、npm是否安装成功：

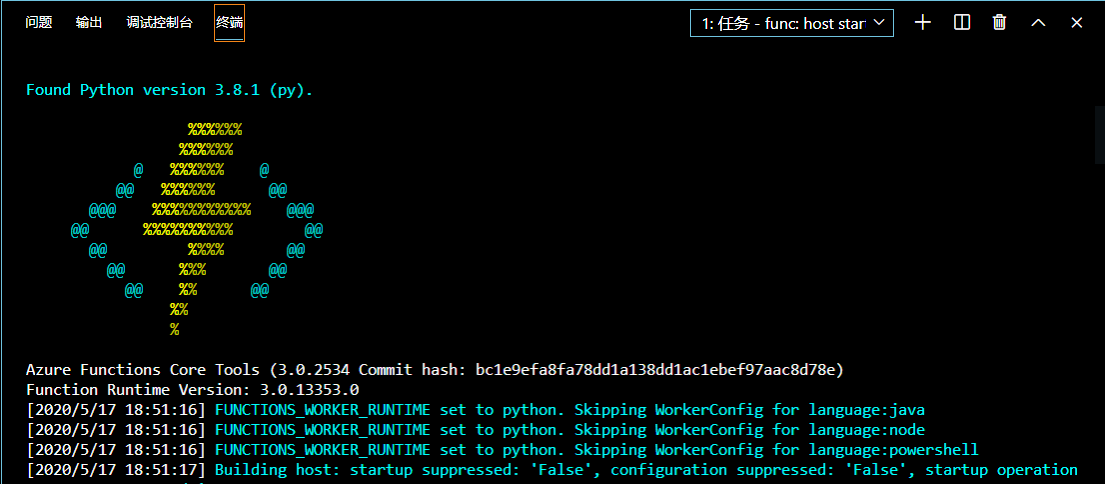
node -v 查询node.js的版本

npm -v 查询npm的版本

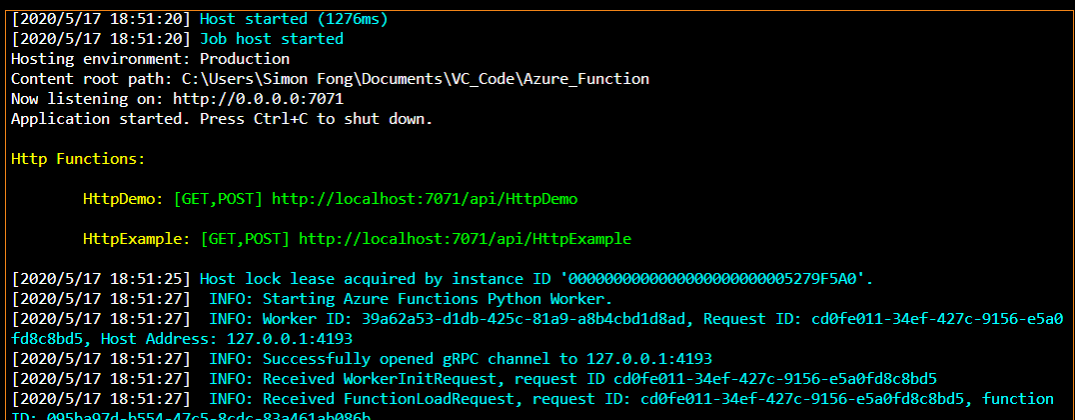
3、在vscode中的命令栏中，输入命令：

npm i -g azure-functions-core-tools@3 --unsafe-perm true

After installed azure function core tool, press F5 to debug. Information show in Terminal:

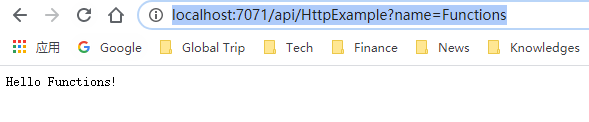


Url of function show:



Copy url to browser tab:

<http://localhost:7071/api/HttpExample?name=Functions>



Ctrl + C to stop debugger

### Publish the project to Azure

1. Azure: Functions area => Deploy to function app
2. Configuration setting:

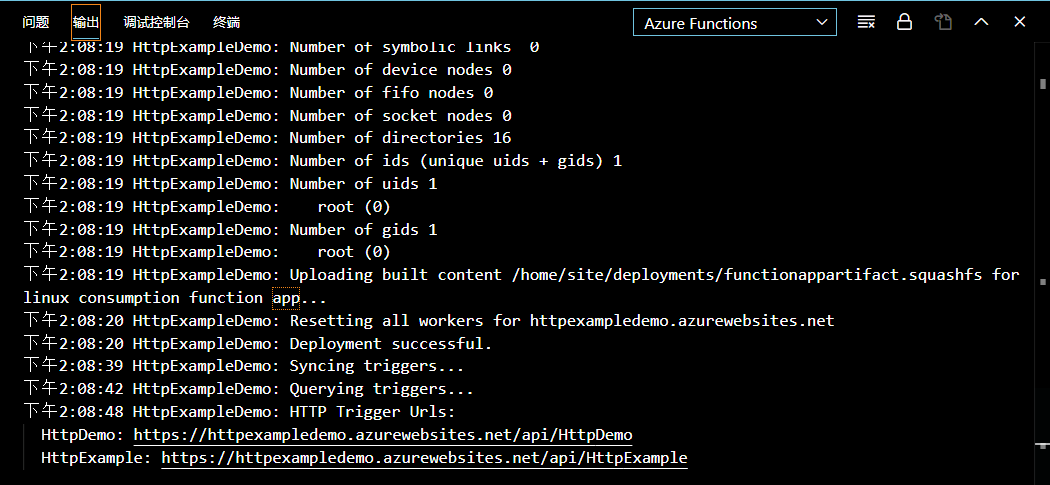
Select Function App in Azure: + Create new Function App

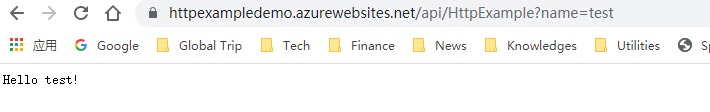
Enter a globally unique name for the function app: HttpExampleDemo

Select a runtime: Python 3.8

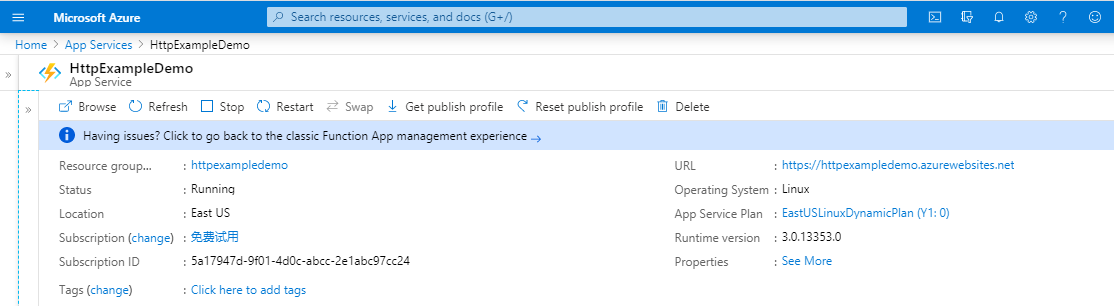
Select a location for new resources: East US

1. When completed, the following Azure resources are created in your subscription





Manage Azure Function



## 存储和传输数据与应用(Azure Storage services)

### Data services：

* Azure Blobs: A massively scalable object store for text and binary data. Also includes support for big data analytics through Data Lake Storage Gen2.
* Azure Files: Managed file shares for cloud or on-premises deployments.
* Azure Queues: A messaging store for reliable messaging between application components.
* Azure Tables: A NoSQL store for schemaless storage of structured data.
* Azure Disks: Block-level storage volumes for Azure VMs.

#### Blob storage

storing massive amounts of unstructured data, such as text or binary data

Blob storage is ideal for:

* Serving images or documents directly to a browser.
* Storing files for distributed access.
* Streaming video and audio.
* Storing data for backup and restore, disaster recovery, and archiving.
* Storing data for analysis by an on-premises or Azure-hosted service.

Link: <https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blobs-introduction>

#### Azure Files

network file shares, multiple VMs can share the same files with both read and write access

scenarios:

#### Queue storage

store and retrieve messages

up to 64 KB in size

Queues are generally used to store lists of messages to be processed asynchronously.

#### Table storage

Azure Table storage is now part of Azure Cosmos DB.

#### Disk storage

a virtual hard disk (VHD)

### Create an Azure Storage account

1. Portal => Storage accounts => Add => Create new
2. Configuration Setting:

**Basic Tab:**

**Project details**

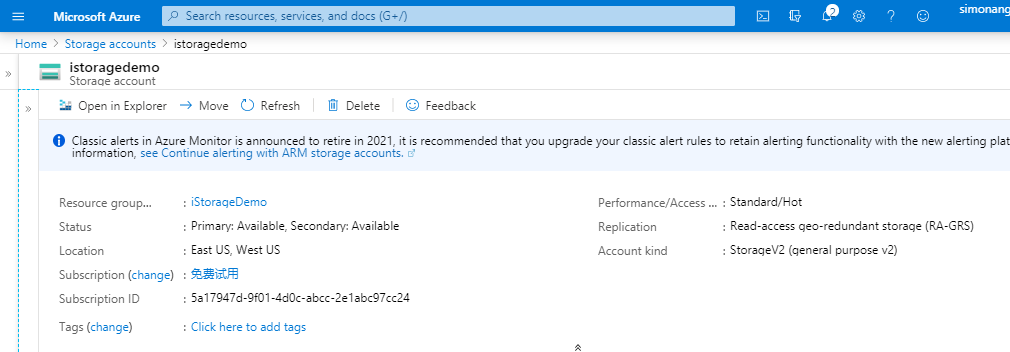
Resource group: iStorageDemo

**Instance details**

Storage account name: istoragedemo

Location: (US) East US

1. "Review + Create" => "Create"



### Upload, download, and list blobs with the Azure portal

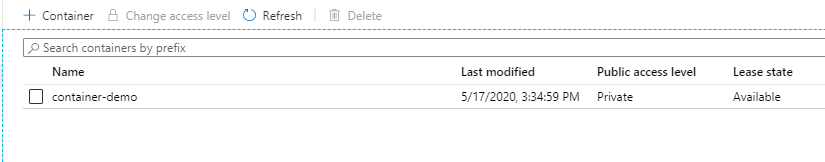
#### Create a container

1. Portal => storage account => Containers => + Container

New container

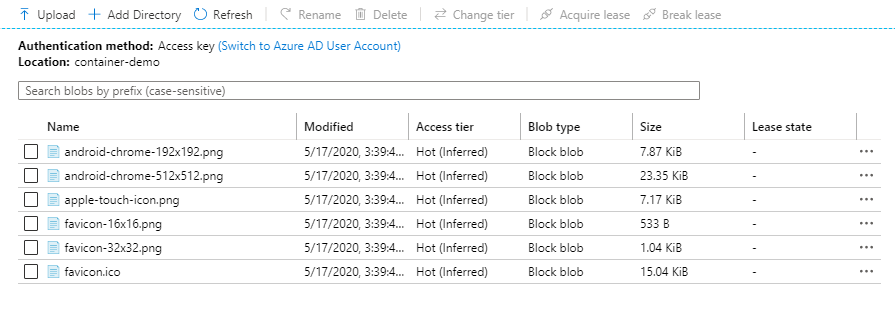
Name: container-demo

Public access level: Private(no anonymous access)



#### Upload a block blob

1. Container|<container-demo> => Upload
2. select file, "Upload"



#### Download a block blob

1. Container|<container-demo>
2. select file, "Download"

# 教程：

资源：

<https://docs.microsoft.com/zh-cn/learn/azure/?tab=tab-start-a-learning-path>

## 使用AZ部署网站

<https://docs.microsoft.com/zh-cn/learn/paths/deploy-a-website-with-azure-app-service/>

创建带有SQL数据库的asp.net app

<https://docs.microsoft.com/en-us/azure/app-service/app-service-web-tutorial-dotnet-sqldatabase>

## 数据库

Create a serverless database

https://docs.microsoft.com/en-us/azure/sql-database/sql-database-serverless#create-new-database-in-serverless-compute-tier

Quickstart: Use Visual Studio Code to connect and query an Azure SQL Database

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-connect-query-vscode>

Quickstart: Use Python to query an Azure SQL database

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-connect-query-python?tabs=windows>

Azure Queue Storage queue

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-add-output-binding-storage-queue-vs-code?pivots=programming-language-javascript>