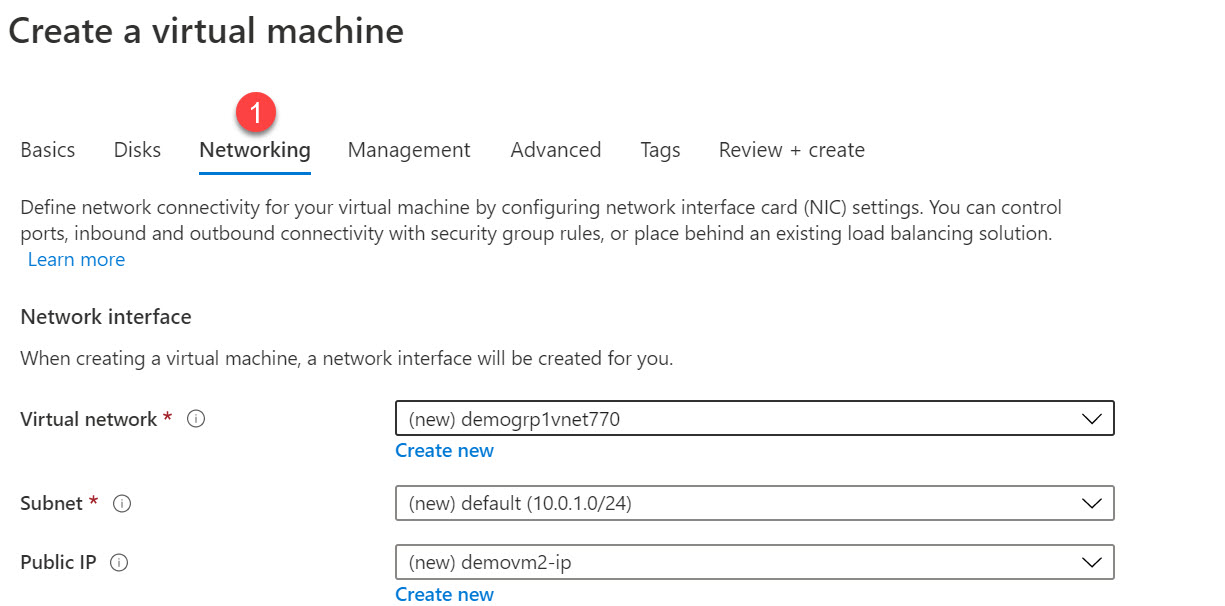


Virtual Machines - Accelerated networking

**Azure Virtual Machines - Accelerated networking**

Accelerated networking enables the use of single root I/O virtualization (SR-IOV) to a VM for a virtual machine. This provides better network performance for the virtual machine. This helps reduce latency, jitter, and CPU utilization for workloads that run on the Azure virtual machines

When you create a virtual machine in Azure using the wizard, when you go to the networking section



You can actually see the setting on the page



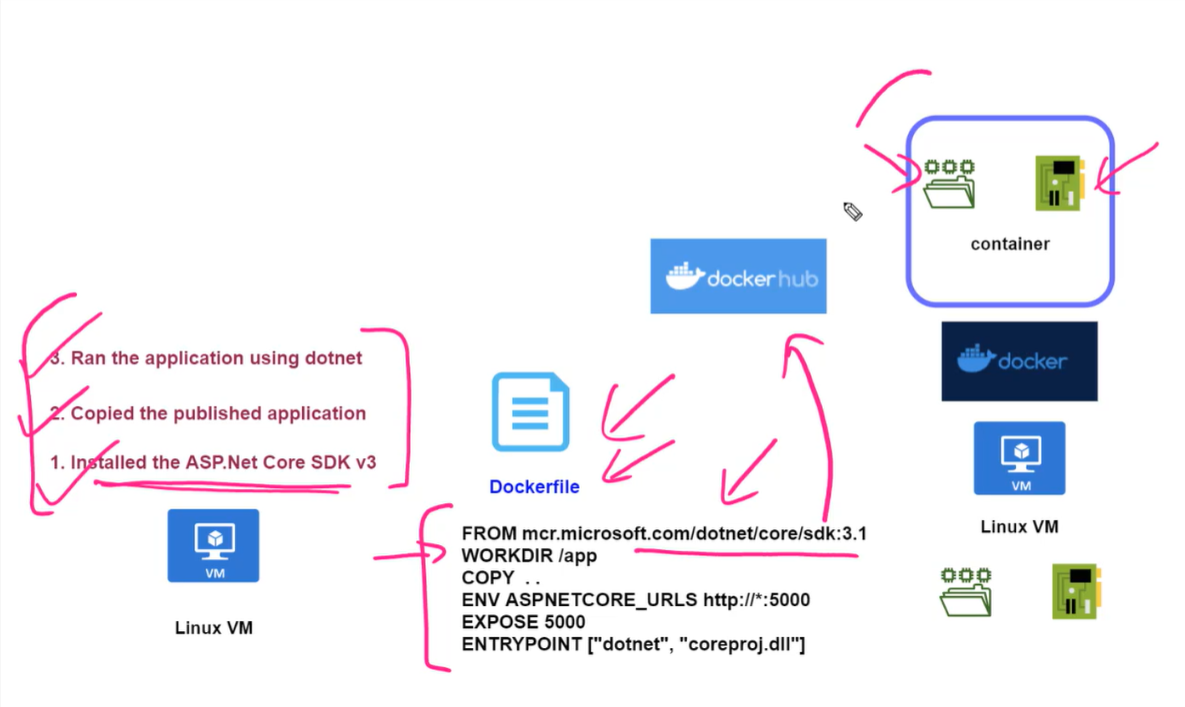
For more information on Accelerated networking , one can go to the following URL

<https://docs.microsoft.com/en-us/azure/virtual-network/create-vm-accelerated-networking-powershell>

Backup of VM:

* Create Recovery service vault
* Create Back Up Policy
  + Backup Schedule
  + Retention Range
  + Retention of back up points
* Perform Recovery based on restore points
  + File Recovery (run script)
  + VM Recovery
    - Create new VM (create storage account as staging area)
    - Replace Existing One (Only disk is replaced)

Containers and Docker:



WebApp Service

<https://docs.microsoft.com/en-us/azure/app-service/>

App deployment on WebAPP using Azure CLI:

$gitrepo="https://github.com/alashro/demoapp1000"

az group create --location centralus --name staging-grp

az appservice plan create --name newappplan2000 --resource-group staging-grp --sku B1

az webapp create --name newapp2000 --resource-group staging-grp --plan newappplan2000

az webapp deployment source config --name newapp2000 --resource-group staging-grp --repo-url $gitrepo --branch master --manual-integration

CORS:

az webapp cors add --allowed-origins

[--ids]

[--name]

[--resource-group]

[--slot]

[--subscription]

az webapp cors add --allowed-origins https://myapps.com --name MyWebApp --resource-group MyResourceGroup --subscription MySubscription

az storage cors add --methods {DELETE, GET, HEAD, MERGE, OPTIONS, POST, PUT}

--origins

--services

[--account-key]

[--account-name]

[--allowed-headers]

[--connection-string]

[--exposed-headers]

[--max-age]

[--sas-token]

[--subscription]

[--timeout]

az storage cors clear --services b --connection-string $connstr

az storage cors add --connection-string $connstr --origins '\*' --methods GET --allowed-headers 'x-ms-meta-abc,content-type,x-ms-blob-type,x-ms-meta-data\*,x-ms-meta-target\*' --exposed-headers 'x-ms-meta-\*' --max-age 200 --services blob

Azure Web Apps - Deployment Slots - PowerShell

**PowerShell commands for Azure Web Apps**

Please refer to the below important PowerShell commands for Azure Web Apps

**// First you can go ahead and create variables that can be used in the PowerShell script. Here we are setting the location for the web application, the name of a new resource group and the name of the new Azure Web App**

$location="Central US"

$resourcegrp="newgrp"

$webappname="demoapp4040"

**// Next we issue the command to create a new resource group**

New-AzResourceGroup -Name $resourcegrp -Location $location

**// Next we issue the command to create a new App Service Plan**

New-AzAppServicePlan -Name $webappname -Location $location -ResourceGroupName $resourcegrp -Tier Standard

**// Next we issue the command to create a new Web App**

New-AzWebApp -Name $webappname -Location $location -ResourceGroupName $resourcegrp -AppServicePlan $webappname

**// Next we issue the command to create a new Web App deployment slot**

New-AzWebAppSlot -Name $webappname -ResourceGroupName $resourcegrp -Slot "staging"

Functions:

<https://docs.microsoft.com/en-us/azure/azure-functions/>

Sample:

#r "Newtonsoft.Json"

using System.Net;

using Microsoft.AspNetCore.Mvc;

using Microsoft.Extensions.Primitives;

using Newtonsoft.Json;

public static async Task<IActionResult> Run(HttpRequest req, ILogger log)

{

log.LogInformation("C# HTTP trigger function processed a request.");

string requestBody = await new StreamReader(req.Body).ReadToEndAsync();

Course obj;

obj = JsonConvert.DeserializeObject<Course>(requestBody);

log.LogInformation((obj.id).ToString());

log.LogInformation(obj.name);

log.LogInformation((obj.rating).ToString());

return (ActionResult)new OkObjectResult(obj);

}

public class Course

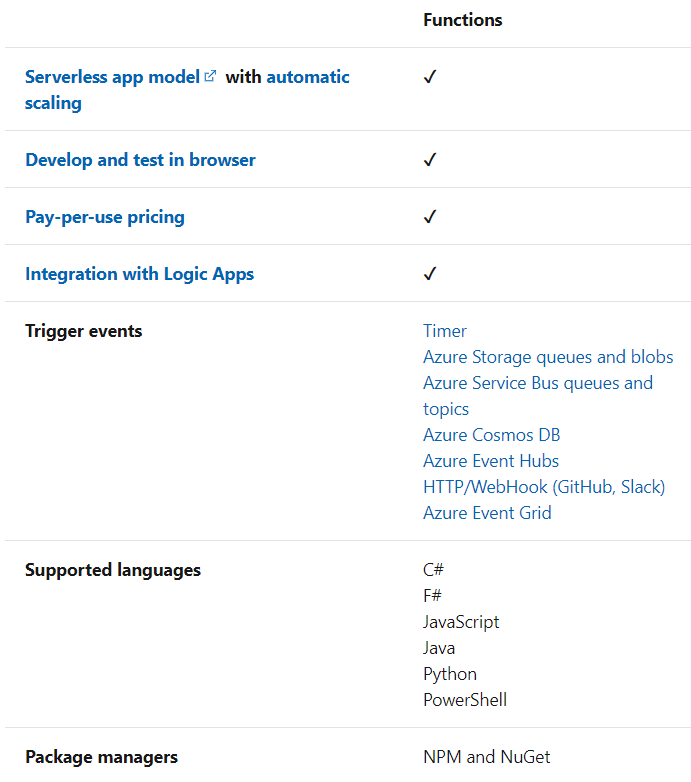
{

public int id {get;set;}

public string name {get;set;}

public double rating;

}



## Scenarios

In many cases, a function [integrates with an array of cloud services](https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings) to provide feature-rich implementations.

The following are a common, but by no means exhaustive, set of scenarios for Azure Functions.

| **SCENARIOS** | |
| --- | --- |
| **If you want to...** | **then...** |
| **Build a web API** | Implement an endpoint for your web applications using the [HTTP trigger](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-http-webhook) |
| **Process file uploads** | Run code when a file is uploaded or changed in [blob storage](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-blob) |
| **Build a serverless workflow** | Chain a series of functions together using [durable functions](https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-overview) |
| **Respond to database changes** | Run custom logic when a document is created or updated in [Cosmos DB](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-cosmosdb-v2) |
| **Run scheduled tasks** | Execute code at [set times](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-timer) |
| **Create reliable message queue systems** | Process message queues using [Queue Storage](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-queue), [Service Bus](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-service-bus), or [Event Hubs](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-event-hubs) |
| **Analyze IoT data streams** | Collect and process [data from IoT devices](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-event-iot) |
| **Process data in real time** | Use [Functions and SignalR](https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-signalr-service) to respond to data in the moment |

Durable Functions:

<https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-overview?tabs=csharp>

▪ beratungsgruppe wirth + partner (JOB)

▪ COTEC GmbH

▪ FISBA AG

▪ GFH GmbH

▪ Instrument Systems GmbH

▪ Laser 2000

▪ LASER COMPONENTS GmbH

▪ OPTROVISION-Bayernphotonics

▪ SCANLAB GmbH

▪ SPINNER GmbH

▪ **SUSS MicroOptics**