

- 1) Drag and Drop Question. You create a web application. You publish the source code of the web application to a GitHub repository by using Microsoft Visual Studio. You create a website by using the Azure management portal. You must continuously deploy the web application from the GitHub repository website to the Azure website. You need to deploy the source code of the web application. Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Actions	Answer Area
Select the repository and the branch from which to deploy the Azure website.	
Select GitHub as the source control method.	
Configure the Azure website to use the Always On option.	
In the Azure management portal, configure web endpoint monitoring.	
In the Azure management portal, choose the option to set up deployment from source control.	
Sign in to GitHub by using your deployment credentials.	

Answer:

Actions	Answer Area
Select the repository and the branch from which to deploy the Azure website.	In the Azure management portal, choose the option to set up deployment from source control.
Select GitHub as the source control method.	Select GitHub as the source control method.
Configure the Azure website to use the Always On option.	Sign in to GitHub by using your deployment credentials.
In the Azure management portal, configure web endpoint monitoring.	
In the Azure management portal, choose the option to set up deployment from source control.	Select the repository and the branch from which to deploy the Azure website.
Sign in to GitHub by using your deployment credentials.	

- 2) A company creates an API and makes it accessible on an Azure website. External partners use the API occasionally. The website uses the Standard web hosting plan. Partners report that the first API call in a sequence of API calls occasionally takes longer than expected to run. Subsequent API calls consistently perform as expected. You need to ensure that all API calls perform consistently. What should you do?
- a) Configure the website to use the Basic web hosting plan.
 - b) Enable Always On support.
 - c) Configure the website to automatically scale.
 - d) Add a trigger to the web.config file for the website that causes the website to recycle periodically.

Answer: B

- 3) Drag and Drop Question. Your team uses a proprietary source control product. You use FTP to manually deploy an Azure website. You must move your source code from the proprietary source control product to a secure onpremises Git versioning system. Instead of deploying the website by using FTP, the website must automatically deploy to Azure each time developers check-in source files. You need to implement the new deployment strategy. Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Actions	Answer Area
In the Azure management portal, configure websites to support deployment from the local Git repository.	
In the Azure management portal, configure websites to support deployment from external repository sources.	
In the Azure management portal, configure websites to support deployment from Microsoft Visual Studio Online.	
Commit the website to Azure.	
Create the website and add it to the local Git repository.	

Answer:

Actions	Answer Area
In the Azure management portal, configure websites to support deployment from the local Git repository.	Create the website and add it to the local Git repository.
In the Azure management portal, configure websites to support deployment from external repository sources.	In the Azure management portal, configure websites to support deployment from the local Git repository.
In the Azure management portal, configure websites to support deployment from Microsoft Visual Studio Online.	Commit the website to Azure.
Commit the website to Azure.	
Create the website and add it to the local Git repository.	

- 4) You deploy a website to Azure. When the website starts, it loads and caches common data. Updates to the website must occur without downtime or performance degradation that is noticeable to users. You need to upgrade to a new version of website code. What should you do?

A.

Create a staging slot for the new version of the website. Run the following Windows Powershell command, and then deploy the new code.

```
Switch-AzureWebsiteSlot -Name "MyWebsiteName"
```

B.

Create a staging slot for the new version of the website. Deploy the new code to the slot. Then run the following Windows Powershell command

```
Switch-AzureWebsiteSlot -Name "MyWebsiteName"
```

C.

Run the following Windows Powershell command.
`New-AzureWebsite -Name "staging" -Location "East US"`

Deploy the new code to the staging site. Then run the following Windows Powershell command
`Switch-AzureWebsiteSlot -Name "MyWebsiteName"`

D.

Create a new staging slot for the new version of the website. Run the following Windows Powershell command.

`Switch-AzureWebsiteSlot -Name "MyWebsiteName"`
Then deploy the new code to the staging slot.

Answer: B

- 5) You have a website that is hosted on Azure. You connect to the site by using the URI <http://www.contoso.com>. You plan to publish a new version of the website. You need to acquire the publishing profile for the website.

Which two actions will achieve the goal? (Select two.)

- A. Run the following Windows PowerShell cmdlet: `Get-AzurePublishSettingsFile`
- B. Run the following Windows PowerShell cmdlet: `Get-AzureSubscription`
- C. Navigate to the following URI:
<https://www.contoso.com/download/publishprofile.aspx>
- D. Navigate to the following URI:
<https://windows.azure.com/download/publishprofile.aspx>

Answer: **A, D**

- 6) You deploy a stateless ASP.NET application to an Azure website. You scale out the application by adding website instances. Only newly signed in users are routed to the recently added website instances. Users must be evenly distributed among all of the instances. You need to configure the environment to ensure that the load balancer evenly distributes requests. What should you do?

A.

Add the following markup to the web.config file for the application:

```
<system.webServer>
  <httpProtocol>
    <customHeaders>
      <add name="Arr-Disable-Session-Affinity" value="False" />
    </customHeaders>
  </httpProtocol>
</system.webServer>
```

B. Configure auto scaling rules based on metrics

C.

Add the following markup to the web.config file for the application:

```
<system.webServer>  
  <httpProtocol>  
    <customHeaders>  
      <add name="Arr-Disable-Session-Affinity" value="True" />  
    </customHeaders>  
  </httpProtocol>  
</system.webServer>
```

D. Enable AlwaysOn Support

Answer: C