

- **Vendor: Microsoft**
- **Exam Code: AZ-400**
- **Exam Name: Implementing and Operating Cisco Data Center Core Technologies (DCCOR)**
- **New Updated Questions from [Braindump2go](#) (Updated in [August/2021](#))**

**[Visit Braindump2go and Download Full Version AZ-400 Exam Dumps](#)**

**QUESTION 297**

**Case Study 2 - Contoso, Ltd  
Background**

Contoso, Ltd. is a manufacturing company that has a main office in Chicago. Contoso plans to improve its IT development and operations processes by implementing Azure DevOps principles. Contoso has an Azure subscription and creates an Azure DevOps organization.

The Azure DevOps organization includes:

- The Docker extension
- A deployment pool named Pool7 that contains 10 Azure virtual machines that run Windows Server 2016

The Azure subscription contains an Azure Automation account.

Contoso plans to create projects in Azure DevOps as shown in the following table.

Project name	Project details
Project 1	Project1 will provide support for incremental builds and third-party SDK components
Project 2	Project2 will use an automatic build policy. A small team of developers named Team2 will work independently on changes to the project. The Team2 members will not have permissions to Project2.
Project 3	Project3 will be integrated with SonarQube
Project 4	Project4 will provide support for a build pipeline that creates a Docker image and pushes the image to the Azure Container Registry. Project4 will use an existing Dockerfile.
Project 5	Project5 will contain a Git repository in Azure Reports and a continuous integration trigger that will initiate a build in response to any change except for changes within /folder1 of the repository.
Project 6	Project6 will provide support for build and deployment pipelines. Deployment will be allowed only if the number of current work items representing active software bugs is 0.
Project 7	Project7 will contain a target deployment group named Group7 that maps to Pool7. Project7 will use Azure Automation State Configuration to maintain the desired state of the computers in Group7.

**Technical requirements**

Contoso identifies the following technical requirements:

- Implement build agents for Project1.
- Whenever possible, use Azure resources.
- Avoid using deprecated technologies.
- Implement a code flow strategy for Project2 that will:
  - Enable Team2 to submit pull requests for Project2.
  - Enable Team2 to work independently on changes to a copy of Project2.

**[AZ-400 Exam Dumps](#) [AZ-400 Exam Questions](#) [AZ-400 PDF Dumps](#) [AZ-400 VCE Dumps](#)**

**<https://www.braindump2go.com/az-400.html>**

- Ensure that any intermediary changes performed by Team2 on a copy of Project2 will be subject to the same restrictions as the ones defined in the build policy of Project2.

- Whenever possible implement automation and minimize administrative effort.
- Implement Project3, Project5, Project6, and Project7 based on the planned changes
- Implement Project4 and configure the project to push Docker images to Azure Container Registry. You add the virtual machines as managed nodes in Azure Automation State Configuration. You need to configure the managed computers in Pool7. What should you do next?

- A. Modify the RefreshMode property of the Local Configuration Manager (LCM).
- B. Run the Register-AzureRmAutomationDscNode Azure Powershell cmdlet.
- C. Modify the ConfigurationMode property of the Local Configuration Manager (LCM).
- D. Install PowerShell Core.

**Answer: B**

**Explanation:**

The Register-AzureRmAutomationDscNode cmdlet registers an Azure virtual machine as an APS Desired State Configuration (DSC) node in an Azure Automation account.

Scenario: The Azure DevOps organization includes:

The Docker extension

A deployment pool named Pool7 that contains 10 Azure virtual machines that run Windows Server 2019

Project 7	Project7 will contain a target deployment group named Group7 that maps to Pool7. Project7 will use Azure Automation State Configuration to maintain the desired state of the computers in Group7.
-----------	---

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azurermsautomation/register-azurermsautomationdscnode>

**QUESTION 298**

**Case Study 2 - Contoso, Ltd**

**Background**

Contoso, Ltd. is a manufacturing company that has a main office in Chicago.

Contoso plans to improve its IT development and operations processes by implementing Azure DevOps principles.

Contoso has an Azure subscription and creates an Azure DevOps organization.

The Azure DevOps organization includes:

- The Docker extension
- A deployment pool named Pool7 that contains 10 Azure virtual machines that run Windows Server 2016

The Azure subscription contains an Azure Automation account.

Contoso plans to create projects in Azure DevOps as shown in the following table.

Project name	Project details
Project 1	Project1 will provide support for incremental builds and third-party SDK components
Project 2	Project2 will use an automatic build policy. A small team of developers named Team2 will work independently on changes to the project. The Team2 members will not have permissions to Project2.
Project 3	Project3 will be integrated with SonarQube
Project 4	Project4 will provide support for a build pipeline that creates a Docker image and pushes the image to the Azure Container Registry. Project4 will use an existing Dockerfile.
Project 5	Project5 will contain a Git repository in Azure Reports and a continuous integration trigger that will initiate a build in response to any change except for changes within /folder1 of the repository.
Project 6	Project6 will provide support for build and deployment pipelines. Deployment will be allowed only if the number of current work items representing active software bugs is 0.
Project 7	Project7 will contain a target deployment group named Group7 that maps to Pool7. Project7 will use Azure Automation State Configuration to maintain the desired state of the computers in Group7.

**Technical requirements**

Contoso identifies the following technical requirements:

- Implement build agents for Project1.
- Whenever possible, use Azure resources.
- Avoid using deprecated technologies.
- Implement a code flow strategy for Project2 that will:
  - Enable Team2 to submit pull requests for Project2.
  - Enable Team2 to work independently on changes to a copy of Project2.
  - Ensure that any intermediary changes performed by Team2 on a copy of Project2 will be subject to the same restrictions as the ones defined in the build policy of Project2.
- Whenever possible implement automation and minimize administrative effort.
- Implement Project3, Project5, Project6, and Project7 based on the planned changes
- Implement Project4 and configure the project to push Docker images to Azure Container Registry.

Drag and Drop Question

You need to implement the code flow strategy for Project2 in Azure DevOps.

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**

Create a fork

Create a branch

Add a build policy for the fork

Add a build policy for the master branch

Create a repository

Add an application access policy.

**Answer Area**



Answer:

**Actions**

Create a branch

Add a build policy for the master branch

Add an application access policy.

**Answer Area**

Create a repository

Create a fork

Add a build policy for the fork



**Explanation:**

Step 1: Create a repository

A Git repository, or repo, is a folder that you've told Git to help you track file changes in. You can have any number of repos on your computer, each stored in their own folder.

Step 2: Create a fork

Step 3: Add a build policy for the fork

Build policies help teams protect their important branches of development. Policies enforce your team's code quality and change management standards.

Scenario:

Implement a code flow strategy for Project2 that will:

Enable Team2 to submit pull requests for Project2.

Enable Team2 to work independently on changes to a copy of Project2. Ensure that any intermediary changes

performed by Team2 on a copy of Project2 will be subject to the same restrictions as the ones defined in the build policy of Project2.

Project2 will use an automatic build policy. A small team of developers named Team2 will work independently on changes to the project. The Team2 members will not have permissions to Project2.

Reference:

<https://docs.microsoft.com/en-us/azure/devops/repos/git/manage-your-branches>

### QUESTION 299

#### Case Study 2 - Contoso, Ltd

##### Background

Contoso, Ltd. is a manufacturing company that has a main office in Chicago.

Contoso plans to improve its IT development and operations processes by implementing Azure DevOps principles.

Contoso has an Azure subscription and creates an Azure DevOps organization.

The Azure DevOps organization includes:

- The Docker extension
- A deployment pool named Pool7 that contains 10 Azure virtual machines that run Windows Server 2016

The Azure subscription contains an Azure Automation account.

Contoso plans to create projects in Azure DevOps as shown in the following table.

Project name	Project details
Project 1	Project1 will provide support for incremental builds and third-party SDK components
Project 2	Project2 will use an automatic build policy. A small team of developers named Team2 will work independently on changes to the project. The Team2 members will not have permissions to Project2.
Project 3	Project3 will be integrated with SonarQube
Project 4	Project4 will provide support for a build pipeline that creates a Docker image and pushes the image to the Azure Container Registry. Project4 will use an existing Dockerfile.
Project 5	Project5 will contain a Git repository in Azure Reports and a continuous integration trigger that will initiate a build in response to any change except for changes within /folder1 of the repository.
Project 6	Project6 will provide support for build and deployment pipelines. Deployment will be allowed only if the number of current work items representing active software bugs is 0.
Project 7	Project7 will contain a target deployment group named Group7 that maps to Pool7. Project7 will use Azure Automation State Configuration to maintain the desired state of the computers in Group7.

##### Technical requirements

Contoso identifies the following technical requirements:

- Implement build agents for Project1.
- Whenever possible, use Azure resources.
- Avoid using deprecated technologies.
- Implement a code flow strategy for Project2 that will:
  - Enable Team2 to submit pull requests for Project2.
  - Enable Team2 to work independently on changes to a copy of Project2.
  - Ensure that any intermediary changes performed by Team2 on a copy of Project2 will be subject to the same restrictions as the ones defined in the build policy of Project2.
- Whenever possible implement automation and minimize administrative effort.
- Implement Project3, Project5, Project6, and Project7 based on the planned changes
- Implement Project4 and configure the project to push Docker images to Azure Container Registry.

Drag and Drop Question

You need to configure Azure Automation for the computers in Pool7.

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
Run the <code>Import-AzureRmAutomationDscConfiguration</code> Azure PowerShell cmdlet	
Create a Desired State Configuration (DSC) configuration file that has an extension of <code>.ps1</code> .	
Run the <code>New-AzureRmResourceGroupDeployment</code> Azure PowerShell cmdlet.	⏪ ⏩
Run the <code>Start-AzureRmAutomationDscCompilationJob</code> Azure PowerShell cmdlet.	⏪ ⏩
Create an Azure Resource Manager template file that has an extension of <code>.json</code> .	

**Answer:**

Actions	Answer Area
	Create a Desired State Configuration (DSC) configuration file that has an extension of <code>.ps1</code> .
	Run the <code>Import-AzureRmAutomationDscConfiguration</code> Azure PowerShell cmdlet.
Run the <code>New-AzureRmResourceGroupDeployment</code> Azure PowerShell cmdlet.	Run the <code>Start-AzureRmAutomationDscCompilationJob</code> Azure PowerShell cmdlet.
Create an Azure Resource Manager template file that has an extension of <code>.json</code> .	

**Explanation:**

Step 1: Create a Desired State Configuration (DSC) configuration file that has an extension of `.ps1`.

Step 2: Run the `Import-AzureRmAutomationDscConfiguration` Azure Powershell cmdlet The `Import-AzureRmAutomationDscConfiguration` cmdlet imports an APS Desired State Configuration (DSC) configuration into Azure Automation. Specify the path of an APS script that contains a single DSC configuration.

Example:

```
PS C:\>Import-AzureRmAutomationDscConfiguration -AutomationAccountName "Contoso17"-ResourceGroupName "ResourceGroup01" -SourcePath "C:\DSC\client.ps1" -Force
```

This command imports the DSC configuration in the file named `client.ps1` into the Automation account named `Contoso17`. The command specifies the `Force` parameter. If there is an existing DSC configuration, this command replaces it.

Step 3: Run the `Start-AzureRmAutomationDscCompilationJob` Azure Powershell cmdlet The `Start-AzureRmAutomationDscCompilationJob` cmdlet compiles an APS Desired State Configuration (DSC) configuration in Azure Automation.

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azurem.automation/import-azurermautomationdscconfiguration>  
<https://docs.microsoft.com/en-us/powershell/module/azurem.automation/start-azurermautomationdsccompilationjob>

**QUESTION 300**

**Case Study 3 - Woodgrove Bank**

**Overview**

**General Overview**

Woodgrove Bank is a financial services company that has a main office in the United Kingdom.

**Technical Requirements and Planned Changes**

**Planned Changes**

Woodgrove Bank plans to implement the following project management changes:

[AZ-400 Exam Dumps](#) [AZ-400 Exam Questions](#) [AZ-400 PDF Dumps](#) [AZ-400 VCE Dumps](#)

<https://www.braindump2go.com/az-400.html>

- Implement Azure DevOps for project tracking.
  - Centralize source code control in private GitHub repositories.
  - Implement Azure Pipelines for build pipelines and release pipelines.
- Woodgrove Bank plans to implement the following changes to the identity environment:

- Deploy an Azure AD tenant named woodgrovebank.com.
- Sync the Active Directory domain to Azure AD.
- Configure App1 to use a service principal.
- Integrate GitHub with Azure AD.

Woodgrove Bank plans to implement the following changes to the core apps:

- Migrate App1 to ASP.NET Core.
- Integrate Azure Pipelines and the third-party build tool used to develop App2.

Woodgrove Bank plans to implement the following changes to the DevOps environment:

- Deploy App1 to Azure App Service.
- Implement source control for the DB1 schema.
- Migrate all the source code from TFS1 to GitHub.
- Deploy App2 to an Azure virtual machine named VM1.
- Merge the POC branch into the GitHub default branch.
- Implement an Azure DevOps dashboard for stakeholders to monitor development progress.

#### Technical Requirements

Woodgrove Bank identifies the following technical requirements:

- The initial databases for new environments must contain both schema and reference data.
- An Azure Monitor alert for VM1 must be configured to meet the following requirements:
  - Be triggered when average CPU usage exceeds 80 percent for 15 minutes.
  - Calculate CPU usage averages once every minute.
- The commit history of the POC branch must replace the history of the default branch.
- The Azure DevOps dashboard must display the metrics shown in the following table.

Number	Required data
1	A comparison between the work the development team planned to deliver and what was delivered
2	The status of the environments in a release definition
3	The total number of results from a work item query

- Access to Azure DevOps must be restricted to specific IP addresses.
- Page load times for App1 must be captured and monitored.
- Administrative effort must be minimized.

You need to meet the technical requirements for controlling access to Azure DevOps.

What should you use?

- Azure Multi-Factor Authentication (MFA)
- on-premises firewall rules
- conditional access policies in Azure AD
- Azure role-based access control (Azure RBAC)

**Answer: C**

**Explanation:**

Scenario: Access to Azure DevOps must be restricted to specific IP addresses.

Azure DevOps is authenticated through Azure Active Directory. You can use Azure AD's conditional access to prevent logins from certain geographies and address ranges.

Reference:

<https://www.rebeladmin.com/2018/08/step-step-guide-configure-location-based-conditional-access-policies/>

**QUESTION 301**

**Case Study 3 - Woodgrove Bank**

**Overview**

**General Overview**

Woodgrove Bank is a financial services company that has a main office in the United Kingdom.

**Technical Requirements and Planned Changes**

**Planned Changes**

Woodgrove Bank plans to implement the following project management changes:

- Implement Azure DevOps for project tracking.
- Centralize source code control in private GitHub repositories.
- Implement Azure Pipelines for build pipelines and release pipelines.

Woodgrove Bank plans to implement the following changes to the identity environment:

- Deploy an Azure AD tenant named woodgrovebank.com.
- Sync the Active Directory domain to Azure AD.
- Configure App1 to use a service principal.
- Integrate GitHub with Azure AD.

Woodgrove Bank plans to implement the following changes to the core apps:

- Migrate App1 to ASP.NET Core.
- Integrate Azure Pipelines and the third-party build tool used to develop App2.

Woodgrove Bank plans to implement the following changes to the DevOps environment:

- Deploy App1 to Azure App Service.
- Implement source control for the DB1 schema.
- Migrate all the source code from TFS1 to GitHub.
- Deploy App2 to an Azure virtual machine named VM1.
- Merge the POC branch into the GitHub default branch.
- Implement an Azure DevOps dashboard for stakeholders to monitor development progress.

**Technical Requirements**

Woodgrove Bank identifies the following technical requirements:

- The initial databases for new environments must contain both schema and reference data.
- An Azure Monitor alert for VM1 must be configured to meet the following requirements:
  - Be triggered when average CPU usage exceeds 80 percent for 15 minutes.
  - Calculate CPU usage averages once every minute.
- The commit history of the POC branch must replace the history of the default branch.
- The Azure DevOps dashboard must display the metrics shown in the following table.

<b>Number</b>	<b>Required data</b>
1	A comparison between the work the development team planned to deliver and what was delivered
2	The status of the environments in a release definition
3	The total number of results from a work item query



- Access to Azure DevOps must be restricted to specific IP addresses.
- Page load times for App1 must be captured and monitored.
- Administrative effort must be minimized.

You need to configure Azure Pipelines to control App2 builds.  
Which authentication method should you use?

- A. Windows NTLM
- B. certificate
- C. SAML
- D. personal access token (PAT)

**Answer:** D

**Explanation:**

Scenario: Deploy App2 to an Azure virtual machine named VM1.

A personal access token (PAT) is used as an alternate password to authenticate into Azure DevOps.

Reference:

<https://docs.microsoft.com/en-us/azure/devops/organizations/accounts/use-personal-access-tokens-to-authenticate>

### **QUESTION 302**

#### **Case Study 3 - Woodgrove Bank**

##### **Overview**

##### **General Overview**

Woodgrove Bank is a financial services company that has a main office in the United Kingdom.

##### **Technical Requirements and Planned Changes**

##### **Planned Changes**

Woodgrove Bank plans to implement the following project management changes:

- Implement Azure DevOps for project tracking.
- Centralize source code control in private GitHub repositories.
- Implement Azure Pipelines for build pipelines and release pipelines.

Woodgrove Bank plans to implement the following changes to the identity environment:

- Deploy an Azure AD tenant named woodgrovebank.com.
- Sync the Active Directory domain to Azure AD.
- Configure App1 to use a service principal.
- Integrate GitHub with Azure AD.

Woodgrove Bank plans to implement the following changes to the core apps:

- Migrate App1 to ASP.NET Core.
- Integrate Azure Pipelines and the third-party build tool used to develop App2.

Woodgrove Bank plans to implement the following changes to the DevOps environment:

- Deploy App1 to Azure App Service.
- Implement source control for the DB1 schema.
- Migrate all the source code from TFS1 to GitHub.
- Deploy App2 to an Azure virtual machine named VM1.
- Merge the POC branch into the GitHub default branch.
- Implement an Azure DevOps dashboard for stakeholders to monitor development progress.

##### **Technical Requirements**

Woodgrove Bank identifies the following technical requirements:

- The initial databases for new environments must contain both schema and reference data.
- An Azure Monitor alert for VM1 must be configured to meet the following requirements:
  - Be triggered when average CPU usage exceeds 80 percent for 15 minutes.
  - Calculate CPU usage averages once every minute.
- The commit history of the POC branch must replace the history of the default branch.
- The Azure DevOps dashboard must display the metrics shown in the following table.

Number	Required data
1	A comparison between the work the development team planned to deliver and what was delivered
2	The status of the environments in a release definition
3	The total number of results from a work item query

- Access to Azure DevOps must be restricted to specific IP addresses.
- Page load times for App1 must be captured and monitored.
- Administrative effort must be minimized.

#### Hotspot Question

You need to configure the alert for VM1. The solution must meet the technical requirements.

Which two settings should you configure? To answer, select the appropriate settings in the answer area.

NOTE: Each correct selection is worth one point.

#### Alert logic

Threshold  Static  Dynamic

Operator  Greater than  Less than  Equal to  Not equal to

Aggregation type \*  Average  Minimum  Maximum  Sum

Threshold value \*

#### Condition preview

Whenever the average percentage cpu is greater than <logic undefined> %

#### Evaluated based on

Aggregation granularity (Period) \*  5 minutes  15 minutes  30 minutes  1 hour

Frequency of evaluation  Every 1 Minute  Every 5 Minutes  Every 15 Minutes  Every 30 Minutes  Every 1 Hour

Answer:

#### Alert logic

Threshold  Static  Dynamic

Operator  Greater than  Less than  Equal to  Not equal to

Aggregation type \*  Average  Minimum  Maximum  Sum

Threshold value \*

#### Condition preview

Whenever the average percentage cpu is greater than <logic undefined> %

#### Evaluated based on

Aggregation granularity (Period) \*  5 minutes  15 minutes  30 minutes  1 hour

Frequency of evaluation  Every 1 Minute  Every 5 Minutes  Every 15 Minutes  Every 30 Minutes  Every 1 Hour

#### Explanation:

Setting 1: Threshold value

Set to 80 %

Scenario: An Azure Monitor alert for VM1 must be configured to meet the following requirements:

Be triggered when average CPU usage exceeds 80 percent for 15 minutes.

Calculate CPU usage averages once every minute.

Setting 2: Aggregation granularity  
 Set to 15 minutes.

**QUESTION 303**

**Case Study 3 - Woodgrove Bank**

**Overview**

**General Overview**

Woodgrove Bank is a financial services company that has a main office in the United Kingdom.

**Technical Requirements and Planned Changes**

**Planned Changes**

Woodgrove Bank plans to implement the following project management changes:

- Implement Azure DevOps for project tracking.
- Centralize source code control in private GitHub repositories.
- Implement Azure Pipelines for build pipelines and release pipelines.

Woodgrove Bank plans to implement the following changes to the identity environment:

- Deploy an Azure AD tenant named woodgrovebank.com.
- Sync the Active Directory domain to Azure AD.
- Configure App1 to use a service principal.
- Integrate GitHub with Azure AD.

Woodgrove Bank plans to implement the following changes to the core apps:

- Migrate App1 to ASP.NET Core.
- Integrate Azure Pipelines and the third-party build tool used to develop App2.

Woodgrove Bank plans to implement the following changes to the DevOps environment:

- Deploy App1 to Azure App Service.
- Implement source control for the DB1 schema.
- Migrate all the source code from TFS1 to GitHub.
- Deploy App2 to an Azure virtual machine named VM1.
- Merge the POC branch into the GitHub default branch.
- Implement an Azure DevOps dashboard for stakeholders to monitor development progress.

**Technical Requirements**

Woodgrove Bank identifies the following technical requirements:

- The initial databases for new environments must contain both schema and reference data.
- An Azure Monitor alert for VM1 must be configured to meet the following requirements:
  - Be triggered when average CPU usage exceeds 80 percent for 15 minutes.
  - Calculate CPU usage averages once every minute.
- The commit history of the POC branch must replace the history of the default branch.
- The Azure DevOps dashboard must display the metrics shown in the following table.

<b>Number</b>	<b>Required data</b>
1	A comparison between the work the development team planned to deliver and what was delivered
2	The status of the environments in a release definition
3	The total number of results from a work item query

- Access to Azure DevOps must be restricted to specific IP addresses.
- Page load times for App1 must be captured and monitored.
- Administrative effort must be minimized.

**Drag and Drop Question**

You need to replace the existing DevOps tools to support the planned changes.

What should you use? To answer, drag the appropriate tools to the correct targets. Each tool may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Tools	Answer Area
Azure Boards	Trello: <span style="border: 1px dashed black; padding: 5px;">Tool</span>
Azure Artifacts	Bamboo: <span style="border: 1px dashed black; padding: 5px;">Tool</span>
GitHub Actions	BitBucket: <span style="border: 1px dashed black; padding: 5px;">Tool</span>
Azure Pipelines	
Azure Test Plans	
GitHub repositories	

**Answer:**

Tools	Answer Area
Azure Artifacts	Trello: <span style="border: 1px solid black; padding: 5px;">Azure Boards</span>
GitHub Actions	Bamboo: <span style="border: 1px solid black; padding: 5px;">Azure Pipelines</span>
Azure Test Plans	BitBucket: <span style="border: 1px solid black; padding: 5px;">GitHub repositories</span>

**Explanation:**

Box 1: Azure Boards

Azure Boards can be used to track work with Kanban boards, backlogs, team dashboards, and custom reporting. You can create multiple Trello boards, which are spaces to store tasks (for different work contexts, or even private boards).

You can easily share Trello boards with another person.

Box 2: Azure Pipelines

You can use Bamboo to implement CI/CD (Continuous Integration and Continuous Delivery) for a simple Azure function app using Atlassian Bamboo. Bamboo does continuous delivery of the project from source code to deployment. It has stages including Build, Test and Deploy.

Software teams in every industry are upgrading their continuous delivery pipeline with Bamboo. Easy build import from popular open source tools, user and group import from JIRA, seamless integration with Bitbucket, and native support for Git, Hg, and SVN means you'll be building and deploying like a champ.

Box 3: GitHub repositories

Bitbucket can be used as the Git repository, but you can use any other Git repository (Like TFS Git) for source control of the code.

Reference:

<https://www.trustradius.com/compare-products/azure-devops-services-vs-trello>

<https://marketplace.visualstudio.com/items?itemName=ms-vsts.vss-services-bamboo>

<https://www.c-sharpcorner.com/article/cicd-implementation-for-an-azure-function-app-using-atlassian-bamboo-server/>

#### **QUESTION 304**

You use GitHub for source control.

A file that contains sensitive data is committed accidentally to the Git repository of a project.

You need to delete the file and its history from the repository.

Which two tools can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. the git filter-branch command
- B. BFG Repo-Cleaner
- C. the git rebase command
- D. GitHub Desktop

**Answer:** AB

**Explanation:**

To entirely remove unwanted files from a repository's history you can use either the git filter-branch command or the BFG Repo-Cleaner open source tool.

Reference:

<https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/removing-sensitive-data-from-a-repository>

#### **QUESTION 305**

Your company uses GitHub for source control. The company has a team that performs code reviews.

You need to automate the assignment of the code reviews. The solution must meet the following requirements:

Prioritize the assignment of code reviews to team members who have the fewest outstanding assignments. Ensure that each team member performs an equal number of code reviews in any 30-day period.

Prevent the assignment of code reviews to the team leader.

Which two actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Clear Never assign certain team members.
- B. Select If assigning team members, don't notify the entire team.
- C. Select Never assign certain team members.
- D. Set Routing algorithm to Round robin.
- E. Set Routing algorithm to Load balance.

**Answer:** AE

**Explanation:**

A: To always skip certain members of the team, select Never assign certain team members. Then, select one or more team members you'd like to always skip. In this case select the team leader.

E: The load balance algorithm chooses reviewers based on each member's total number of recent review requests and considers the number of outstanding reviews for each member. The load balance algorithm tries to ensure that each team member reviews an equal number of pull requests in any 30 day period.

Incorrect Answers:

D: The round robin algorithm chooses reviewers based on who's received the least recent review request, focusing on alternating between all members of the team regardless of the number of outstanding reviews they currently have.

Reference:

<https://docs.github.com/en/organizations/organizing-members-into-teams/managing-code-review-assignment-for-your-team>

**QUESTION 306**

You have a project in Azure DevOps named Project1. Project1 contains a published wiki. You need to change the order of pages in the navigation pane of the published wiki in the Azure DevOps portal. What should you do?

- A. At the root of the wiki, create a file named .order that defines the page hierarchy.
- B. At the root of the wiki, create a file named wiki.md that defines the page hierarchy.
- C. Rename the pages in the navigation pane.
- D. Drag and drop the pages in the navigation pane.

**Answer: D**

**Explanation:**

Reorder a wiki page

You can reorder pages within the wiki tree view to have pages appear in the order and hierarchy you want.

You can drag-and-drop a page title in the tree view to do the following operations:

- Change the parent-child relationship of a page
- Change the order of the page within the hierarchy

Reference:

<https://docs.microsoft.com/en-us/azure/devops/project/wiki/add-edit-wiki>

**QUESTION 307**

You have a project in Azure DevOps.

You plan to deploy a self-hosted agent by using an unattended configuration script.

Which two values should you define in the configuration script? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. authorization credentials
- B. the project name
- C. the deployment group name
- D. the organization URL
- E. the agent pool name

**Answer: AD**

**Explanation:**

Unattended config:

The agent can be set up from a script with no human intervention. You must pass --unattended and the answers to all questions.

To configure an agent, it must know the URL to your organization or collection and credentials of someone authorized to set up agents. All other responses are optional.

Reference:

<https://docs.microsoft.com/en-us/azure/devops/pipelines/agents/v2-windows>

**QUESTION 308**

You have Azure Pipelines and GitHub integrated as a source code repository.

The build pipeline has continuous integration enabled.

You plan to trigger an automated build whenever code changes are committed to the repository.

You need to ensure that the system will wait until a build completes before queuing another build.

What should you implement?

- A. path filters
- B. batch changes
- C. scheduled builds
- D. branch filters

**Answer: B**

**Explanation:**

**[AZ-400 Exam Dumps](#) [AZ-400 Exam Questions](#) [AZ-400 PDF Dumps](#) [AZ-400 VCE Dumps](#)**

**<https://www.braindump2go.com/az-400.html>**

### Batching CI runs

If you have many team members uploading changes often, you may want to reduce the number of runs you start. If you set batch to true, when a pipeline is running, the system waits until the run is completed, then starts another run with all changes that have not yet been built.

Example:

```
# specific branch build with batching
```

```
trigger:
```

```
batch: true
```

```
branches:
```

```
include:
```

```
-master
```

To clarify this example, let us say that a push A to master caused the above pipeline to run. While that pipeline is running, additional pushes B and C occur into the repository. These updates do not start new independent runs immediately. But after the first run is completed, all pushes until that point of time are batched together and a new run is started.

Reference:

<https://docs.microsoft.com/en-us/azure/devops/pipelines/repos/github>

### QUESTION 309

You are using GitHub as a source code repository.

You create a client-side Git hook on the commit-msg event. The hook requires that each commit message contain a custom work item tag.

You need to make a commit that does not have a work item tag.

Which git commit parameter should you use?

- A. --squash
- B. --no-verify
- C. --message "
- D. --no-post-rewrite

**Answer:** B

**Explanation:**

The commit-msg hook is invoked by git-commit and git-merge, and can be bypassed with the --no-verify option.

Reference:

<https://git-scm.com/docs/githooks>

### QUESTION 310

Drag and Drop Question

You have a project in Azure DevOps named Project1 that contains two Azure DevOps pipelines named Pipeline1 and Pipeline2.

You need to ensure that Pipeline1 can deploy code successfully to an Azure web app named webapp1. The solution must ensure that Pipeline2 does not have permission to webapp1.

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**

Create a service principal in Azure Active Directory.

In Project1, create a service connection.

In Pipeline1, authorize the service connection.

Create a system-assigned managed identity in Azure Active Directory.

In Project1, configure permissions.

In Pipeline1, create a variable.



Answer:

**Actions**

**Answer Area**

Create a service principal in Azure Active Directory.

In Project1, create a service connection.

In Pipeline1, authorize the service connection.

Create a system-assigned managed identity in Azure Active Directory.

In Project1, configure permissions.

In Pipeline1, create a variable.



**Explanation:**

<https://docs.microsoft.com/en-us/azure/devops/pipelines/library/connect-to-azure?view=azure-devops>

**QUESTION 311**

Drag and Drop Question

You need to increase the security of your team's development process.

Which type of security tool should you recommend for each stage of the development process? To answer, drag the appropriate security tools to the correct stages. Each security tool may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.



**Security Tools**

**Answer Area**

Penetration testing

Pull request:

Static code analysis

Continuous integration:

Threat modeling

Continuous delivery:

Answer:

**Security Tools**

**Answer Area**

Pull request:

Threat modeling

Continuous integration:

Static code analysis

Continuous delivery:

Penetration testing

**Explanation:**

Box 1: Threat modeling -

Threat modeling's motto should be, "The earlier the better, but not too late and never ignore."

Box 2: Static code analysis -

Validation in the CI/CD begins before the developer commits his or her code. Static code analysis tools in the IDE provide the first line of defense to help ensure that security vulnerabilities are not introduced into the CI/CD process.

Box 3: Penetration testing -

Once your code quality is verified, and the application is deployed to a lower environment like development or QA, the process should verify that there are not any security vulnerabilities in the running application. This can be accomplished by executing automated penetration test against the running application to scan it for vulnerabilities.

Reference:

<https://docs.microsoft.com/en-us/azure/devops/articles/security-validation-cicd-pipeline?view=vsts>

**QUESTION 312**

Hotspot Question

You have an Azure virtual machine named VM1 that runs Linux.

You plan to deploy the Desired State Configuration (DSC) extension to VM1.

You need to grant the Log Analytics agent the appropriate directory permissions.

How should you complete the command? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Answer Area**

setfacl -m u:omsagent:

<input type="text"/>	<input type="text"/>
r	/lib
x	/etc
rx	/tmp
rwX	/usr

Answer:

**Answer Area**

```
setfacl -m u:omsagent:
```

r	/lib
x	/etc
rx	/tmp
rwX	/usr

**Explanation:**

Box 1: rwX

The Log Analytics agent for Linux runs as the omsagent user. To grant >write permission to the omsagent user, run the command setfacl -m u:omsagent:rwX /tmp.

Box 2: /tmp

Deploying DSC to a Linux node uses the /tmp folder.

Reference:

<https://docs.microsoft.com/en-us/azure/automation/automation-dsc-onboarding>

**QUESTION 313**

Hotspot Question

You are using PowerShell to administer Azure Log Analytics workspaces.

You need to list the available workspaces and their properties.

How should you complete the command? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Answer Area**

		Microsoft.OperationalInsights/workspaces - ExpandProperties
Get-AzResource	-ResourceGroupName	
Get-AzResourceGroup	-ResourceId	
Get-AzResourceProvider	-ResourceType	

Answer:

**Answer Area**

		Microsoft.OperationalInsights/workspaces - ExpandProperties
Get-AzResource	-ResourceGroupName	
Get-AzResourceGroup	-ResourceId	
Get-AzResourceProvider	-ResourceType	

**Explanation:**

Box 1: Get-AzResource

Use the following command to examine the access control mode for all workspaces in the subscription:

PowerShell

```
Get-AzResource -ResourceType Microsoft.OperationalInsights/workspaces -ExpandProperties | foreach {$_ .Name + ": " + $_.Properties.features.enableLogAccessUsingOnlyResourcePermissions
```

Box 2: -ResourceType

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/logs/manage-access>