

|  |
| --- |
| Azure DevOps Lab  **Microsoft Azure** |
| Module 3. Infrastructure as Code: ARM  Home task |

TASK

*Please read all the points below before beginning implementation*

1. Create an ARM template for an initial deployment that will be used for the linked templates execution.
2. Create two linked templates that will create a Storage Account (SA) and a VNET containing two subnets (front-end/back-end).
3. Use Azure Resource Manager template functions to concise your code and follow task requirements (at least one)
4. SA names must be unique - use function for its generation
5. Use object type in the parameter file for setting VNETs’ options
6. Run a deployment of the initial ARM template and deploy required resources.

Result:three valid templates with the required resources. PowerShell/Bash (using AzureCLI) deployment script.

TASK HINTS

1. SA and VNET options (Name, Type, Subnet names etc.) should be passed as parameters AND as variables. You must define their appropriate usage yourself.

REQUIREMENTS

1. One JSON file for describing all Azure resources is forbidden. **Please use linked templates.**

2. Linked templates **must be called from the initial deployment template** (main.json).

3. Main and parameters templates must be named as **main.json** and **parameters.json** accordingly.

4. Maximum **number of parameters** in Main.json is 7.

5. Main.json and parameters.json must be **executed from local folder**. Using **-TemplateParameterUri** and **-TemplateUri** options in PS script is forbidden.

6. All artifacts (JSONs and PS files) must be stored in **Azure Storage Account**. Using any GitHubs or other public repos is forbidden.  
7. Create a PowerShell/Bash (using Azure CLI) deployment script, which will be used for running all your deployments. The script should have the following functionality:

                a. Create resource group.

b. Create storage account and container within for artifacts (For example: JSONs, PS file(s), ZIP files).

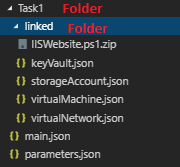
c. Upload the linked templates and other task-related artifacts to the created Storage Account.

d. Execute main.json file for deploy Azure resources.

8. Each ARM json file must have at least **1 output**.

9. All homework **artifacts must be executable** (e.g. if Mentor starts your script execution and it fails - all homework artifacts will be sent back for fixing)

10. All **resources must be deleted** after homework completion.

11. Use the next **folder structure** for storing artifacts. **Subfolder** with resources JSONs must be named **“linked”:**  


Useful links

[Azure Resource Manager template functions](https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-template-functions)

# [Best Practices For Using Azure Resource Manager Templates](https://blogs.msdn.microsoft.com/mvpawardprogram/2018/05/01/azure-resource-manager/)