**What are ARM Templates?**

Azure Portal

Azure CLI

Powershell

ARM Template (Json)

The **template** is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project.

The configuration in an ARM template is written in JSON format and saved with a .json extension.

JSON is a data format that is used to exchange data between a web browser and a server,

**We can Creat resources through**

Azure Portal

Azure CLI

Powershell

ARM Template (Json)

**Why choose ARM templates**

Declarative syntax: ARM templates allow you to create and deploy an entire Azure infrastructure declaratively. For example, you can deploy not only virtual machines, but also the network infrastructure, storage systems, and any other resources you may need.

Repeatable results: Repeatedly deploy your infrastructure throughout the development lifecycle and have confidence your resources are deployed in a consistent manner. Templates are idempotent, which means you can deploy the same template many times and get the same resource types in the same state.

Orchestration: You don't have to worry about the complexities of ordering operations. Resource Manager orchestrates the deployment of interdependent resources so they're created in the correct order.

**An ARM template follow some structure: with 5 components**

{

"$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",

"contentVersion": "",

"parameters": { },

"variables": { },

"resources": [ ],

"outputs": { }

}

T**emplate Consists of 6 Elements**

**$schema** is the location of the JSON schema file that describes the version of the template language.

where you want to deploy the template Management group Level,Subscription level,Resource group level

**Parameters** are values that are provided when deployment is executed in order to customize resource deployment,Which is Static Data

e.g. “Username/pswd”.

**Resources** are used to define the resource types that are deployed or updated in a resource group e.g. a storage account i.e. Microsoft.Storage/storageAccounts.

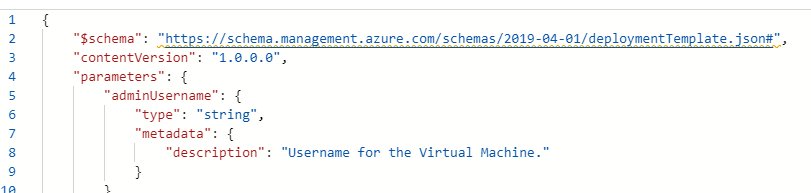
contentVersion It is used to describe the version of the template.

**Variables** Variables You use to simplify your template. Rather than repeating complicated expressions throughout your template ,it Makes your template easier to main ,Which can Change e.g. "storageAccountName": "StorageAccount 1“.

**Outputs** are values that are returned after deployment.

**$schema** It defines the structure of Template &

version of the template.



**ContentVersion** :It is used to describe the version of the template