



# Activate Azure with DevOps

Microsoft Services





# Activate Azure with DevOps

## Module 2: Introduction to Azure Resource Manager

Microsoft Services



# Objectives

After completing this learning unit, you will understand

- Azure Resource Manager
- Resource Providers
- Azure Resource Manager Tools
- Troubleshooting Azure Resource Manager

# Azure Resource Manager

# Benefits of ARM

- Desired-state deployment
- Faster deployments
- Role-based access control (RBAC)
- Resource-provider model
- Common interface for Azure and Azure Stack



# ARM Definitions

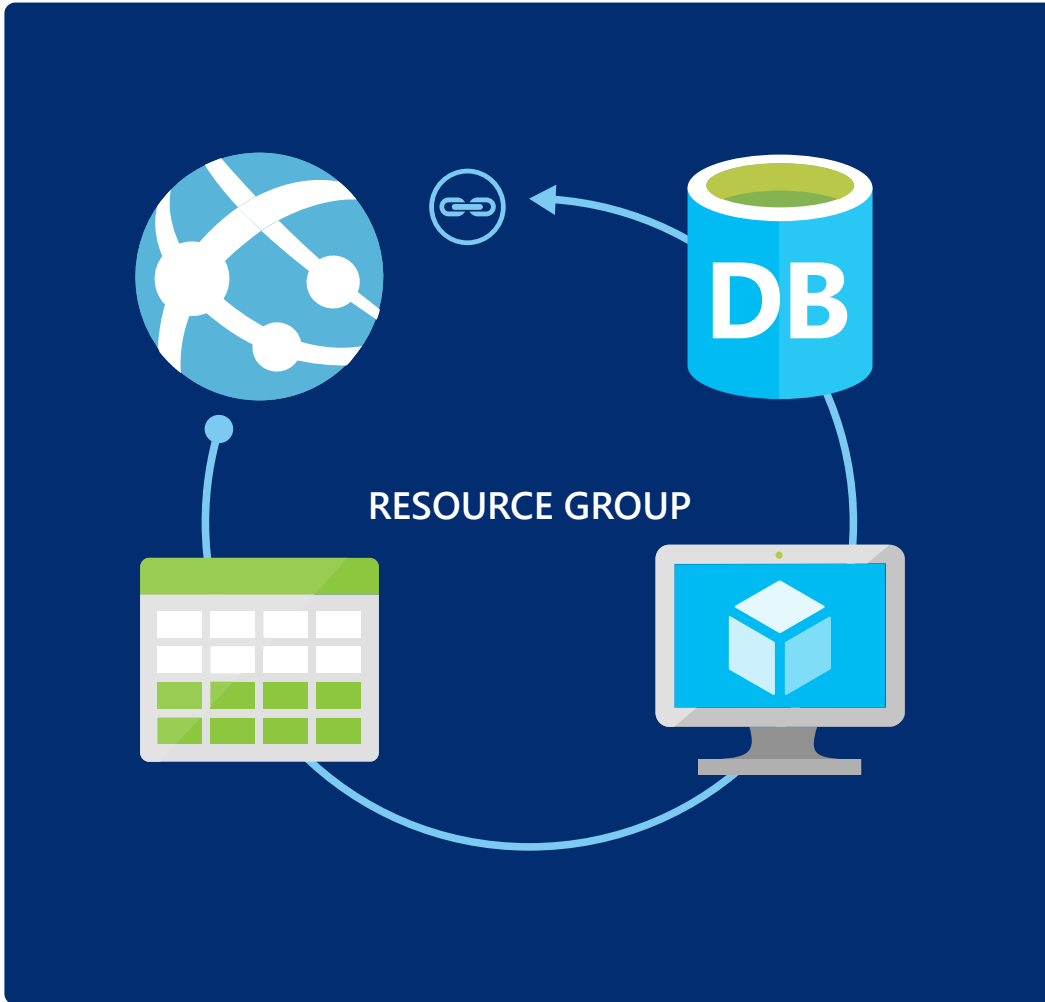
- Resource - Atomic unit of deployment
- Resource group - Collection of resources
- Resource provider - Manages specific kinds of resources
- Resource type - Specifies the type of resource

# Resource Groups

*A Resource Group is a Unit of Management*

- Application Lifecycle Containment – deploy, manage, and monitor all the resources for your solution as a group
- Declarative templates – “Config as Code”
- Grouping – Billing and cost can be applied and rolled up to the group
- Consistent Management Layer
- Access Control – Scope for RBAC permissions

# Azure Resource Groups



- Tightly coupled containers of multiple resources of similar or different types
- Resource groups can span regions
- Every resource *\*must\** exist in one and only one resource group



# Resource Group Lifecycle

Question:

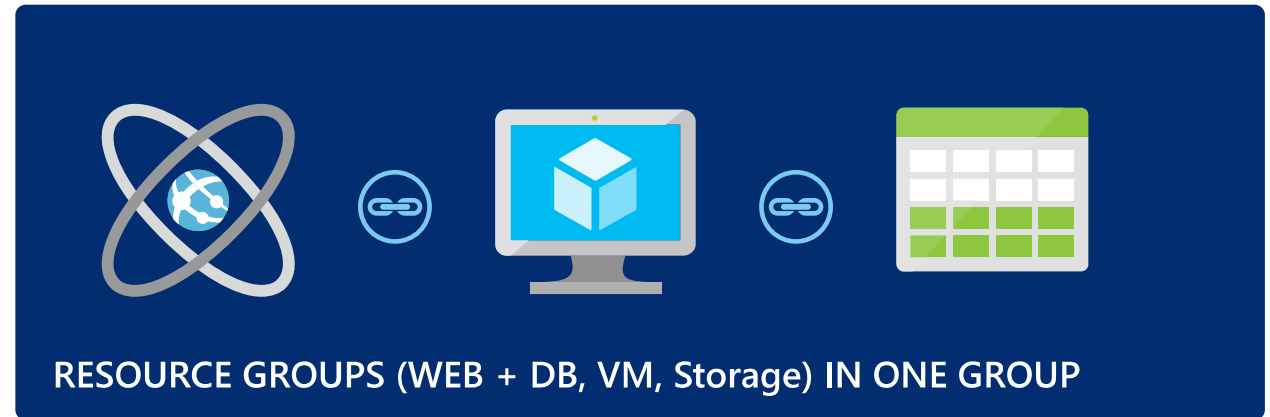
Should these resources be in the same group or a different one?

Hint:

Do they have common lifecycle and management?

Answer:

It's up to you.



OR



# Organizational Concepts

## Resource groups

---

- Tightly coupled to a container of resources
- Follows RBAC rules
- 1 resource group at a time

## Tags

---

- Loosely coupled user or system defined categorization
- Arbitrary boundaries
- 15 tags to use as needed
- *(no RBAC support)*

Resource groups and tags are the building blocks from which users define **applications**.

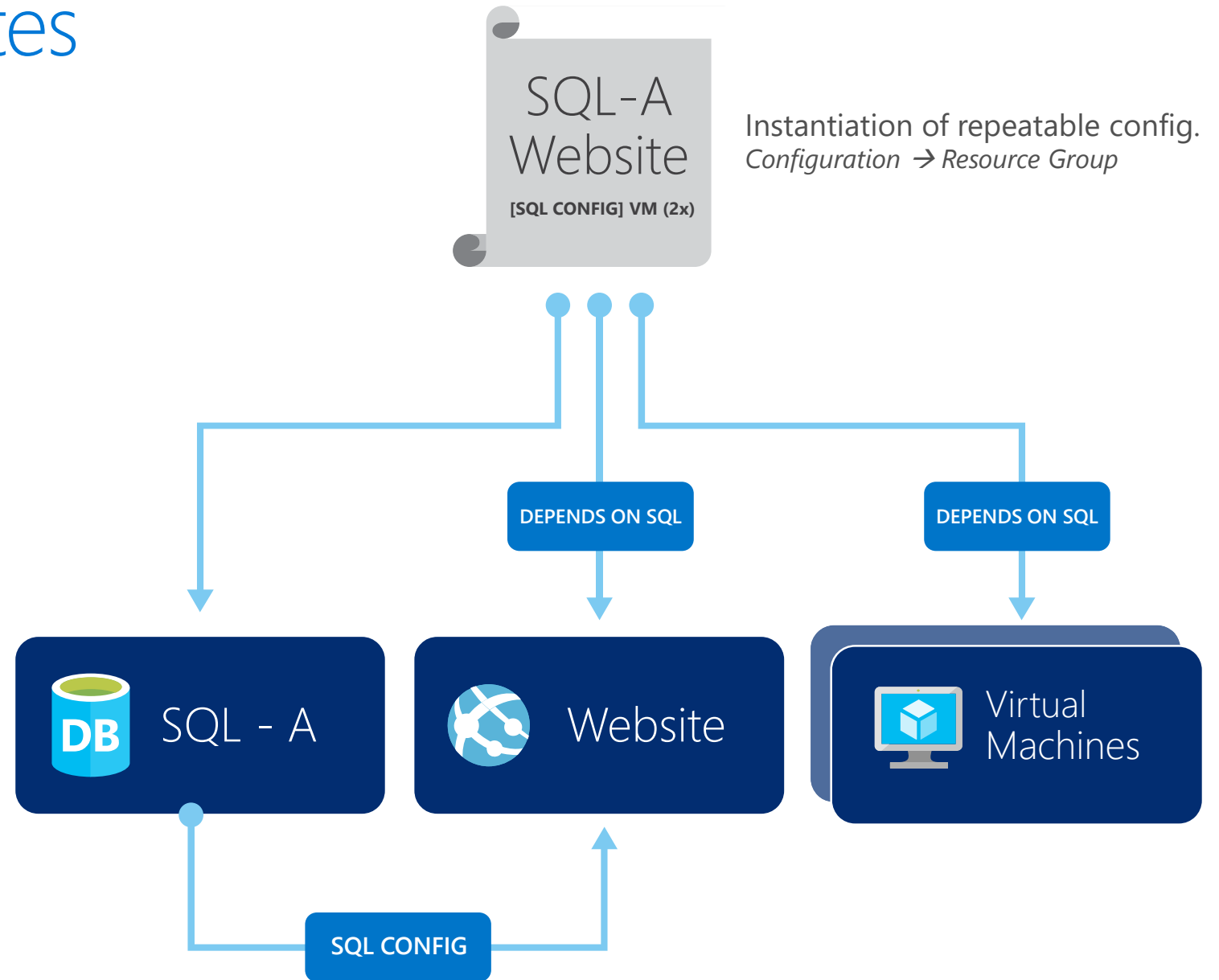
# Deployment Templates

## What?

- Source file, can be checked-in
- Specifies resources, dependencies and connections
- Parameterized input/output

## Why?

- Ensure Idempotency
- Simplify orchestration
- Provide cross-resource configuration and update support



# Demonstration: Resource Groups and Templates



# Resource Providers

# Resource Providers

- Deploy specific types of resources
- Identified by provider namespaces
  - e.g., Microsoft.Compute, Microsoft.Storage, Microsoft.Web
- Resource types
  - Each provider namespace manages one or more resource types: e.g. virtualMachines
  - Different regional availability and apiVersions
- PowerShell
  - Get-AzureLocation
  - Get-AzureRmResourceProvider
  - Get-AzResourceProvider (newer Az Module)

# Resource Provider Types (examples)

Resource Type	Usage
Microsoft.Compute/virtualMachines	Virtual Machines
Microsoft.Storage/storageAccounts	Storage Accounts
Microsoft.Compute/availabilitySets	Availability Sets
Microsoft.Network/networkInterfaces	Virtual Network Interface Card (NIC)
Microsoft.Network/loadBalancers	Azure Load Balancer or Internal Load Balancer
Microsoft.Network/virtualNetworks	Virtual networks
Microsoft.ClassicStorage	ASM Storage
Microsoft.ClassicCompute	ASM Virtual Machines



# Resource Definition Example

- Name
  - Unique for resource group and resource type (e.g., Microsoft.Compute/virtualMachines)
- Id
  - Unique across Azure
- Location
- ResourceType
- ResourceGroup
- Properties
  - Additional properties specific to the resource provider

# Azure Resource Manager Tools

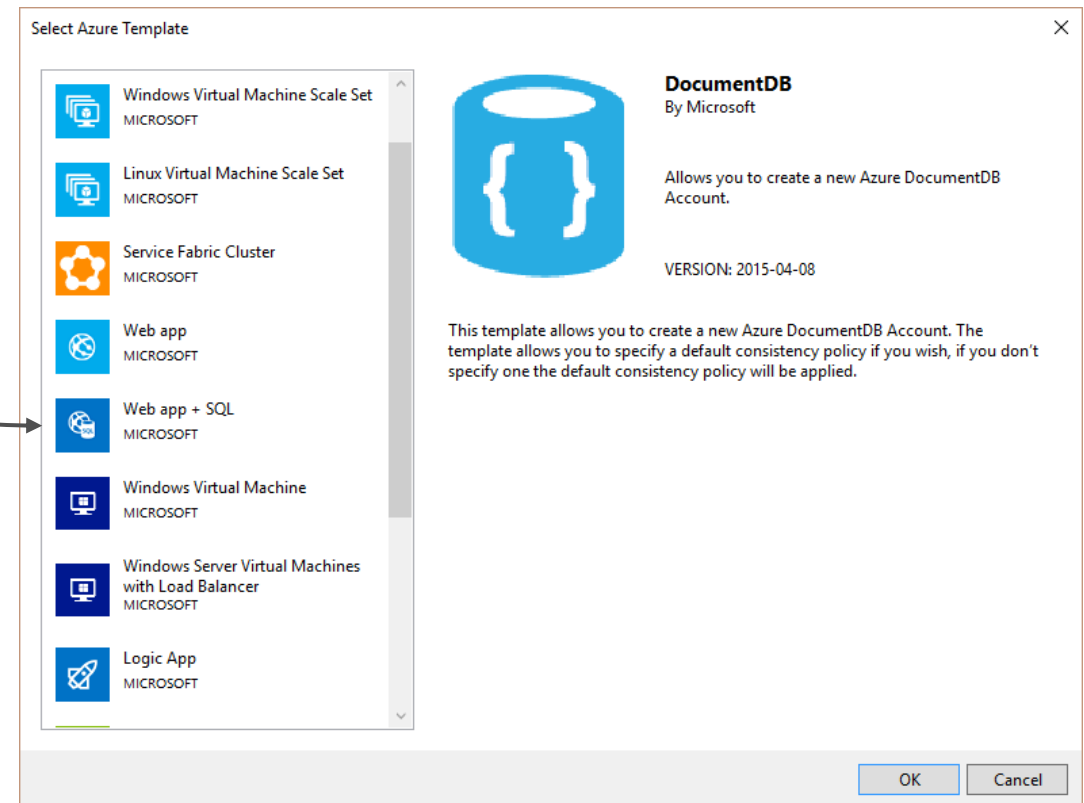
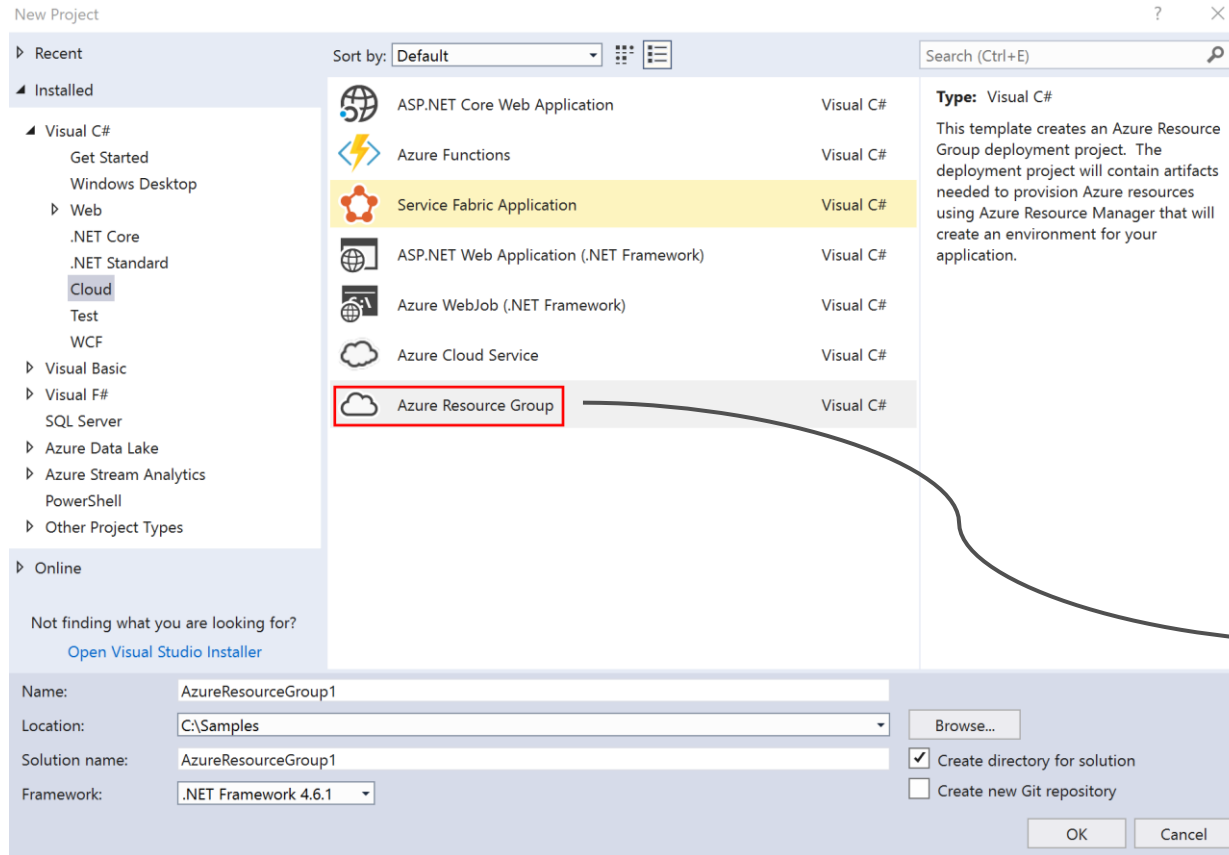
# Tools typically used by IT with ARM

- PowerShell
- Azure CLI
- Azure Cloud Console
- Troubleshooting in the portal
- Visual Studio

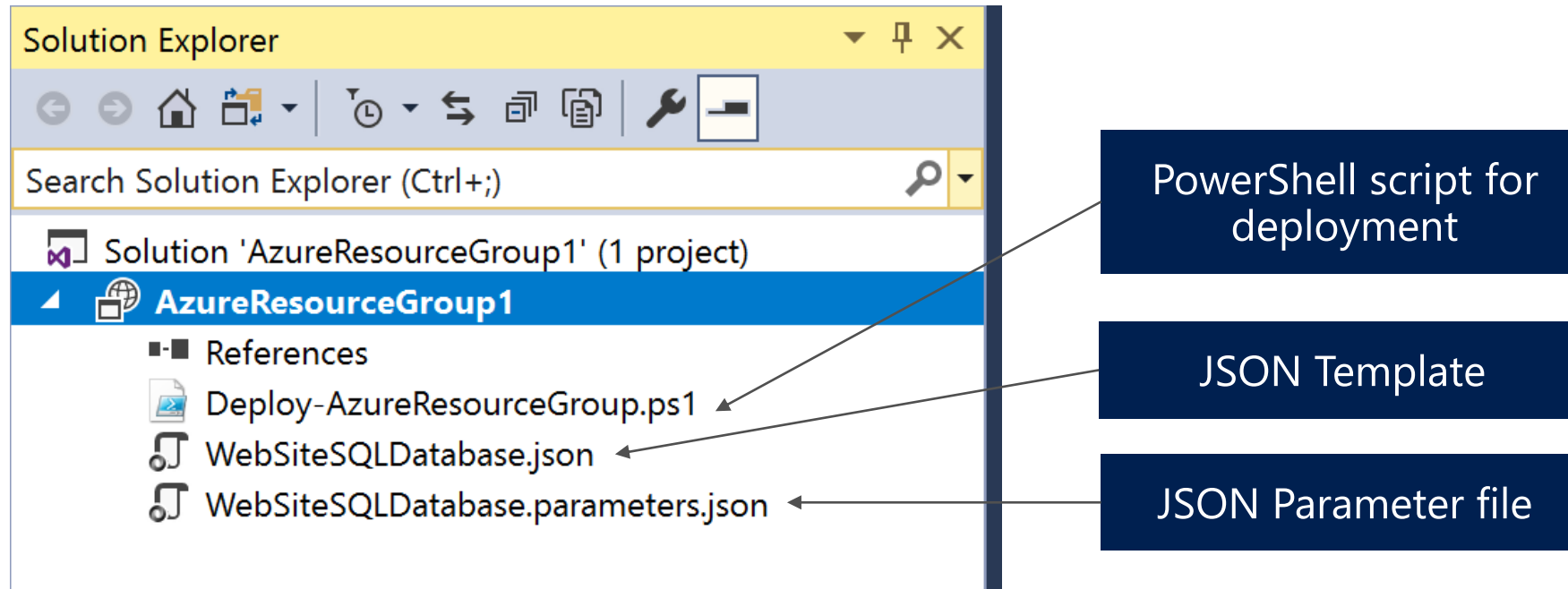
# Visual Studio

- Provides a new template from the gallery
- Allows the deployment of a resource group
- Generates scripts to deploy application

# Template Options



# Files Generated by Template



# JSON Template

- Declarative programming
- Structure elements
  - Parameters
  - Variables
  - Resources
  - Outputs

```
"resources": [  
  {  
    "name": "[variables('sqlserverName')]",  
    "type": "Microsoft.Sql/servers",  
    "location": "[resourceGroup().location]",  
    "tags": {  
      "displayName": "SqlServer"  
    },  
    "apiVersion": "2014-04-01-preview",  
    "properties": {  
      "administratorLogin": "[parameters('administratorLogin')]",  
      "administratorLoginPassword": "[parameters('administratorLoginPassword')]"  
    },  
    "resources": [  
      {  
        "name": "[parameters('databaseName')]",  
        "type": "databases",  
        "location": "[resourceGroup().location]",  
        "tags": {  
          "displayName": "Database"  
        },  
        "apiVersion": "2014-04-01-preview",  
        "dependsOn": [  
          "[concat('Microsoft.Sql/servers/', variables('sqlserverName'))]"  
        ],  
        "properties": {  
          "edition": "[parameters('edition')]",  

```



# Exporting your Template

**PartsUnlimited - Export template**  
Resource group - PREVIEW

Search (Ctrl+/) « **Download** Add to library Deploy

5 resource types cannot be exported yet and are not included in the template. See error details. →

To export related resources, select the resources from the Resource Group view then select the "Export template" option from the tool bar.

Template Parameters CLI PowerShell .NET Ruby

Parameters (16)  
Variables (0)  
Resources (106)  
insightsComponents (micro...  
insightsComponents (micro...  
insightsComponents (micro...  
PartsUnlimitedServer (Micro...  
PartsUnlimitedServer (Micro...  
PartsUnlimitedServer (Micro...  
PartsUnlimitedHostingPlan (...  
WebsiteHostingPlanAlertRul...  
[parameters("alertrules\_Failur...  
[parameters("alertrules\_Failur...  
[parameters("alertrules\_Failur...  
WebsiteAlertRuleForbiddenR...  
WebsiteHostingPlanAlertRul...  
WebsiteAlertRuleServerError...

```
1 {  
2   "$schema":  
   "https://schema.management.azure.com/schemas/2015-01-01/deploymentTem  
   e.json#",  
3   "contentVersion": "1.0.0.0",  
4   "parameters": {  
5     "serverfarms_PartsUnlimited_name": {  
6       "defaultValue": "PartsUnlimited",  
7       "type": "String"  
8     },  
9     "sites_PartsUnlimitedPocNate_name": {  
10      "defaultValue": "PartsUnlimitedPocNate",  
11      "type": "String"  
12    },  
13    "servers_partsunlimitedpocdbserv_name": {  
14      "defaultValue": "partsunlimitedpocdbserv",  
15      "type": "String"  
16    },  
17    "servers_partsunlimitedpocdbservdev_name": {  
18      "defaultValue": "partsunlimitedpocdbservdev",  
19      "type": "String"  
20    },  
21  }
```

# Demonstration: Visual Studio Deployment



# Lab: ARM Templates

Build Azure Resource Manager templates

<https://docs.microsoft.com/en-us/learn/modules/build-azure-vm-templates/index>

