**Azure Storage**

**Azure Storage** is Microsoft’s cloud storage solution for modern data storage scenarios. It provides highly available, durable, scalable, and secure storage for a variety of data objects.

Types of Azure storage

1. **Blob Storage: Used for: Unstructured data like images, videos, backups, documents.**
2. **File Storage (Azure Files):**

* **Used for: Shared file systems accessed via SMB or NFS protocols.**
* **Features: Can be mounted to Azure VMs or on-premises machines using Azure File Sync.**

1. **Queue Storage: Used for: Message queues for communication between application components.**

### **Table Storage:**

* **Used for: NoSQL key-value data with fast access and scalable schema.**
* **Replaced in many cases by: Azure Cosmos DB (Table API) for advanced features.**

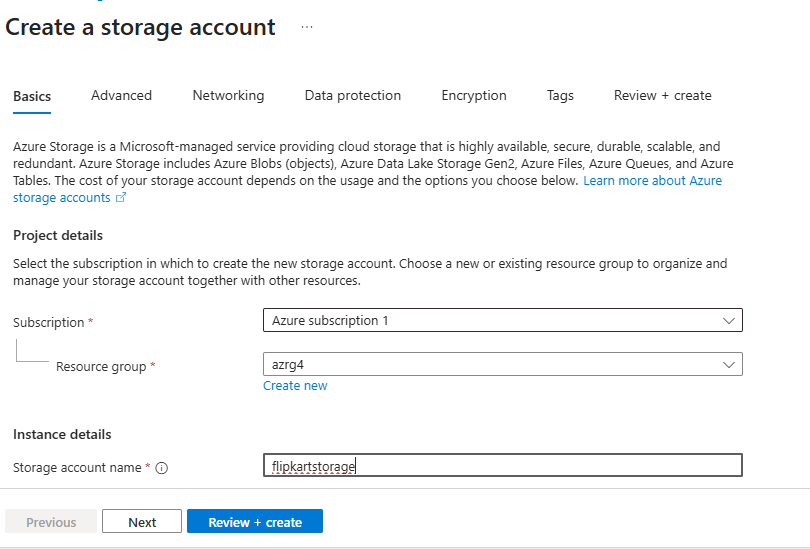
1. **Disk Storage:**

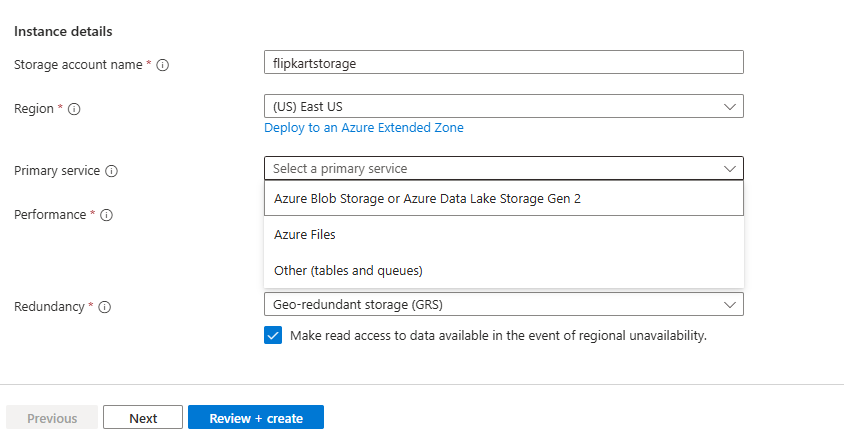
**Used for: Persistent storage for Azure Virtual Machines (VMs).**

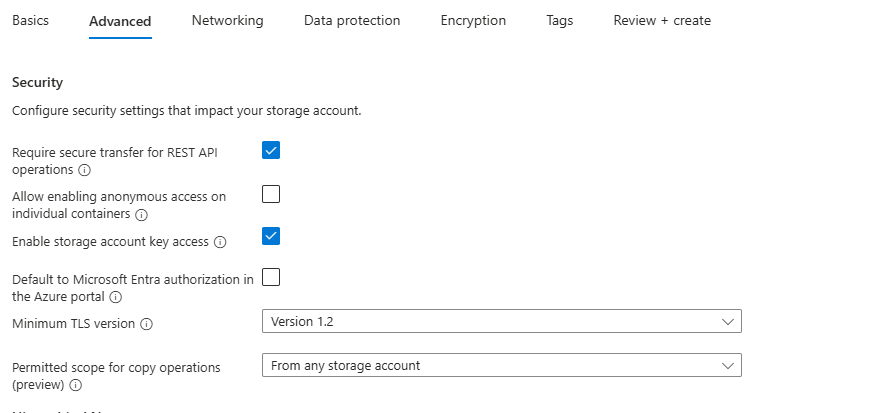
**Types:**

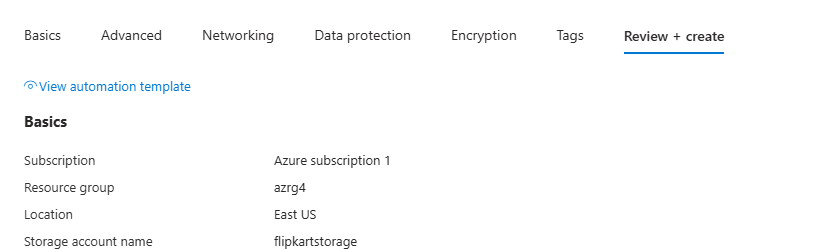
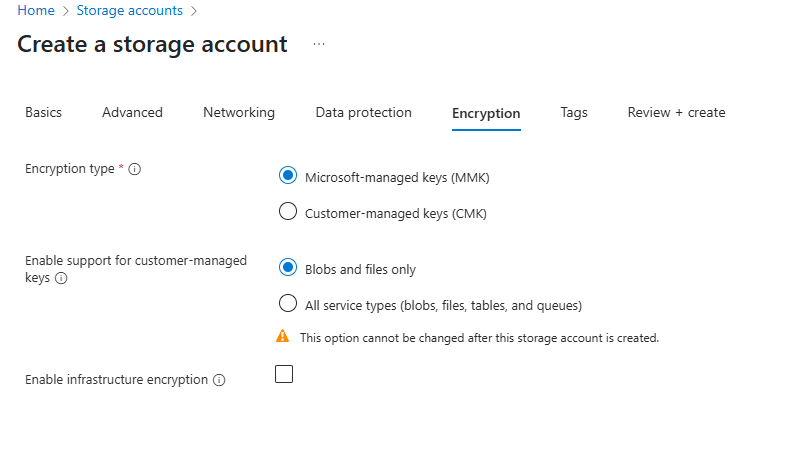
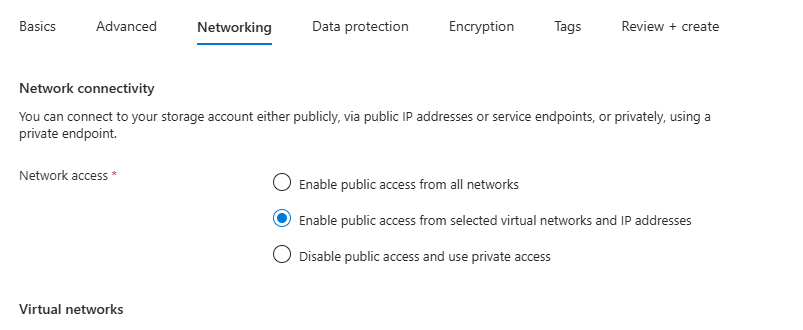
* **Standard HDD**
* **Standard SSD**
* **Premium SSD**
* **Ultra Disk**

**Implementation Screenshots**

****

****

****

****

**Git Hub**

**What is a Repository (Repo)?**

A **repository** is a storage space for your project. It contains all your project files and the version history tracked by **Git**.

## **Local Repository**

* A **local repo** exists on your machine.
* Created with git init or cloned from a remote repo using git clone.
* You make changes, commit them locally, and push them to the remote repo (like GitHub).

### **Structure:**

* .git/ (hidden folder): Contains all the Git metadata.
* Working directory: Your actual files and code.
* Staging area (index): Files marked for commit.

## **Remote Repository (e.g., GitHub)**

* A **remote repo** is hosted on a platform like **GitHub**, **GitLab**, or **Bitbucket**.
* It's used for:  
  + Collaboration
  + Backup
  + CI/CD integration
* You push (upload) and pull (download) code between the local and remote repositories.

COMMANDS

# 1. Create a new local repo

git init

# 2. Connect it to a remote GitHub repo

git remote add origin https://github.com/username/repo-name.git

# 3. Stage files for commit

git add .

# 4. Commit your changes

git commit -m "Initial commit"

# 5. Push to GitHub (remote repo)

git push -u origin main

## **Cloning an Existing GitHub Repo**

git clone https://github.com/username/repo-name.git

cd repo-name

## **Common Git Commands**

## 

| **Command** | **Purpose** |
| --- | --- |
| git status | Check changes in the working directory |
| git add <file> | Stage a file |
| git commit -m "message" | Commit staged files |
| git push | Send commits to remote repo |
| git pull | Fetch and merge from remote |
| git clone <url> | Create local copy of remote repo |

## **Summary Table**

## 

| **Feature** | **Local Repo** | **Remote Repo** |
| --- | --- | --- |
| Location | Your local machine | GitHub / GitLab / Bitbucket |
| Purpose | Development & testing | Collaboration, backup |
| Created via | git init or git clone | GitHub UI / gh repo create |
| Sync with | Remote repo | Local repo |

## 