Diese Aufgaben sind darauf ausgerichtet, den Schülern zu zeigen, wie man mit Netzwerken, Speicher und verschiedenen Azure-Diensten so arbeitet, dass eine effektive Kommunikation zwischen diesen Ressourcen ermöglicht wird.

### 1. \*\*Erstellen eines virtuellen Netzwerks und einer virtuellen Maschine\*\*

\*\*Aufgabe\*\*: Erstelle ein virtuelles Netzwerk (VNet) und eine virtuelle Maschine (VM) und teste die Kommunikation zwischen beiden.

- Erstelle ein VNet mit Subnetzen.

- Erstelle eine VM im VNet.

- Stelle sicher, dass die VM eine private IP-Adresse im Subnetz des VNets erhält.

- Teste die Kommunikation der VM mit einer anderen VM innerhalb desselben VNets (z. B. mit einem Ping-Befehl).

\*\*Zusatzaufgabe\*\*:

- Erstelle eine zweite VM im gleichen VNet und konfiguriere eine statische IP.

- Teste, ob beide VMs miteinander kommunizieren können (z. B. Ping von VM1 zu VM2).

---

### 2. \*\*Kommunikation zwischen VNets über VNet Peering\*\*

\*\*Aufgabe\*\*: Konfiguriere VNet Peering zwischen zwei verschiedenen VNets.

- Erstelle zwei VNets in verschiedenen Regionen oder im gleichen Region.

- Konfiguriere VNet Peering, damit Ressourcen in den beiden VNets miteinander kommunizieren können.

- Teste die Kommunikation zwischen VMs in den beiden VNets, indem du die IP-Adressen der VMs pingen lässt.

\*\*Zusatzaufgabe\*\*:

- Erstelle unterschiedliche Subnetze in jedem VNet und teste, ob die Kommunikation zwischen diesen Subnetzen funktioniert.

---

### 3. \*\*Kommunikation zwischen einem VNet und einem Azure Storage Account\*\*

\*\*Aufgabe\*\*: Erstelle ein VNet und einen Azure Storage Account und ermögliche den Zugriff auf den Storage Account von einer VM im VNet.

- Erstelle ein VNet und eine VM in diesem VNet.

- Erstelle einen Azure Storage Account.

- Aktiviere "Private Endpoint" für den Storage Account, damit der Zugriff nur über das VNet erfolgt.

- Greife von der VM aus auf den Storage Account zu, indem du die Azure CLI oder PowerShell verwendest, um Dateien hochzuladen oder herunterzuladen.

\*\*Zusatzaufgabe\*\*:

- Teste die Kommunikation, indem du versuchst, auf den Storage Account von einer VM außerhalb des VNets zuzugreifen. Es sollte nicht möglich sein.

---

### 4. \*\*Erstellen und Konfigurieren einer Azure Load Balancer für die Kommunikation zwischen VMs\*\*

\*\*Aufgabe\*\*: Erstelle eine Azure Load Balancer und konfiguriere sie für den Lastenausgleich zwischen zwei VMs.

- Erstelle zwei VMs in einem VNet.

- Erstelle einen Load Balancer und konfiguriere ihn so, dass er den eingehenden Verkehr zwischen den beiden VMs verteilt.

- Teste die Funktion, indem du den Load Balancer ansprichst und überprüfst, ob der Verkehr korrekt zwischen den VMs verteilt wird.

\*\*Zusatzaufgabe\*\*:

- Erstelle ein Frontend-IP für den Load Balancer und konfiguriere die Backend-Pool für die VMs.

- Teste die Skalierung, indem du eine weitere VM hinzufügst und überprüfst, ob der Load Balancer automatisch die Kommunikation dorthin weiterleitet.

---

### 5. \*\*Verbindung zu einem Azure SQL Database von einer VM\*\*

\*\*Aufgabe\*\*: Erstelle eine Azure SQL-Datenbank und stelle sicher, dass eine VM über SQL Server Management Studio (SSMS) oder eine andere Anwendung darauf zugreifen kann.

- Erstelle eine Azure SQL-Datenbank.

- Erstelle eine VM im selben VNet.

- Konfiguriere die Firewall-Einstellungen der SQL-Datenbank, um Verbindungen aus der VM zu ermöglichen.

- Verbinde dich von der VM mit der SQL-Datenbank und führe einfache SQL-Abfragen aus.

\*\*Zusatzaufgabe\*\*:

- Teste den Zugriff von einer VM, die sich in einem anderen VNet oder in einer anderen Region befindet, und konfiguriere gegebenenfalls VNet Peering oder VPN für den Zugriff.

---

### 6. \*\*Erstellen einer Verbindung zwischen einem VNet und einer On-Premises Infrastruktur (VPN Gateway)\*\*

\*\*Aufgabe\*\*: Konfiguriere eine VPN-Verbindung zwischen einem Azure VNet und einer On-Premises Infrastruktur.

- Erstelle ein VNet und ein VPN-Gateway in Azure.

- Konfiguriere die On-Premises Seite des VPN-Gateways und stelle sicher, dass die Kommunikation mit dem VNet funktioniert.

- Teste die Kommunikation von einem lokalen Server (z. B. On-Premises VM) zu einer Ressource im Azure VNet (z. B. eine VM oder eine Datenbank).

\*\*Zusatzaufgabe\*\*:

- Teste die Kommunikation, indem du Ressourcen in beiden Umgebungen anpingst oder eine andere Art der Netzwerkkommunikation ausprobierst (z. B. eine Datei übertragen).

---

### 7. \*\*Kommunikation zwischen Azure Kubernetes Service (AKS) und einer Azure SQL-Datenbank\*\*

\*\*Aufgabe\*\*: Erstelle ein Azure Kubernetes Service (AKS) Cluster und ermögliche den Zugriff von einem Pod auf eine Azure SQL-Datenbank.

- Erstelle ein AKS-Cluster.

- Erstelle eine Azure SQL-Datenbank.

- Konfiguriere die nötigen Firewall- und Netzwerkeinstellungen, damit ein Pod innerhalb des AKS-Clusters auf die SQL-Datenbank zugreifen kann.

- Schreibe eine einfache Anwendung im AKS-Cluster, die mit der SQL-Datenbank kommuniziert (z. B. eine Abfrage ausführen).

\*\*Zusatzaufgabe\*\*:

- Teste die Kommunikation, indem du die SQL-Datenbank von einem anderen Pod aus ansprichst, und stelle sicher, dass die Verbindung erfolgreich ist.

---

### 8. \*\*Kommunikation zwischen Azure Functions und Azure Storage\*\*

\*\*Aufgabe\*\*: Erstelle eine Azure Function und konfiguriere diese so, dass sie Daten in einem Azure Blob Storage ablegt.

- Erstelle eine Azure Function, die auf HTTP-Anfragen reagiert.

- Erstelle einen Azure Blob Storage Account.

- Konfiguriere die Azure Function so, dass sie nach dem Empfang einer HTTP-Anfrage Daten in einem Blob speichert.

- Teste die Kommunikation, indem du eine HTTP-Anfrage an die Azure Function sendest und überprüfst, ob die Daten im Blob Storage gespeichert wurden.

\*\*Zusatzaufgabe\*\*:

- Verwende einen Azure Queue Storage als Trigger für die Azure Function und speichere die Daten in einem Blob Storage.

---

### 9. \*\*Kommunikation zwischen Azure Virtual Machine und Azure Redis Cache\*\*

\*\*Aufgabe\*\*: Erstelle eine Azure Redis Cache-Instanz und stelle sicher, dass eine VM darauf zugreifen kann.

- Erstelle eine Redis Cache-Instanz in Azure.

- Erstelle eine VM im selben VNet wie der Redis Cache.

- Konfiguriere die VM und Redis Cache so, dass die VM auf den Redis Cache zugreifen kann (z. B. mit einem Redis-Client).

- Teste die Kommunikation, indem du einen einfachen Redis-Befehl von der VM sendest.

\*\*Zusatzaufgabe\*\*:

- Konfiguriere eine skalierte Anwendung, die den Redis Cache zur Speicherung von Daten verwendet.

---

### 10. \*\*Verbindung von Azure Functions mit Azure Service Bus\*\*

\*\*Aufgabe\*\*: Erstelle eine Azure Function, die auf Nachrichten in einer Azure Service Bus Warteschlange hört.

- Erstelle eine Service Bus-Warteschlange.

- Erstelle eine Azure Function, die auf Nachrichten in der Warteschlange reagiert.

- Sende eine Nachricht an die Warteschlange und überprüfe, ob die Azure Function die Nachricht verarbeitet.

\*\*Zusatzaufgabe\*\*:

- Füge eine Dead-Letter-Warteschlange hinzu und überprüfe, wie die Nachrichten dort landen, wenn sie nicht verarbeitet werden können.

---

Diese Aufgaben decken eine Vielzahl von Szenarien ab, bei denen verschiedene Azure-Ressourcen miteinander kommunizieren. Sie sind ideal, um ein tiefes Verständnis für Netzwerke, Sicherheitskonfigurationen und die Interaktion zwischen den verschiedenen Azure-Diensten zu entwickeln. Viel Erfolg beim Üben!