**Azure App Service – AZ-104 Training Guide**

**1. What is Azure App Service?**

Azure App Service is a **fully managed Platform as a Service (PaaS)** that enables you to build, deploy, and scale **web apps, REST APIs, and mobile backends** without managing infrastructure. It abstracts server management, patching, and scaling, allowing developers to focus on application logic.

**Key Features**

* **Multi-language support:** .NET, Java, Node.js, Python, PHP.
* **OS flexibility:** Runs on **Windows or Linux**, supports custom containers.
* **Global reach:** Deploy apps in multiple Azure regions.
* **Integrated DevOps:** CI/CD with GitHub Actions, Azure DevOps.
* **Built-in security:** SSL/TLS, authentication, and compliance (ISO, SOC, PCI).
* **Scalability:** Auto-scale based on demand.
* **High availability:** SLA-backed uptime, zone redundancy.
* **Networking:** VNet integration, private endpoints.
* **Monitoring:** Application Insights, Azure Monitor.

**2. App Service Plans & Pricing**

An **App Service Plan** defines:

* **Region** (where your app runs).
* **Compute resources** (CPU, memory).
* **Pricing tier** (Free, Shared, Basic, Standard, Premium, Isolated).

**Pricing Tiers**

* **Free/Shared:** For testing and learning (no SLA).
* **Basic:** Low-traffic apps, no auto-scale.
* **Standard:** Production-ready, supports auto-scale and deployment slots.
* **Premium:** High performance, advanced networking, VNet integration.
* **Isolated (App Service Environment):** Dedicated VNet, high security, large scale.

**Scaling Options:**

* **Vertical (Scale Up):** Move to higher tier for more CPU/RAM.
* **Horizontal (Scale Out):** Add more instances; supports **auto-scaling** based on metrics.

**3. Deployment Options**

Azure App Service supports multiple deployment methods:

* **Azure Portal:** Quick manual deployment.
* **Azure CLI / PowerShell:** Scripted deployments.
* **CI/CD Pipelines:** GitHub Actions, Azure DevOps.
* **FTP / WebDeploy:** For manual file uploads.
* **Containers:** Deploy custom Docker images.

**Deployment Best Practices**

* Use **Deployment Slots** (Staging, Production) for zero-downtime deployments.
* Enable **Continuous Deployment** from GitHub or Azure Repos.
* Use **Kudu** for advanced deployment scenarios.
* Avoid deploying directly to production; **swap slots** instead.

**4. Security & Authentication**

Azure App Service provides **built-in authentication and authorization** (Easy Auth):

* Supports **Microsoft Entra ID**, Google, Facebook, GitHub, and OpenID Connect.
* No code changes required; configure via portal.
* Enforces **HTTPS** automatically when enabled.
* Integrates with **Managed Identity** for secure resource access.
* Combine with **RBAC** and **App Service Access Restrictions** for granular control.

**5. Backup & Restore**

* **Supported in Basic, Standard, Premium, and Isolated tiers**.
* **Automatic Backups:** Hourly, retention for 30 days.
* **Custom Backups:** On-demand or scheduled, stored in Azure Storage.
* **Restore Options:** Overwrite existing app, restore to a new app or slot.
* **Best Practice:** Restore to a **staging slot** first, then swap to production.

**6. Monitoring & Diagnostics**

* **Azure Monitor:** Collects metrics and logs.
* **Application Insights:** Tracks performance, failures, and user behavior.
* **App Service Diagnostics:** Built-in troubleshooting tool for performance and availability.
* **Log Streaming:** Real-time logs for debugging.
* **Alerts:** Configure alerts for CPU, memory, and response time.

**7. Networking & Integration**

* **VNet Integration:** Outbound access to resources in a virtual network.
* **Private Endpoints:** Restrict inbound traffic to private IPs.
* **Hybrid Connections:** Connect to on-premises resources.
* **Service Endpoints:** Secure access to Azure services.
* **App Service Environment (ASE):** Fully isolated environment for high-security workloads.

**8. Best Practices for AZ-104 Exam**

* Understand **App Service Plan tiers** and when to use each.
* Know how to **configure deployment slots** and perform **slot swaps**.
* Be familiar with **authentication options** and **Easy Auth**.
* Practice **scaling operations** (manual and auto-scale).
* Learn **backup and restore steps**.
* Use **Application Insights** for monitoring and troubleshooting.
* Understand **VNet integration** and **App Service Environment** scenarios.
* Review **cost optimization strategies** (Reserved Instances, Savings Plans).