**Deployment and DevOps**

Guided by Microsoft on Coursera

**Notes during lessons:**

* Virtual machines are software-based simulations of physical computers that run an operating system and applications independently from the host hardware.
* App Services are a fully managed Platform-as-a-Service, or PaaS, provided by Azure for building, deploying, and scaling web apps and APIs.
* App Services are a fully managed Platform-as-a-Service, or PaaS, provided by Azure for building, deploying, and scaling web apps and APIs.
* Serverless computing is a cloud computing model where developers can run applications without managing servers.
* In the context of Azure Cloud, Resource Groups are logical containers that help you organize and manage related resources, such as virtual machines, storage accounts, and databases, to be managed as a single unit.
* Auto-scaling is a cloud computing feature that automatically adjusts computing resources based on real-time demand.
* Load balancers are tools that distribute incoming network traffic across multiple servers to ensure no single server gets overwhelmed.
* Service Level Agreements are formal commitments from a service provider that defines the expected level of service, such as uptime, performance, or support response times.
* Environment variables are key-value pairs used by an operating system to store configuration settings that applications and processes can access.
* To begin configuring environment variables, navigate to your project's directory in the Visual Studio Code terminal and install the required library by running .net add package .netenv.
* Then, inside the .env file, add variables such as apikey equals 12345secur.
* Finally, in your program.cs file, use the env.load method to load the environment variables from the .env file.
* Dependency management is the process of adding, updating, and maintaining extra libraries and packages, often using tools like NuGet.
* For example, in VSCode terminal, you can run .net add package newtonsoft.json to add a package.
* And to ensure all dependencies are up to date, run .net restore.
* Key configuration tasks: Enviroment variables, Dependency management, Build configuration.
* A build configuration is a set of predefined settings and rules that dictate how a software project is compiled, packaged, and deployed.
* For example, in the VSCode terminal, run .net build to generate the compiled files in your project slash bin folder.
* In the context of Azure App Services, application settings are key value pairs used to configure app behavior without modifying code.
* Connection strings are strings of text used to specify how an application connects to external resources like databases, storage accounts, or message queues.
* Containers are compact, self-contained units that include an application and everything it needs to run, such as libraries and system tools.
* Containerization is a method of packaging an application together with all its dependencies, libraries, and configuration files into a single container.
* Docker is a platform that enables developers to build, package, and run applications in containers.
* Key Deployment Tools: Azure App Service, Azure CLI, Github Action, Integrated Terminal in local.
* The Integrated Terminal in VS Code or Visual Studio Code is a built-in terminal that lets you write, test, and deploy your applications from a single workspace.
* Azure Monitor is a cloud-based service that collects, analyzes, and visualizes transmitted data from applications and infrastructure to enhance performance and reliability.
* Application Insights is a feature of Azure Monitor that helps developers track application performance, detect anomalies, and diagnose issues in real time.
* Autoscaling is the process of automatically adjusting computing resources based on demand to ensure optimal performance and financial cost efficiency.
* Azure Active Directory is a cloud-based identity and access management service that helps organizations securely manage user authentication and access to resources.
* Azure Active Directory is a cloud-based identity and access management service that helps organizations securely manage user authentication and access to resources.
* Multi-factor authentication is a security feature that requires users to verify their identity using multiple authentication methods before accessing an application.
* Multi-factor authentication controls who can log in, but role-based access control defines what users can do once they're inside of an application.
* Encryption at rest is the process of encoding data while it's stored on a disk, database, or other storage medium to protect it from unauthorized access.
* Encryption in transit secures data by encrypting it while it moves between systems, ensuring it cannot be intercepted by unauthorized users.
* Using TLS version 1.3 is a good practice for encrypting data in transit.
* Network security groups are a set of rules used in cloud environments, like Microsoft's Azure, to control inbound and outbound network traffic to resources.
* Azure Security Center is a Cloud security solution that helps protect your Cloud and online premises systems by providing security posture management and advanced threat protection.
* Compliance Manager is a Microsoft tool that helps organizations assess, manage, and improve compliance with industry standards, regulations, and company policies.
* Security audits are systematic evaluations of an organization's IT infrastructure, policies, and procedures to assess security risks and ensure compliance with industry standards.
* Continuous Integration is the practice of automatically merging code into source control tools like GitHub and testing changes frequently to ensure stability and avoid integration issues.
* Continuous Deployment is a software development practice where code changes that pass automated tests are automatically deployed to production without manual intervention.
* CI CD allows teams to improve update speed by automating tasks like code testing and integration.
* The CI CD process improves quality by integrating and delivering new code much more frequently than without this process.
* CI CD also promotes better collaboration by simplifying integration processes and minimizing conflicts between developers.
* A final benefit of CI CD pipelines is that they can handle errors using rollback mechanisms.
* Source Control is a system that tracks and manages changes to code, documents, or other files over time.
* Build Automation is the process of using scripts or tools to automatically compile, package, and deploy software applications without manual intervention.
* Testing Automation involves using specialized software tools to execute tests, compare actual outcomes with expected results, and report findings without human intervention.
* Deployment Automation is the process of automatically releasing and deploying applications to various environments such as development, testing, and production without manual intervention.
* In the context of GitHub Actions, workflows are automated processes defined in YAML files that specify how and when to run CICD tasks.
* A job is a set of steps in a workflow that are executed on the same server that runs your workflow.
* Steps are individual tasks or commands that are executed as a part of the job within a workflow.
* In the context of GitHub Actions, an event is a specific activity in the repository that triggers a workflow to run.
* Azure Pipelines is a cloud-based service that automates the build, test, and deployment processes for applications, supporting continuous integration and continuous delivery, or CICD.
* Azure Repos is a set of version control tools that allows teams to manage their code using tools like Git.
* Azure Artifacts is a universal package repository service that enables developers to manage and share code dependencies and packages from a centralized source.
* Azure Boards is a work management tool within Azure DevOps that helps teams plan, track, and manage software development projects.
* In the context of CICD, alerts are typically generated by the CICD platform or monitoring tools integrated into the pipeline, and are automated notifications that inform developers of build failures, deployment issues, or performance concerns.
* Dashboards are software tools or built-in features within CICD platforms that provide a centralized and visual representation of key metrics, build statuses, deployment progress, and system performance.
* Audit logs are software tools or built-in features within CICD platforms that track and record all significant actions within the pipeline.
* In back-end development, real-time monitoring refers to continuously tracking the performance, health and status of various application server-side components.
* Log monitoring is the process of collecting, analyzing, and tracking log data from applications, servers, databases, and network devices in real-time.
* Application performance monitoring is a set of practices for monitoring and managing the performance and availability of software applications.
* Application Insights is an application performance monitoring service in Azure Monitor that helps developers detect, diagnose, and optimize application performance and reliability.
* Centralized logging is a system that collects, aggregates, and stores logs from multiple sources such as applications, servers, and cloud services into a single platform for easier monitoring and analysis.
* Structured logging formats logs into key-value pairs, allowing each log entry to be easily queried, filtered, and analyzed.
* Retention policies are rules that define how long data, such as logs, backups, or documents, should be stored before being archived or deleted.
* Azure Automation is a cloud-based automation platform that enables users to automate repetitive tasks, manage configurations, and orchestrate workflows across Azure and hybrid environments, like those that include both on-premise data centers and cloud-based resources.
* Azure Logic Apps is a cloud-based service that enables users to automate workflows and integrate applications, data, and services across cloud and on-premises environments.
* Azure Runbooks can be used to automate frequent tasks through predefined scripts, tasks such as system maintenance, incident response, and cloud resource management.
* Azure Functions is a serverless computing service that enables developers to run event-driven code without managing infrastructure.
* Automated patching is the process of automatically applying software updates, security patches, and bug fixes to systems, applications, and infrastructure without manual intervention.
* Service updates refers to enhancements, bug fixes, security patches, and new features released by a service provider to improve the functionality, security, and performance of a cloud service or software platform.
* Patch compliance refers to the process of ensuring that all systems, applications, and devices within an organization are up-to-date with the latest security patches and software updates.
* Infrastructure as Code is the practice of managing and provisioning computing infrastructure through machine-readable configuration files rather than manual processes.
* Configuration management is the process of automating system setup and ensuring uniform configurations across multiple environments.
* Rollback strategies are plans or processes for reverting a system, application, or infrastructure to its previous stable state in case of failure or issues during an update, deployment, or change.
* Build automation refers to automatically compiling, testing, and preparing code for deployment whenever changes are made.
* Test automation refers to the practice of automatically running tests such as unit tests, integration tests, and end-to-end tests whenever code is committed or deployed.
* Deployment automation refers to the process of automatically deploying code to various environments, such as development, staging, and production, as part of the CI/CD pipeline.
* Pipeline configuration refers to setting up and defining the steps or stages in a CI/CD pipeline.
* Error detection is the process of identifying and recognizing faults, inconsistencies, or issues in data transmission, software execution, or system operations.
* Dependency management is the process of handling external libraries, frameworks, and packages that a software project relies on to function correctly.
* Version control conflicts occur when multiple contributors make conflicting changes to the same file or codebase, preventing automatic merging.
* Performance monitoring is the process of tracking, analyzing, and optimizing the performance of applications, systems, or networks to ensure they operate efficiently and reliably.
* In the context of DevOps, infrastructure automation refers to the use of tools and scripts to automatically manage and provision IT infrastructure, such as servers, networks, databases, and storage.
* In the context of DevOps, monitoring automation refers to the use of automated tools and processes to continually track the performance, availability, and health of systems, applications, and infrastructure.