AZURE DEVOPS CODING ASSESSMENT

NAME-SWARNA LATHA GUVVALA

DATE-12/01/2024

EXPLAIN PROCESS OF AZURE DEVOPS CICD PIPELINE?

The process of an Azure DevOps CI/CD pipeline can be explained in several stages:

Build Definition: In this stage, a build definition is created. It includes tasks like fetching code from a repository, building the application, and running unit tests. The build definition can be customized according to the application's requirements.

Continuous Integration (CI): Whenever code is pushed to the repository, Azure DevOps triggers a build. The CI process is designed to compile the code, run tests, and generate any necessary outputs like compiled code, minified files, or packages.

Release Definition: Once the build process is successful, a release definition is created. This definition includes tasks like deploying the application to a test environment, validating the deployment, and promoting the application to a production environment.

Continuous Deployment (CD): After the build process, the CD process begins. This involves automatically deploying the application to various stages, such as testing, staging, and production. The deployment process may involve tasks like copying files to the target servers, executing scripts, or restarting services.

Continuous Feedback and Monitoring: The CD process includes steps to gather feedback from the application's end-users. This feedback is used to make informed decisions about the next stages of the CI/CD pipeline. Additionally, the application is monitored for its performance, functionality, and user satisfaction.

EXPLAIN AZURE SQL SERVER POOL INTEGRATION WITH AZURE SYNAPSE?

Azure Synapse enables organizations to process and analyse large volumes of data stored in Azure SQL Server pools. The integration of Azure Synapse with Azure SQL Server pools allows users to run queries, manage databases, and optimize query performance across multiple data sources.

Here is an explanation of the integration process:

Azure SQL Server Pool: First, create an Azure SQL Server pool by following the necessary steps. Once the server pool is created, create a database in the pool.

Azure Synapse Workspace: Next, create an Azure Synapse workspace. This workspace will serve as the hub for running analytical queries and managing your data.

Serverless SQL Pool: Within the Synapse workspace, create a serverless SQL pool. This pool allows to execute T-SQL queries without having to manage infrastructure.

Database Connection: Connect the serverless SQL pool to the Azure SQL Server pool database. This can be done using the Synapse Studio in the Azure portal.

Query Execution: After connecting the serverless SQL pool to the Azure SQL Server pool database, can run T-SQL queries on the data stored in the pool. The results of these queries can be viewed directly in the Synapse Studio.

Integration with Other Data Sources: Azure Synapse allows to integrate with other data sources like Azure Data Lake Storage and Cosmos DB. This enables to perform distributed data processing tasks and perform analytical queries on large volumes of data.

Performance Optimization: Azure Synapse provides various features to optimize query performance across data sources. This includes features like indexing, statistics management, and dynamic management views.