Cognizant Digital Nurture 4.0 – Week 5:

**M. Aditya Naidu  
4992389**

# Topic: Microservices

## 1. Introduction to Microservices

Microservices is an architectural style that structures an application as a collection of small, independent services. Each service is self-contained and responsible for a specific business capability.

## 2. Key Characteristics of Microservices

• Independent Deployment – Services can be deployed individually without affecting others.

• Scalability – Each service can be scaled independently.

• Technology Diversity – Each microservice can use different programming languages or databases.

• Resilience – Failure in one service doesn't impact the entire system.

• DevOps Friendly – Encourages continuous integration and delivery.

## 3. Example – Spring Boot Microservice

ProductServiceApplication.java

import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class ProductServiceApplication {  
 public static void main(String[] args) {  
 SpringApplication.run(ProductServiceApplication.class, args);  
 }  
}

Output: Launches the Product microservice.

ProductController.java

import org.springframework.web.bind.annotation.\*;  
  
import java.util.\*;  
  
@RestController  
@RequestMapping("/products")  
public class ProductController {  
  
 @GetMapping  
 public List<String> getProducts() {  
 return List.of("Laptop", "Mobile", "Tablet");  
 }  
}

Output: Exposes a REST endpoint returning a list of products.

## 4. Microservices vs Monolithic Architecture

|  |  |
| --- | --- |
| Monolithic | Microservices |
| Single deployable unit | Multiple deployable units |
| Tightly coupled components | Loosely coupled services |
| Scaling is challenging | Each service can be scaled independently |
| Harder to adopt new tech | Can use diverse tech stacks |