BDD Explained with Cucumber and Karate (With Examples)

1. What is BDD?

BDD (Behavior Driven Development) is an approach where we test the behavior of an application from a user or business perspective, not the internal code structure.

Real-Life Example:
ATM Machine behavior: - If correct PIN is entered → cash should be dispensed - If wrong PIN is entered → error message should be shown

BDD is written using Gherkin language:
Given ATM machine is ready When user enters correct PIN Then cash should be dispensed

2. BDD with Cucumber

Cucumber is a tool that implements BDD. In Cucumber, we write behavior in a feature file and actual automation code in step definition files.

Cucumber Feature File Example:
Feature: User API Scenario: Get user Given user API is available When user sends GET request Then response status should be 200

Cucumber Step Definition (Java):
@When("user sends GET request") public void sendRequest() { // API call code }

3. BDD with Karate

Karate also follows BDD, but it does not require step definitions. Everything (behavior + automation) is written in one feature file.

Karate Feature File Example:
Feature: User API Scenario: Get user Given url 'https://reqres.in/api/users/2' When method get Then status 200

4. Scenario Outline (Data Driven BDD in Karate)

Scenario Outline: Get user by id Given url 'https://reqres.in/api/users/' When method get Then status Examples: | id | status | | 1 | 200 | | 2 | 200 | | 23 | 404 |

5. Key Difference Summary

BDD is the approach. Cucumber and Karate are tools. Cucumber needs step definitions, while Karate allows direct API testing inside feature files.

Interview Tip: Karate follows pure BDD without step definitions, making tests more readable, faster to write, and easier to maintain.