📌 Part 1: Theoretical Analysis

1. Short Answer Questions (Written in Report PDF)

Q1: AI-Driven Code Generation Tools (GitHub Copilot)

- How it reduces time:
  - Autocompletes repetitive code
  - Suggests entire functions based on comments
  - Reduces debugging time with smarter suggestions
- Limitations:
  - May generate insecure or inefficient code
  - Lacks deep contextual understanding
  - Potential licensing/copyright issues

Q2: Supervised vs. Unsupervised Learning in Bug Detection

Aspect	**Supervised Learning**	**Unsupervised Learning**
Data Needed	Labeled bug reports	Raw logs/code
Use Case	Classifying known bugs	Detecting anomalies
Accuracy	Higher for known bugs	Finds novel issues

Q3: Bias in UX Personalization

- Why it matters:
  - AI may reinforce stereotypes (e.g., gender-based recommendations)
  - Could exclude underrepresented users
- Example: Amazon’s biased hiring tool

2. Case Study Analysis (AI in DevOps)

- How AIOps improves deployment:
  - Predictive Failure Detection (e.g., log analysis)
  - Automated Rollback (if deployment fails)
- Examples:
  - Netflix: Uses AI to detect anomalies in microservices
  - Google: AI-powered canary deployments