3. Build and release pipelines
└─ **3.4 Deployment strategy and execution**
└─ **3.4.1 Blue green, canary, ring, progressive exposure, A/B**

1. What is blue-green deployment and when should it be used?
2. What is canary deployment and what problem does it solve?
3. What is a ring deployment in Azure DevOps?
4. How does progressive exposure differ from blue-green and canary deployments?
5. What is A/B testing in deployment strategies?
6. How do you minimize downtime using blue-green deployment in Azure?
7. How can Azure DevOps Pipelines implement canary releases?
8. What are best practices for monitoring during a canary or ring deployment?
9. How do feature flags support progressive exposure and A/B testing?
10. What are the rollback strategies for blue-green and canary deployments?

1. What is blue-green deployment and when should it be used?
Blue-green deployment runs two identical production environments ("blue" and "green"). Traffic switches from blue to green after validation, minimizing downtime and risk. Use when you require near-zero downtime and easy rollback.

2. What is canary deployment and what problem does it solve?
Canary deployment releases new changes to a small subset of users first. It mitigates risk by exposing only a fraction of traffic to new changes. If issues are detected, rollback is quick and impact is limited.

3. What is a ring deployment in Azure DevOps?
A ring deployment rolls out releases in controlled stages ("rings"), each representing a user or environment group. Inner rings (e.g., internal users) get updates first, followed by broader rings (e.g., all users). This enables staged rollout and feedback at each step.

4. How does progressive exposure differ from blue-green and canary deployments?
Progressive exposure gradually increases the percentage of users exposed to a new release based on real-time feedback. Unlike blue-green (all at once) and canary (fixed groups), exposure is dynamically adjusted, allowing rapid rollback or pause if issues occur.

5. What is A/B testing in deployment strategies?
A/B testing directs user segments to different application versions simultaneously (A vs. B) to compare behavior or feature impact. It's used to evaluate new features or UI changes by analyzing real user metrics before full rollout.

6. How do you minimize downtime using blue-green deployment in Azure?
Deploy to the "green" slot/environment, validate, then use Azure Traffic Manager or App Service swap to instantly redirect production traffic. This ensures seamless cutover with minimal or no downtime.

7. How can Azure DevOps Pipelines implement canary releases?
Use deployment jobs and environment approvals to release to a small subset of instances or users first. Gradually increase scope by adding stages that deploy to more targets as health checks pass.

8. What are best practices for monitoring during a canary or ring deployment?

- Enable detailed telemetry and alerts (e.g., Application Insights).
- Monitor error rates, latency, and user experience in real-time.
- Automate rollback if metrics exceed defined thresholds.

9. How do feature flags support progressive exposure and A/B testing?

Feature flags allow you to enable or disable features for specific users or groups without redeploying code. They control exposure dynamically, supporting gradual rollout (progressive exposure) or randomized experiments (A/B testing).

10. What are the rollback strategies for blue-green and canary deployments?

For blue-green, switch traffic back to the previous environment. For canary, halt rollout and reroute all users to the stable version, removing problematic instances from the canary group.