2. Source control strategy
 └ **2.1 Branching and workflows**
   └ **2.1.2 Pull request workflows with policies**

---

1. What is a pull request (PR) workflow and why is it important in DevOps?
2. How do you enforce branch policies for pull requests in Azure DevOps and GitHub?
3. What types of status checks can be required for a PR?
4. How can you require code review approvals before merging a PR?
5. How do you prevent direct pushes to protected branches?
6. What are best practices for PR size and review process?
7. How do you automate checks and enforce policy compliance in PRs?
8. How can you enforce security and compliance checks in PR workflows?
9. What is auto-complete or auto-merge in pull request workflows?
10. How do you configure and use code owner policies in PR workflows?

---

1. **What is a pull request (PR) workflow and why is it important in DevOps?**

A PR workflow is a structured process to review, validate, and merge code changes. It ensures code quality, security, and team collaboration before integration.

---

2. **How do you enforce branch policies for pull requests in Azure DevOps and GitHub?**

Configure branch protection rules (*GitHub*) or branch policies (*Azure DevOps*) to require

- PR reviews,
- status checks,
- and other controls before merging.

---

3. **What types of status checks can be required for a PR?**

Common checks include

- successful CI builds,
- code coverage thresholds,
- security scans,
- and passing automated tests.

---

4. **How can you require code review approvals before merging a PR?**

Set branch protection to

- require one or more reviewer approvals,
- optionally require review from code owners or specific teams.

---

5. **How do you prevent direct pushes to protected branches?**

Enable branch protection rules or policies that restrict who can push and require all changes to go through PRs.

---

6. **What are best practices for PR size and review process?**

- Keep PRs small and focused,
- include clear descriptions,
- require meaningful reviews,
- and use checklists to ensure standards.

7. **How do you automate checks and enforce policy compliance in PRs?**
   - Integrate CI/CD pipelines,
   - configure required status checks,
   - use automated code analysis,
   - and enforce policies as blocking conditions for merging.

---

8. **How can you enforce security and compliance checks in PR workflows?**

Add automated security scanners (e.g., for secrets, dependencies),
require results as passing status checks before merge.

---

9. **What is auto-complete or auto-merge in pull request workflows?**

It allows PRs to be merged automatically when all policies and checks are satisfied, reducing manual intervention and wait time.

---

10. **How do you configure and use code owner policies in PR workflows?**

Define a CODEOWNERS file specifying responsible individuals or teams for files/folders. Require code owner review in branch protection settings for critical code areas.