2. Source control strategy
  └ **2.2 Repository Configuration and Maintenance**
      └ **2.2.3 Configure permissions, tags**

---

1. What are the permission levels available in Azure Repos and GitHub repositories?
2. How do you assign repository permissions in Azure DevOps?
3. How are permissions managed in GitHub (users, teams, roles)?
4. What is the difference between repository, branch, and object-level permissions?
5. How do you use branch protection rules to restrict actions?
6. What are best practices for granting repository permissions?
7. How do you configure repository tags in Git and Azure DevOps?
8. What are the use cases for tags in source control?
9. How do you protect or restrict tag creation and deletion?
10. How can you audit or review permission and tag changes in repositories?

---

1. What are the permission levels available in Azure Repos and GitHub repositories?
Azure Repos:
  - Reader
  - Contributor
  - Branch Creator
  - Project Administrator
  - Custom

GitHub:
  - Read
  - Triage
  - Write
  - Maintain
  - Admin

---

2. How do you assign repository permissions in Azure DevOps?
Go to *Project Settings → Repositories → Select repository → Security → Add users/groups → Assign roles or explicit permissions*.

---

3. How are permissions managed in GitHub (users, teams, roles)?
Assign roles at repository or organization level. Use teams to group users. Set roles for users or teams via repository Settings → Manage access.

---

4. What is the difference between repository, branch, and object-level permissions?
Repository: Control access to entire repo.
Branch: Restrict actions (push, merge) on specific branches.
Object: Limit access to individual files/paths (rare, typically via code owners).

---

5. How do you use branch protection rules to restrict actions?
In *Settings → Branches*, add branch protection rules (e.g., require PR reviews, restrict push, enforce status checks, prevent force pushes or deletions).

6. What are best practices for granting repository permissions?
Grant least privilege required, use teams/groups, avoid Admin unless necessary, periodically review access.

---

7. How do you configure repository tags in Git and Azure DevOps?
Create:
git tag v1.0 (local),
git push origin v1.0 (remote).
Azure DevOps: Use web UI or Git CLI; tags sync with repository.

---

8. What are the use cases for tags in source control?
Mark

- releases,
- important milestones,
- stable builds,
- deployment points.

---

9. How do you protect or restrict tag creation and deletion?
GitHub: Use branch protection for refs/tags/* with GitHub Enterprise.
Azure DevOps: Restrict who can create/delete tags via repository security settings.

---

10. How can you audit or review permission and tag changes in repositories?
Azure DevOps: Use Audit Logs (Organization settings).
GitHub: Use Audit Log in Enterprise/Organization settings; review git history for tag changes via git log --tags.