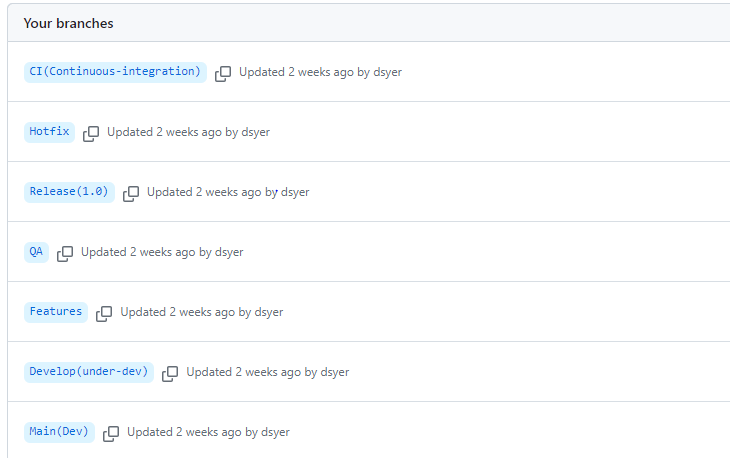
1. DevOps is a combination of software development (DEV) and operations (OPS) which is a set of tools, Practices(methods) and a cultural philosophy that automate and integrate the processes between software development and IT teams. It helps to evolve and improve products at faster pace than using traditional practices followed by organization.

Use Cases:

1. **CI/CD**: used for automating development pipeline allowing for continuous integration and deployment of code changes.
2. **Infrastructure-as-code** (Iaas): helps automating the provisioning and configuration of infrastructure, making it easier to manage and scale.
3. **Monitoring and alerting**: prepare monitoring and alerting systems to detect and respond to issues in real-time.
4. **Collaborative Development**: Collaboration among developers, testers, and operations teams. This collaboration leads to better code quality, faster issue resolution, and more efficient use of resources.
5. Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) computing power, and various services without direct active management by the user. It relies on sharing of resources to achieve consistency and typically uses a pay-as-you-go model, which can help in reducing capital expenses but may also lead to unexpected operating expenses for users.

Use Cases:

1. **Scalable Web Applications**: It is used to provide scalability, allowing applications to handle varying levels of traffic without major infrastructure changes.
2. **Data Storage and Analytics:** They providecost effective and also leverage cloud-based data analytics tools for processing and analyzing large datasets.
3. **Disaster Recovery and Backup**: They offer robust disaster recovery and backup solutions, ensuring data is protected and can be restored in case of unexpected events.
4. **IOT (Internet of Things)**: Cloud computing is essential for managing and processing data generated by IOT devices. They provide the necessary infrastructure and services to handle massive data streams.
5. **Development and Testing Environments**: It also offer machine learning and AI services that allow organizations to build, train, and deploy AI models without significant infrastructure investments.
6. Git Environment Branching Strategy for Spring-Pet clinic Application
7. Create Main Branch (DEV): This branch is for only production-ready code. It should only contain stable and tested code. Develop Branch is only for ongoing development. Adding new feature Branches will be created under this branch only.
8. Feature Branch: Developers create feature branches from develop for developing new features or making changes. Different types of branches created for different types of features.
9. Testing Branch (QA): Testers create branches from develop for writing and running test cases. They can be used for finding errors and creating test cases for better deployment.
10. Release Branches: After product ready we have to deploy it in a new release branch and can be named as like ‘release/1.0”
11. Hotfix Branches: These are used as critical resource branch which is used for updating issues occurred in main branch then merging it into main and develop branch
12. Continuous Integration (CI) Branch: Set up a CI/CD pipeline that automatically builds and tests every push to the develop branch.



Branches have been created for different purposes as mentioned above. We have to push the spring pet clinic into the main repository of created branches.

