

MERGING STRATEGY

OPTION 1:

To ensure that the `MY_TEST_BRANCH` is up-to-date and will safely integrate into `master`, follow these steps:

Step 1: Understand the state of branches

- Confirm the current state of `master` branch and `MY_TEST_BRANCH`
 - The `master` branch is at commit `4f19f63479` (tag: `v0.0.2`).
 - The `MY_TEST_BRANCH` is at commit `ec1b16e0b4`.

Step 2: Update `MY_TEST_BRANCH` with the latest `master`

- Rebase or merge the latest `master` into `MY_TEST_BRANCH` to make sure it runs with all the latest changes from `master`
- We can use rebase or merge strategy
 - Merge: Preserves the branch commit history and is safer when working in teams
 - Rebase: Creates a linear history and is cleaner but rewrites commit history, which can sometimes be an issue

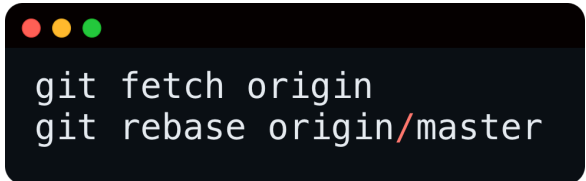
Rebase: `MY_TEST_BRANCH` onto `master`

- Switch to `MY_TEST_BRANCH`



```
git checkout MY_TEST_BRANCH
```

- Rebase onto `master`
 - This applies the changes from `MY_TEST_BRANCH` on top of the latest `master` commit



```
git fetch origin  
git rebase origin/master
```

- If there are conflicts, Resolve conflicts as needed

```
# Edit conflicted files, stage them, and continue
git add <file>
git rebase --continue
```

- To abort Rebase

```
git rebase --abort
```

Merge: `master` into `MY_TEST_BRANCH`

- Switch to `MY_TEST_BRANCH`

```
git checkout MY_TEST_BRANCH
```

- Merge `master` into the branch
 - This creates merge commit that preserves the history of both branches.

```
git fetch origin
git merge origin/master
```

- If there are conflicts, Resolve conflicts as needed

```
# Edit conflicted files and stage them
git add <file>
# Complete the merge
git commit
```

Step 3: Run tests on `MY_TEST_BRANCH`

- Run the entire regression test suite to verify the `MY_TEST_BRANCH` is compatible with the latest changes from `master`
 - Ensure all tests pass locally and in your CI/CD pipeline to confirm stability

```
npm run test # Or the appropriate command for your test suite
```

Step 4: Push and update MY_TEST_BRANCH

- If all tests pass and conflicts are resolved, push the updated MY_TEST_BRANCH to the remote repository



```
git push origin MY_TEST_BRANCH
```

Step 5: Verify MY_TEST_BRANCH branch on CI/CD

- Ensure that your CI/CD pipeline runs and passes all tests on the updated MY_TEST_BRANCH
- This step ensures that merging will not introduce issues in master branch

Step 6: Review and Create a Pull Request

- Create a pull request from MY_TEST_BRANCH to master in version control system
- Conduct code review to check for code issues before merging

Step 7: Merge into master

- If the PR checks are passed and there are no further issues, merge the PR into master

OPTION 2:

Step 1: Fetch the latest code from the master branch



```
git fetch origin
```

Step 2: Create a temporary branch to test the merge from `master` without impacting `MY_TEST_BRANCH`

```
git checkout MY_TEST_BRANCH  
git checkout -b TEMPORARY_TEST_BRANCH
```

Step 3: Merge the changes from `master` to the `TEMPORARY_TEST_BRANCH` to check for merge conflicts and ensure it works with the most recent code

```
git merge origin/master  
or  
git rebase origin/master
```

Step 4: Resolve conflicts as needed

```
git add <resolved-files> # Add resolved files  
git commit # Commit the resolved merge
```

Step 5: Run all tests

- Execute all tests locally to verify that the `TEMPORARY_TEST_BRANCH` works as expected after merging.

```
npm run test # Or the appropriate command for your test suite
```

Step 6: Verify `TEMPORARY_TEST_BRANCH` branch on CI/CD

- Ensure that your CI/CD pipeline runs and passes all tests on the updated `TEMPORARY_TEST_BRANCH`
- This step ensures that merging will not introduce issues in `master` branch



```
git push origin TEMPORARY_TEST_BRANCH
```

- Check your CI/CD pipeline to ensure all tests pass

Step 7: Update MY_TEST_BRANCH

- If all the test passes on the TEMPORARY_TEST_BRANCH, merge TEMPORARY_TEST_BRANCH to MY_TEST_BRANCH, to merge the latest changes from master



```
git checkout MY_TEST_BRANCH  
git merge TEMPORARY_TEST_BRANCH
```

Step 8: Merge MY_TEST_BRANCH to master

- Create a pull request from MY_TEST_BRANCH to master in version control system
- Conduct code review to check for code issues before merging
- If the PR checks are passed and there are no further issues, merge the PR into master