



Bab 2 — Git Branch & Workflow Lanjutan



Konsep Dasar Branch

Penjelasan	Contoh
Branch adalah timeline paralel di Git yang memungkinkan pengembangan fitur atau bugfix tanpa mengganggu branch utama.	main: A--B--C feature: \ D--E
Setiap branch bisa di- <i>merge</i> atau dihapus kapan pun.	—



Perintah Dasar Branch

Perintah	Fungsi
git branch <nama>	Membuat branch baru
git checkout <nama>	Pindah ke branch tertentu
git checkout -b <nama>	Membuat dan langsung berpindah branch
git switch <nama>	Versi baru dari checkout
git switch -c <nama>	Buat dan pindah branch (versi modern)



Mengubah & Menghapus Branch

Perintah	Fungsi
git branch -m <nama_baru>	Mengubah nama branch aktif
git push origin :<lama>	Menghapus branch lama di remote
git push origin <baru>	Push branch baru ke remote
git branch -d <nama>	Hapus branch lokal (aman)
git branch -D <nama>	Hapus branch lokal (paksa)
git push origin --delete <nama>	Hapus branch di remote

Merge Branch

Perintah	Fungsi
git checkout main	Pindah ke branch utama
git merge <branch>	Menggabungkan branch ke main
git merge --abort	Membatalkan merge jika konflik

Catatan:

Jika ada konflik, Git akan menandai bagian file seperti ini:

```
<<<<<<< HEAD
```

```
kode branch aktif
```

```
=====
```

```
kode branch yang digabung
```

```
>>>>>>> feature
```

Merge Conflict

Perintah	Fungsi
git add <file>	Tandai file setelah konflik diselesaikan
git commit	Selesaikan proses merge
git merge --abort	Batalkan merge sepenuhnya

Git Stash (Simpan Perubahan Sementara)

Perintah	Fungsi
git stash push -m "pesan"	Simpan perubahan sementara
git stash list	Lihat daftar stash
git stash show stash@{0}	Lihat isi stash
git stash apply stash@{0}	Ambil stash tanpa menghapusnya
git stash pop stash@{0}	Ambil + hapus stash
git stash clear	Hapus semua stash

Cherry Pick

Perintah	Fungsi
git cherry-pick <hash>	Ambil commit tertentu dari branch lain
git cherry-pick --continue	Lanjutkan setelah konflik
git cherry-pick --abort	Batalkan cherry pick

Tag (Menandai Commit / Versi)

Perintah	Fungsi
git tag <nama_tag> <hash>	Buat tag baru
git tag	Lihat daftar tag
git tag -d <nama_tag>	Hapus tag lokal
git push origin --delete <tag>	Hapus tag di remote
git checkout <nama_tag>	Pindah ke commit dengan tag tertentu

Rebase

Perintah	Fungsi
git rebase <branch>	Menulis ulang commit agar sejajar dengan branch lain
git rebase --continue	Lanjutkan rebase setelah konflik
git rebase --abort	Batalkan rebase
—	Rebase menghasilkan riwayat linear , berbeda dari merge yang bercabang

Merge Squash

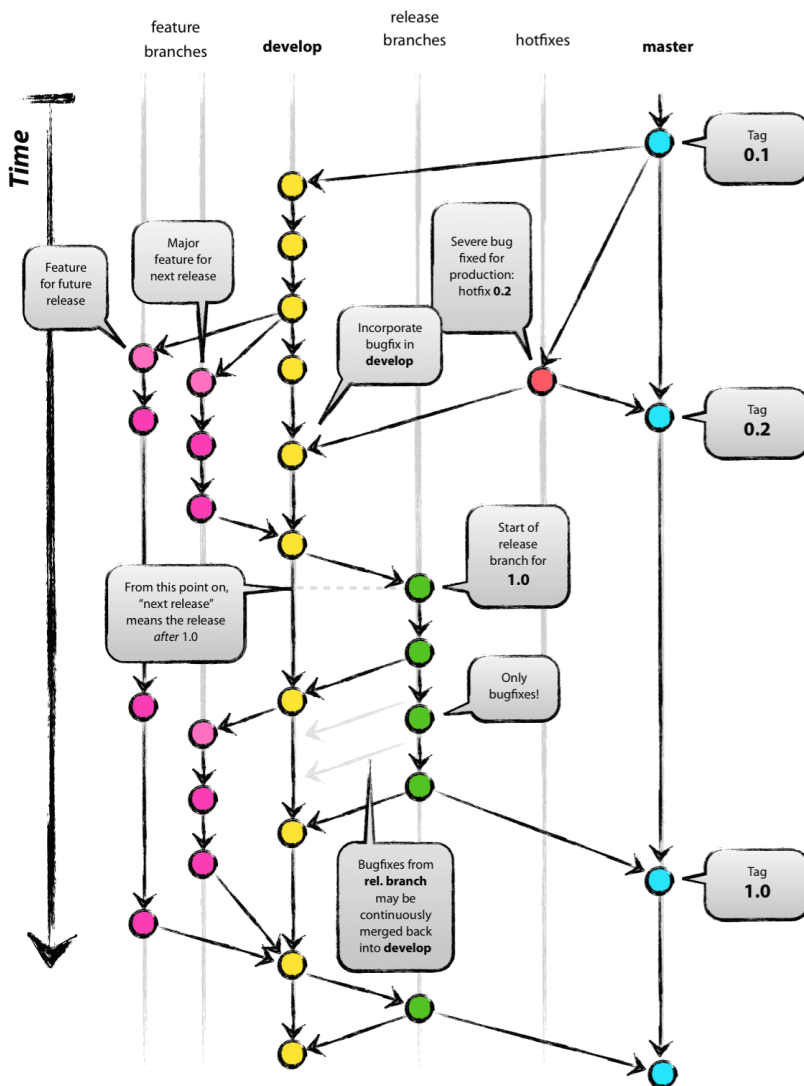
Perintah	Fungsi
git merge --squash <branch>	Gabungkan semua commit menjadi satu
git commit -m "pesan"	Commit hasil gabungan
—	Hasilnya satu commit rapi tanpa merge commit di log

💡 Pindah Branch Saat Ada Perubahan Belum Commit

Tujuan	Perintah
Simpan sementara	git stash
Simpan permanen	git add . && git commit -m "WIP"
Buang perubahan	git restore .

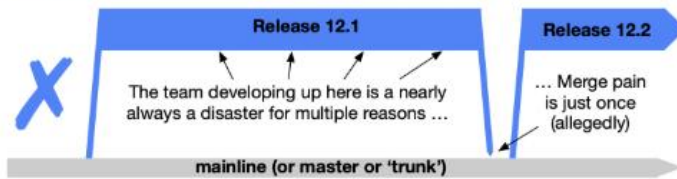
⚙️ Workflow di Dunia Industri

🌱 1. Gitflow Workflow

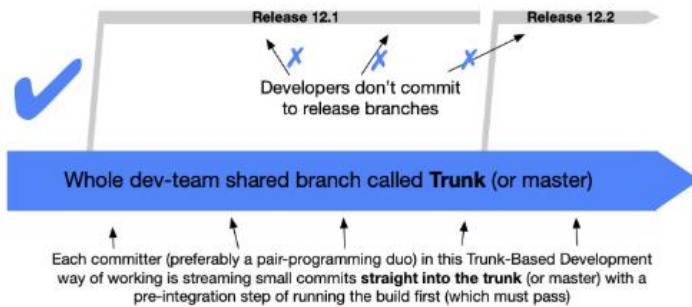


⚡ 2. Trunk-Based Development

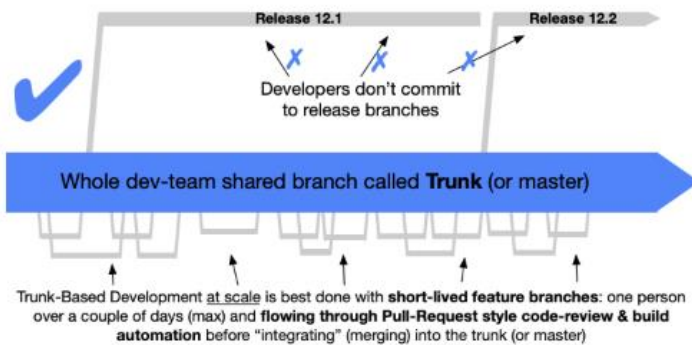
Shared branches off mainline/main/trunk are bad at any release cadence:



Trunk-Based Development For Smaller Teams:



Scaled Trunk-Based Development:



🍴 3. Forking Workflow

