



# Integrando com github

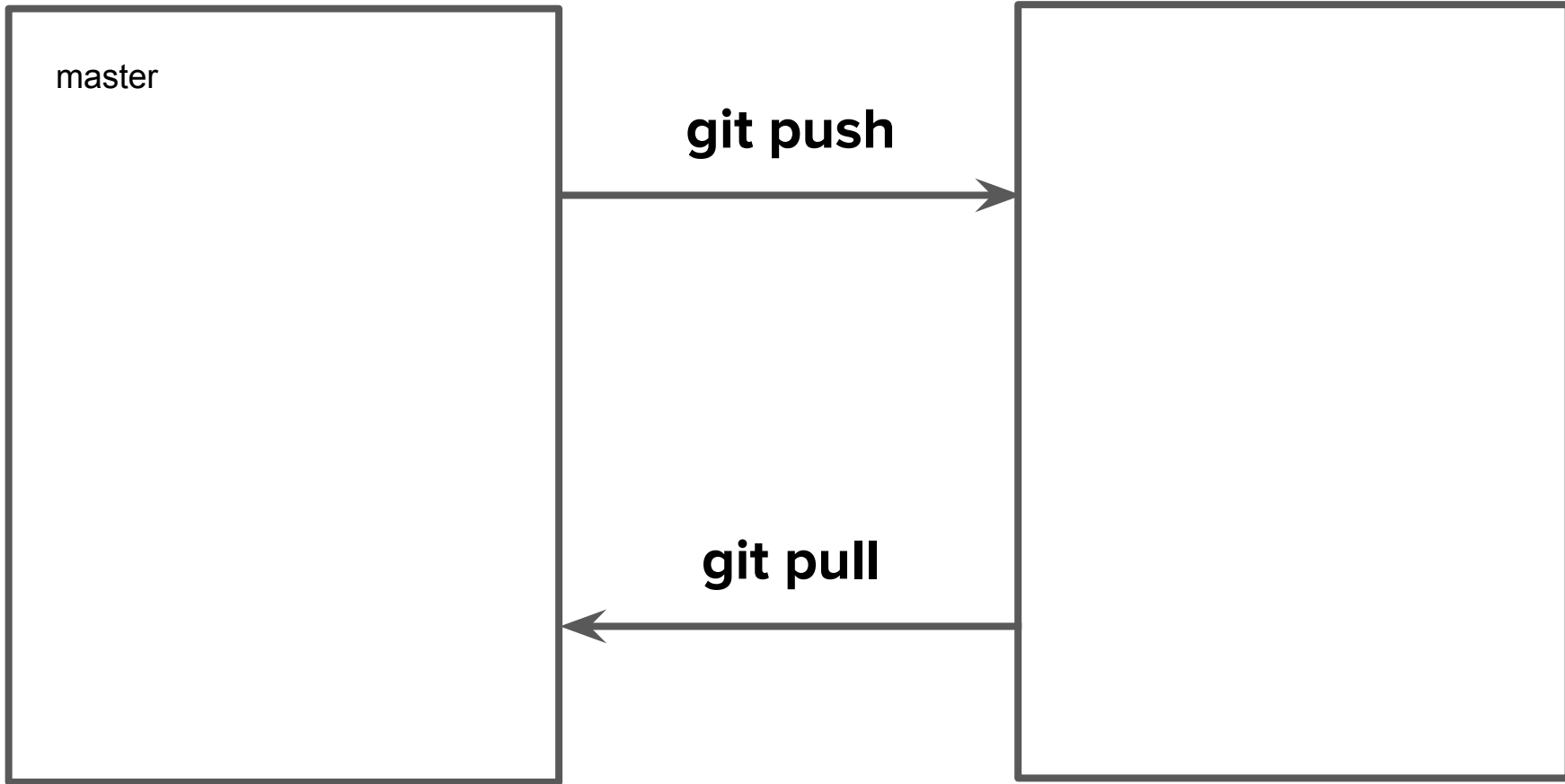
# Repositório Local

# Repositório Remoto

master

**git push**

**git pull**



# Comandos básicos de git

- git push (manda da sua máquina para o github)
  - git push -u origin {nome-da-branch}
- git pull (pega atualizações do github para sua máquina)

# Repositório Local

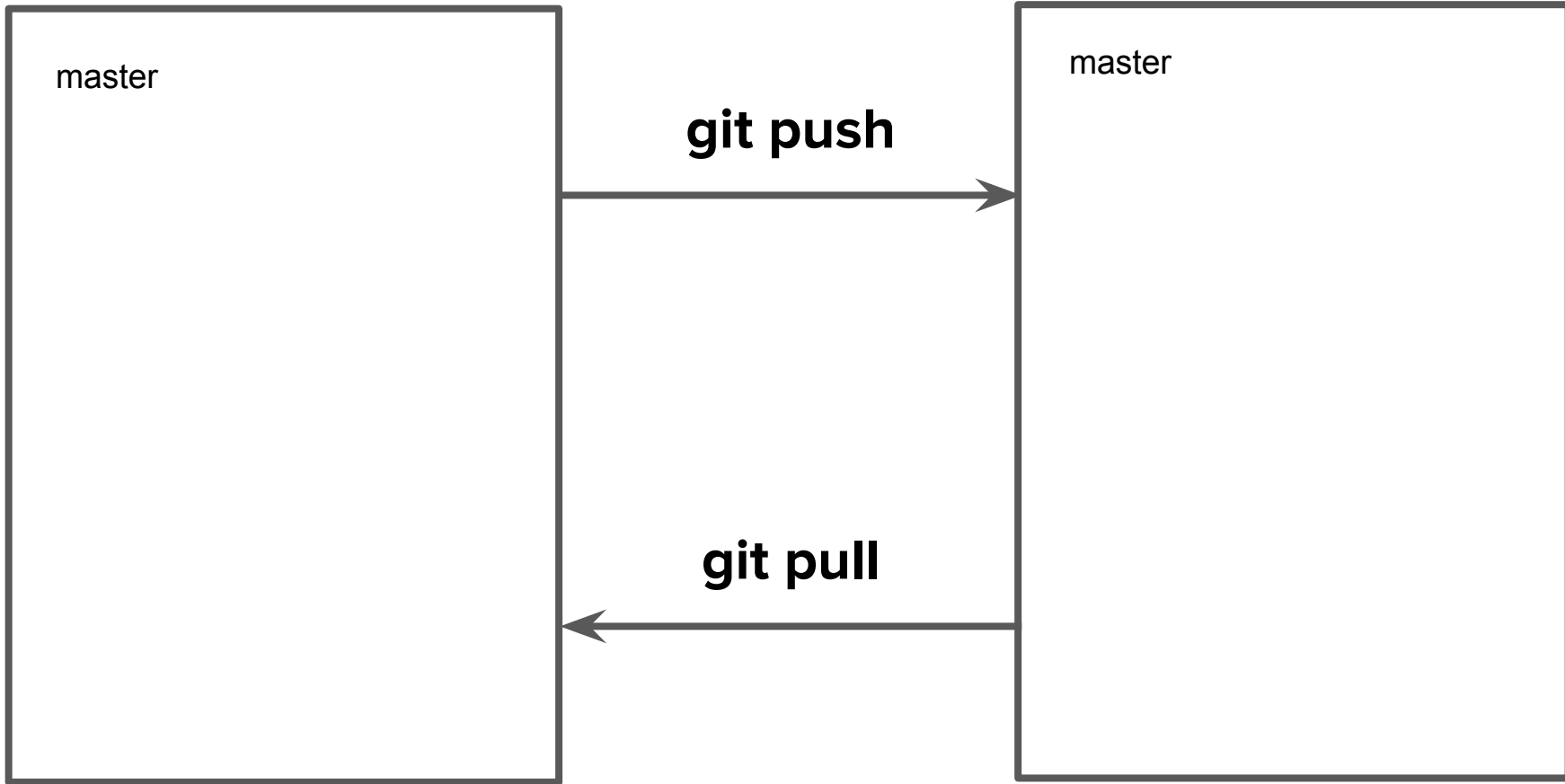
# Repositório Remoto

master

**git push**

master

**git pull**



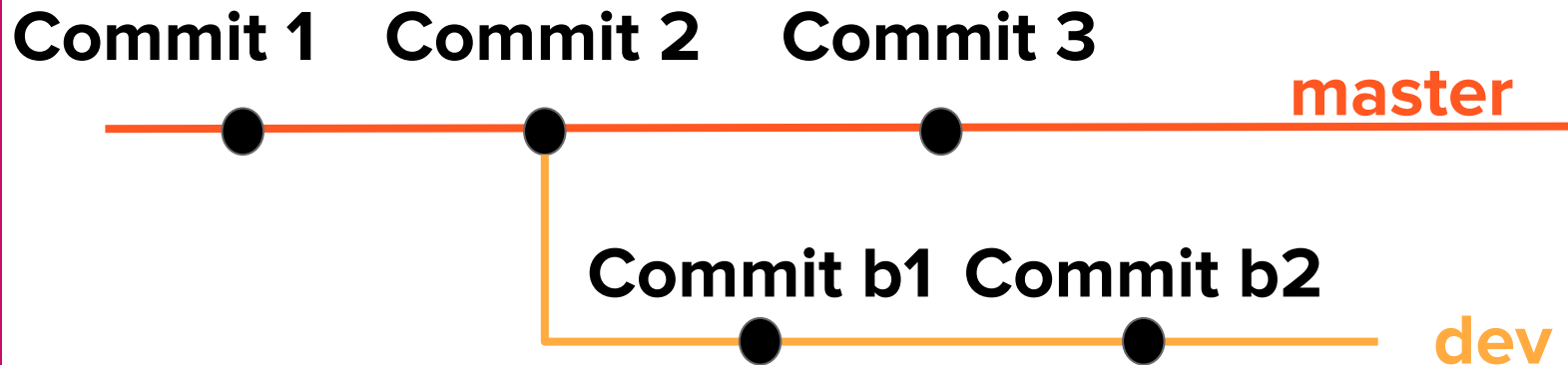
# Exercício

- Crie um novo repositório no github SEM um readme dessa vez
- Olhando para as instruções do github, busquem a instrução que começa com “git remote add origin {url}” e rodem essa instrução
- em seguida rodem: git push -u origin master



# Além do básico

# Branches



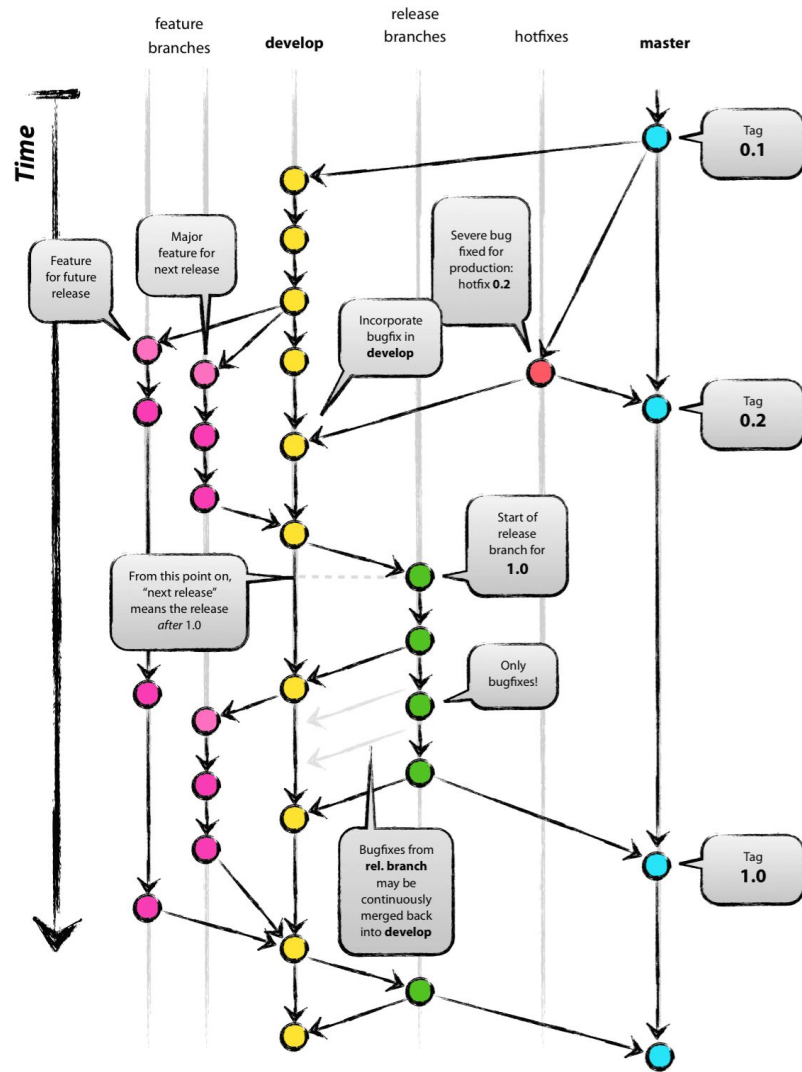
# Estratégia de branches

- Git Flow
- Github Flow
- Gitlab flow
- Your Flow

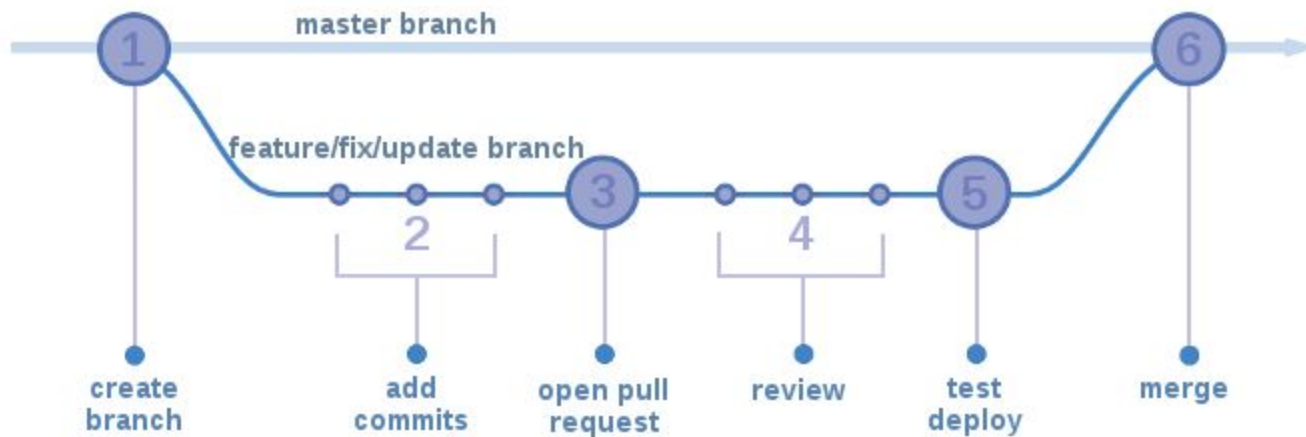


FASE 1 — FASE 2 — **FASE 3** — FASE 4 — FASE 5

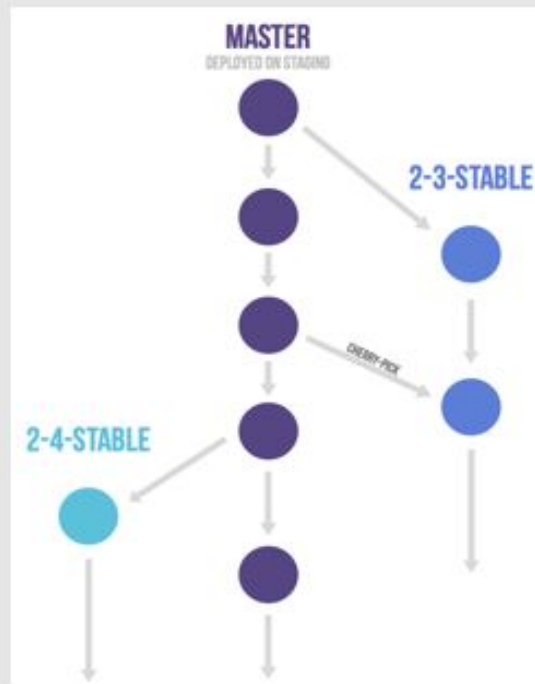
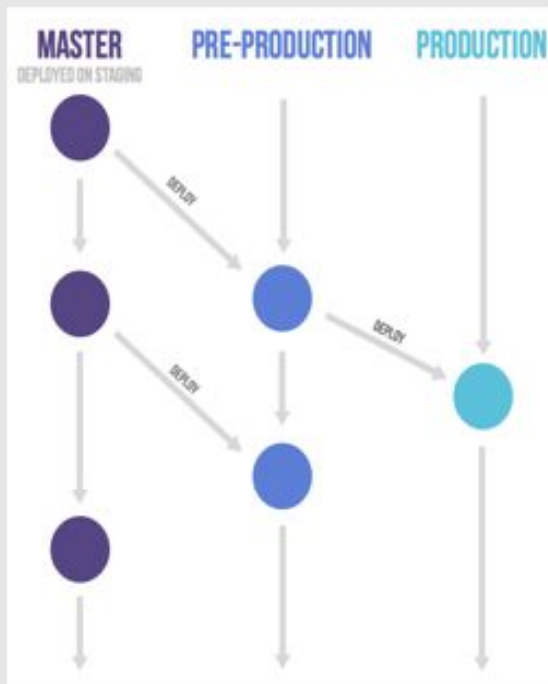
# Git Flow



# Github Flow



# Gitlab Flow



# Branches

- git branch: Informa a branch atual
- git branch {nome}: Cria uma branch, mas permanece na atual
- git branch -D {nome}: deleta uma branch
- git switch -c {nome}: Cria uma branch e vai pra ela
- git checkout -b {nome}: Cria uma branch e vai pra ela
- git switch {nome}: Para navegar entre branches
- git checkout {nome}: Para navegar entre branches

# Exercicio

- Usando o projeto da aula 2, Git-Renner-Aula02, crie 2 branches:
  - a. A branch de nome readme deve ser criada a partir do commit que adicionou o readme no projeto
  - b. A branch de nome web-files deve ser criada a partir do commit que adicionou os arquivos vazios.
- Na branch readme faça um novo commit, modificando o readme adicionando qual seu computador
- Na branch web-files crie um arquivo README com apenas um titulo # Read Me

## Repositório Local

master

readme

web-files

## Repositório Remoto

master

**git push**

**git pull**



# Branches

- `git push origin {branch_name}`
- `git push origin readme`

## Repositório Local

master

readme

web-files

## Repositório Remoto

master

readme

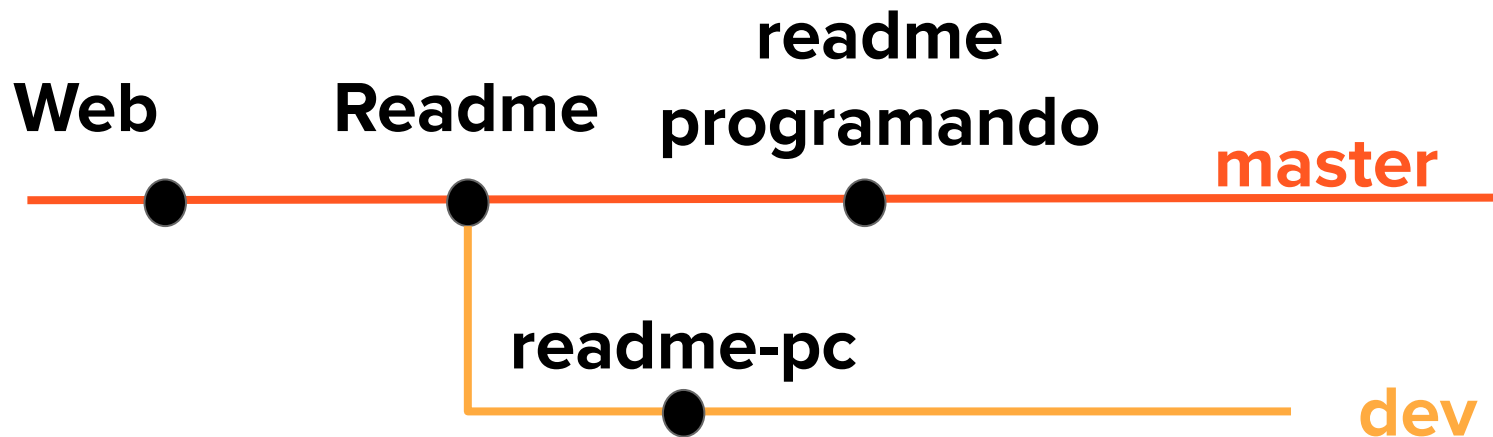
**git push**

**git pull**

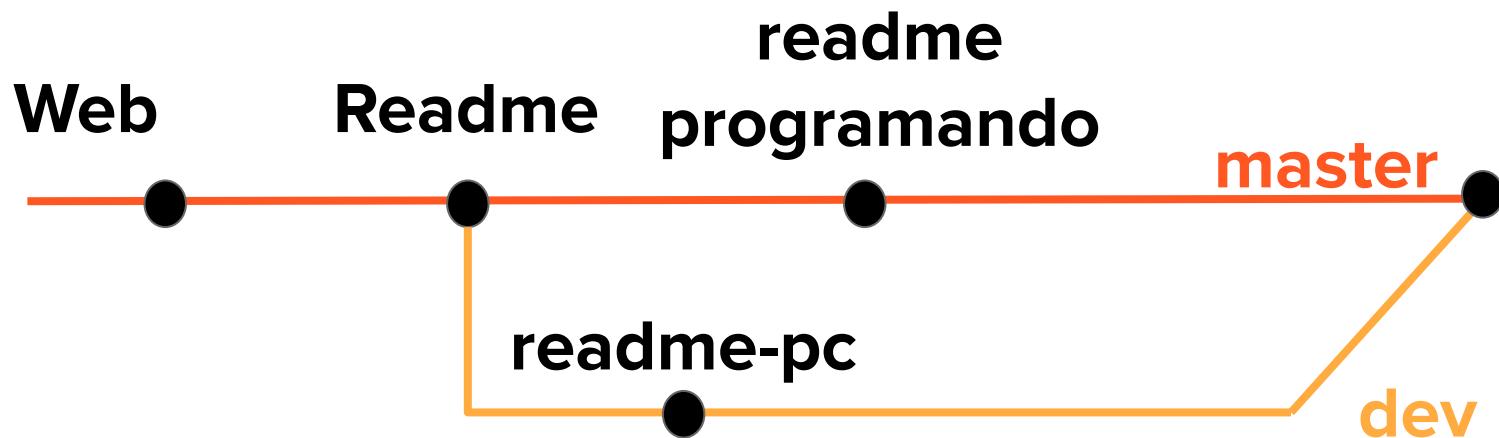




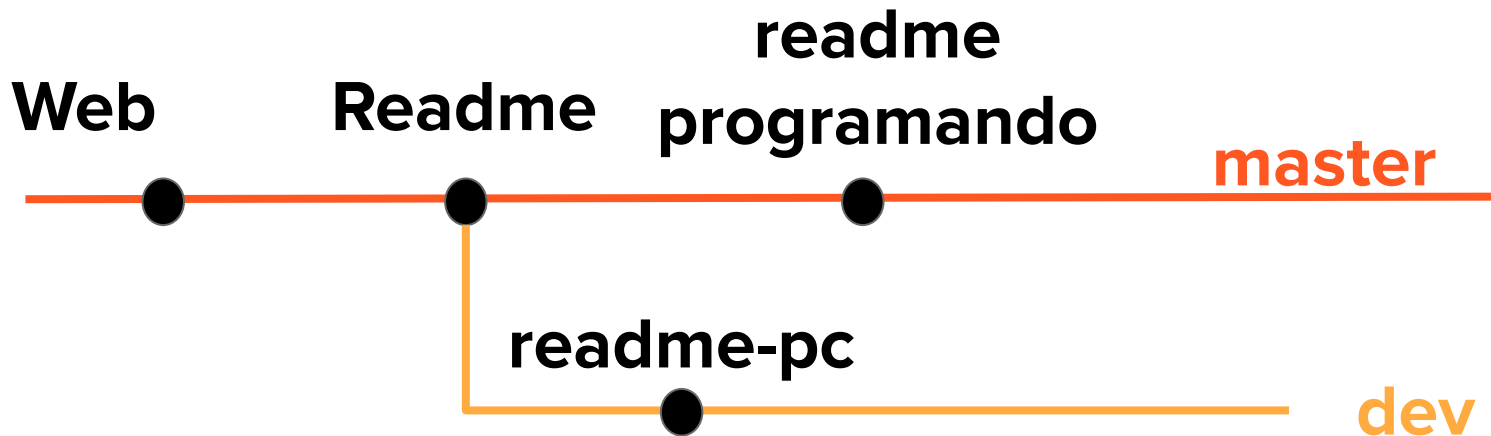
# Conflito



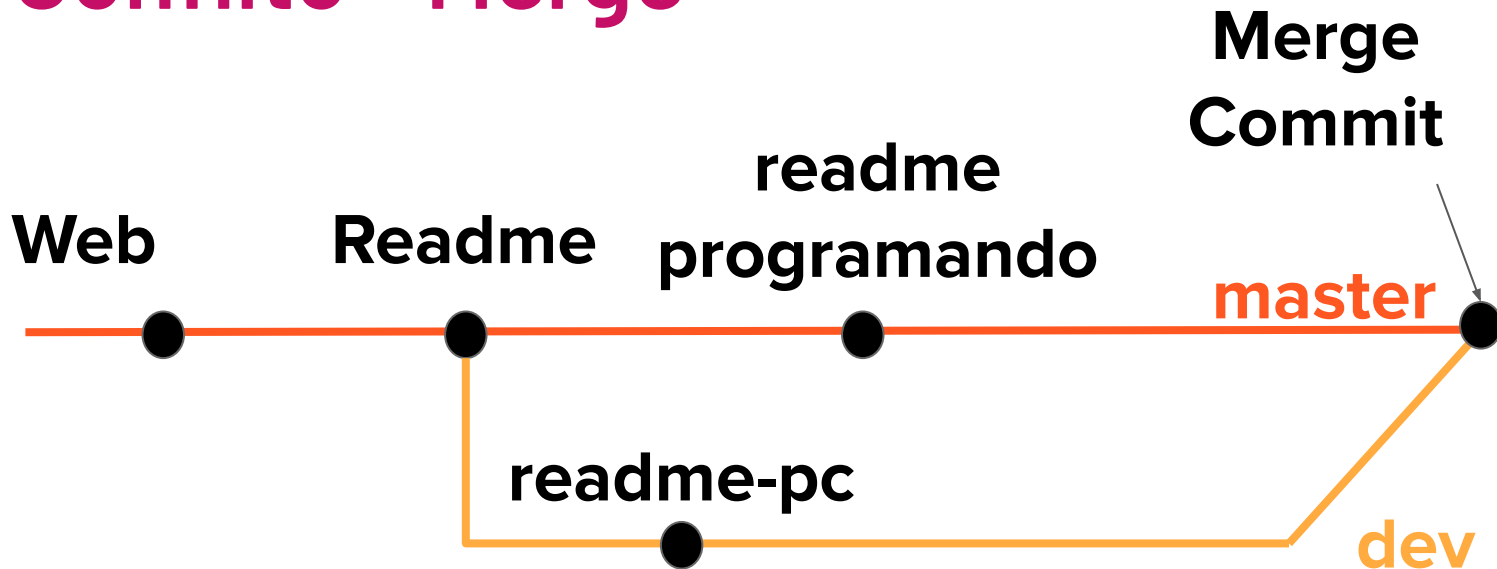
# Conflito



# Conflito - Merge



# Conflito - Merge

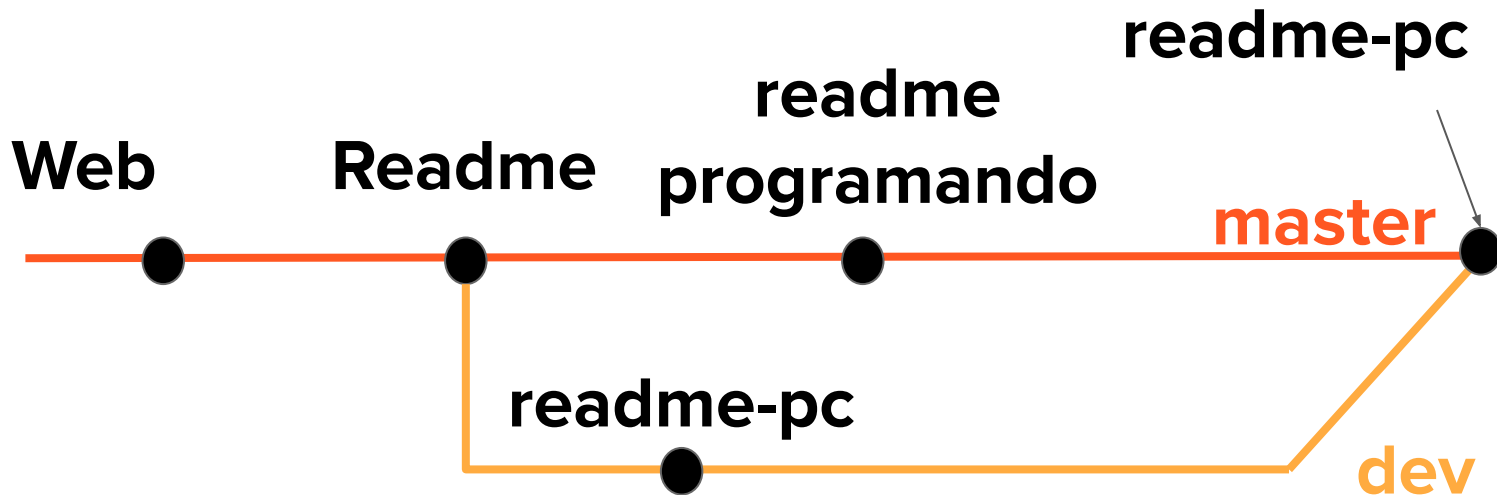


# Git - Merge

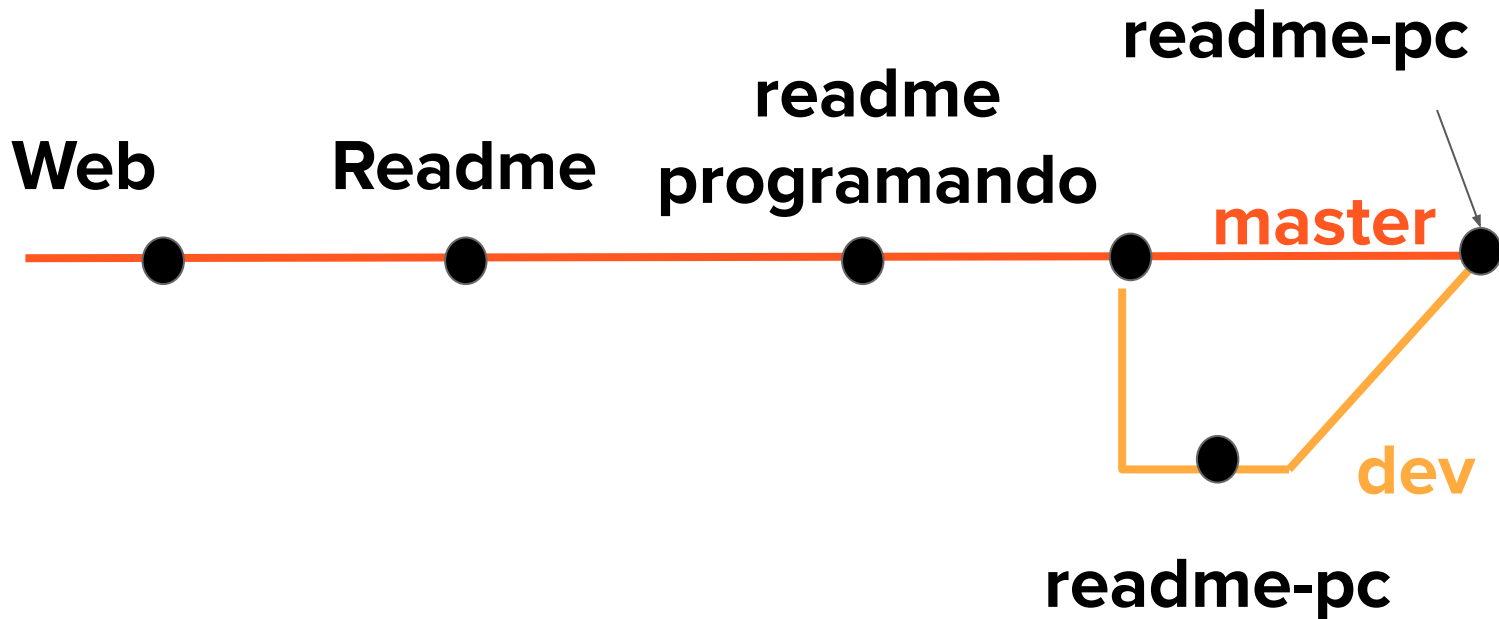
- Dentro da master digite  
git merge readme  
Isso vai tentar mergear a branch readme na branch master

Deu conflito? Por que? Como corrigir?

# Conflito - rebase



# Conflito - rebase



# Git - Rebase

- Dentro da web-files digite  
git rebase master
- Dentro da master digite  
git rebase web-files

Isso vai tentar fazer rebase da branch web-files dentro da branch master

Deu conflito? Por que? Como corrigir?



# Pull Request e Code Review

- Crie uma nova branch chamada feature1 e outra chamada feature 2 a partir da master
- Crie um novo commit na feature1 modificando mais uma vez o seu README.md adicionando o topico “# Feature 1” no final do arquivo
- Enviem essa branch para o projeto do Github de voces
- No Github criem um Pull Request da branch feature1 para a master
- Faça um Merge/Rebase/Squash

# Pull Request e Code Review

- Crie um issue no github
- Crie uma branch cujo nome contenha a numeracao do issue
- Faça um commit que contenha “Fixes #1” trocando 1 pela numeração do issue
- Faça um Pull request que também contenha a mesma frase do commit
- Faça um Merge/Rebase/Squash

# Pull Request e Code Review

- Voltem para a branch feature2 e criem um commit contendo o topico “#Feature2” no final do arquivo readme
- Crie um pull request a partir da branch feature2