Limpando arquivos untracked

* O comando **git clean** vai verificar e limpar arquivos que não estão sendo trackeados
* Ou seja, todos que você não utilizou **git add**
* Utilizado para arquivos que são gerados automaticamente, por exemplo, e atrapalham a visualização do que é realmente importante
* **git clean -f**
* Utilizar o **git add** antes do git clean para salvar oq deseja

Otimizando o repositório

* O comando **git gc** é uma abreviação para garbage collector
* Ele identifica arquivos que não são mais necessários e os exclui
* Isso fará com que o repositório seja otimizado em questões de performance

Checando integridade de arquivos

* O comando **git fsck** é uma abreviação de File System Check
* Esta instrução verifica a integridade de arquivos e sua conectividade
* Verificando assim possíveis corrupções em arquivos
* Comando de rotina, utilizado para ver se está tudo certo com nossos arquivos

Reflog

* O **git reflog** vai mapear todos os seus passos no repositório, até uma mudança de branch é inserida neste log
* Já o **git log**, que vimos anteriormente, apenas armazena os commits de um branch
* Os reflogs ficam salvos até expirar, o tempo de expiração padrão é de 30 dias
* **git reset –hard** <hash>

Transformando o repo para arquivo

* Com o comando git archive podemos transformar o repo em um arquivo compactado, por exemplo
* O comando é **git archive –format zip –output master\_files.zip master**
* E então a master vai estar zipada no arquivo master\_files.zip