# Exercícios GIT

**Pergunta 1**: Explica a finalidade dos 8 comandos git que aprendeste (init, status, log, add, commit, push, clone, pull)

**Pergunta 2**: Quem criou o sistema GIT e para que serve?

**Pergunta 3**: Explica por palavras tuas os principais conceitos da gestão de projetos de software com o GIT. (Working Directory, Staging, Repository, Remote Repo, Etc...)

**Pergunta 4**: Pesquisa na Internet 3 sistemas de armazenamento de código tipo GitHub e indica o seu endereço online (URL).

**Pergunta 5**: Dá a tua opinião pessoal sobre a utilidade desta tecnologia e como pensas vir a usá-la.

RESPOSTAS

1:

* **Git init**: cria um novo repositório em um diretório já existente
* **Git status**: mostra o estado atual do repositório, incluindo modificações, adicionados ou removidos
* **Git log**: exibe o histórico de commits feitos
* **Git add**: adiciona os arquivos a área de staging, basicamente o preparando para o commit
* **Git commit**: cria um commit com as alterações feitas na área de staging
* **Git push**: envia as alterações locais em um repositório remoto
* **Git clone**: é criado uma copia de um repositório remoto em seu próprio diretório local
* **Git pull**: busca as alterações feitas de um repositório remoto e incorpora-as ou as copia ao seu repositório local

2: O sistema GIT foi criado por Linus Torvalds em 2005, basicamente sendo um sistema de controle de versão distribuído, ou seja, permite que varias pessoas em simultâneo trabalhem em um projeto ao mesmo tempo

3: Conceitos importantes da gestão de projetos de software com o GIT:

* **Repositório**: aonde é armazenado todos os arquivos do projeto e o histórico de alterações; pode ser local ou remoto
* **Commits**: registo de uma ou mais alterações feitas no arquivo do projeto; cada commit possui uma mensagem para descrever o estado atual do projeto
* **Staging**: aonde os ficheiros “aguardam” para serem lançados ou commitados, basicamente um estado intermediário do ficheiro
* **Working directory**: local onde se é editado o projeto os arquivos do projeto, diretamente de sua máquina local
* **Remote Repo**: através de outros meios como GItHub ou GItLab é possível usar uma versão online de seu repositório; outras pessoas também podem colaborar

4:

* **GitHub**: sendo seu foco projetos open source e colaborações publicas, o github pertencente a Microsoft tem pontos altos em uma comunidade enorme, interfaces simples e é muito usado em entrevistas de emprego e portfolios.

URL: <https://github.com/>

* **GItLab**: focado em integração total do ciclo **DevOps** (processo iterativo e automatizado que otimiza a entrega de software), possui sistemas de issues e planejamento de projetos; também pode ser auto-hospedado (você instala em seu próprio servidor).

URL: <https://about.gitlab.com/>

* **BitBucket**: Possui integração com ferramentas de **Atlassian** (como **jira** e **trello**; sendo jira uma ferramenta de monitoramento de tarefas e trello uma app de gerenciamento de projetos)

URL: <https://bitbucket.org/>

5: Desde a primeira vez que ouvi falar, não entendia muito sobre o assunto, mas quando comecei a me aprofundar e a entender melhor eu percebi o quanto esta tecnologia merecia ser mais usada e conhecida, o conceito de poder trabalhar em um projeto e compartilhar com outros colaboradores em meios remotos ou locais é algo revolucionário, sem contar que é extremamente útil para fazer trabalhos comuns escolares, com a ótima facilidade de exportar e salvar em qualquer sistema de armazenamento de código como o próprio **GitHub**, (que recentemente, comecei a usar bastante para projetos) e acredito que vou usar ainda mais e usufruir totalmente da tecnologia para futuros trabalhos e projetos.