**Atividade sobre Git/Github – Projeto Macaibo**

1. **Controle de versão é um sistema (tem um banco de dados simples que mantêm todas as alterações nos arquivos sob controle de revisão) que registra alterações em um arquivo ou conjunto de arquivos ao longo do tempo.**
2. **Git (desenvolvido por Linus Torvalds, o criador do sistema operacional Linux) é um sistema de controle de versão distribuído open-source, ou seja, gratuito. Ele é utilizado para a criação de um histórico de alterações em código-fonte de projetos de desenvolvimento de software. Já o GitHub é um repositório remoto, criado como um serviço online de hospedagem de repositórios do Git.**
3. **Um repositório Git é um armazenamento digital que permite salvar versões do código de um projeto e acessar essas versões quando necessário. Ele é composto por todos os arquivos e pastas associados a um determinado projeto e podem ser públicos, internos ou particulares, como também podem ter vários colaboradores.**
4. **Esse comando cria um novo subdiretório chamado .git que contém todos os arquivos necessários de seu repositório – um esqueleto de repositório Git. Neste ponto, nada no projeto é monitorado ainda.**
5. **Quando esse comando é executado o Git recebe uma cópia completa de praticamente todos os dados que o servidor possui. Cada versão de cada arquivo no histórico do projeto é obtida.**
6. **A principal diferença entre os comandos git fetch e git pull é que o primeiro apenas busca e faz o download das alterações do repositório remoto (faz o download das informações para o repositório local), enquanto o segundo, além disto, mescla as alterações com o branch local (branch com o qual um programador, por exemplo, deseje mesclar as informações da versão escolhida do repositório remoto).**
7. **Um branch no Git é uma ramificação, ou seja, uma linha de desenvolvimento independente que funciona como um ponteiro (uma variável que armazena o endereço de memória de um objeto, permitindo que o programador acesse e modifique o valor dessa variável diretamente na memória) para um commit específico no repositório.**
8. **É um comando do Git que permite integrar alterações de uma ramificação (branch) para outra, ou seja, mesclar as alterações de um determinado commit com outro commit.**
9. **É uma ferramenta do Git que permite verificar o estado dos arquivos no repositório local, mostrando informações sobre os arquivos que foram adicionados, modificados ou excluídos, sendo útil para que os desenvolvedores acompanhem as alterações feitas no código e tomem as medidas necessárias para versionar e compartilhar essas alterações.**
10. **É usado para transferir alterações locais de um repositório Git para um repositório remoto. Literalmente, faz um upload das informações do repositório local para o repositório remoto.**
11. **O GitHub Actions é uma plataforma de integração e entrega contínua (CI/CD de classe mundial) que permite automatizar processos de desenvolvimento de software: compilação, testes, pipeline de implantação, criação de fluxos de trabalho, implantação de pull requests.**
12. **Para criar uma branch no Git, utiliza-se o comando git branch <branchname>, onde <branchname> é o nome da branch a ser criada. Para alternar entre branches no Git, use o comando git checkout <branchname> .**
13. **Mostra alterações entre a árvore de trabalho e o índice ou uma árvore, alterações entre o índice e uma árvore, alterações entre duas árvores, alterações resultantes de uma mesclagem, alterações entre dois objetos Blob ou alterações entre dois arquivos no disco.**