Respostas

1. Git é um software de controle de versões desenhado por Linus Torvalds, o criador do Linux. O propósito do Git é levar um registro de mudanças e coordenar o trabalho de várias pessoas em um repositório compartilhado.
2. Para criar um novo repositório, você vai usar o comando git init . git init é um comando único que você usa durante a configuração inicial de um novo repositório. A execução desse comando cria um novo subdiretório . git no diretório de trabalho atual.
3. git add significa Adicionar, ou seja, você pode adicionar um arquivo qualquer. Exemplo: git add nomedomeuarquivoaqui. Já em git commit é muito importante, pois indica as alterações que você fez no seu projeto. Ou seja, o commit é a realização de um conjunto de mudanças, alterações que você realizou em seu projeto.
4. O comando git push permite que você envie (ou em tradução literal, empurre) os commits de sua branch e repositório Git local para o seu repositório remoto. Para poder fazer um git push para seu repositório remoto, você deve garantir que todas as suas alterações no repositório local sejam confirmadas.
5. O comando git branch permite criar, listar, renomear e excluir ramificações. Um branch no Git é simplesmente um ponteiro móvel para um desses commits. O nome do branch padrão no Git é master . Conforme você começa a fazer commits, você recebe um branch master que aponta para o último commit que você fez. Cada vez que você faz um novo commit, ele avança automaticamente.
6. Use o comando git status para checar o estado atual do repositório.
7. Você só pode resolver conflitos de merge no GitHub causados por alterações concorrentes na linha, como quando as pessoas fazem alterações diferentes na mesma linha do mesmo arquivo em diferentes branches no seu repositório Git. Para todos os outros tipos de conflito de merge, você deve resolver o conflito localmente na linha de comando.
8. git é um software VCS local que permite aos desenvolvedores salvar snapshots de seus projetos ao longo do tempo. Geralmente é melhor para uso individual. GitHub é uma plataforma baseada na web que incorpora os recursos de controle de versões do git para que possam ser usados colaborativamente. Também inclui recursos de gerenciamento de projetos e equipes, assim como oportunidades para networking e codificação social.
9. Projetos públicos: qualquer um pode ver, forkar e baixar o código fonte. Projetos privados: apenas os usuários que você liberar terão acesso ao código.
10. Clonar um repositório

No GitHub.com, navegue até a página principal do repositório.
Acima da lista de arquivos, clique em Código

Copie a URL do repositório.
Para clonar o repositório usando HTTPS, em "HTTPS", clique em .
Para clonar o repositório usando uma chave SSH, incluindo um certificado emitido pela autoridade de certificação SSH da sua organização, clique em SSH e em .
Para clonar um repositório usando a GitHub CLI, clique em GitHub CLI e em .

Abra Git Bash.
Altere o diretório de trabalho atual para o local em que deseja ter o diretório clonado.
Digite git clone e cole a URL já copiada.
git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
Pressione ENTER para criar seu clone local.
$ git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
> Cloning into `Spoon-Knife`...
> remote: Counting objects: 10, done.
> remote: Compressing objects: 100% (8/8), done.
> remove: Total 10 (delta 1), reused 10 (delta 1)
> Unpacking objects: 100% (10/10), done.
Clonar um repositório vazio
Um repositório vazio não contém arquivos. Muitas vezes, isso é feito se você não inicializar o repositório com um README ao criá-lo.
No GitHub.com, navegue até a página principal do repositório.
Para clonar seu repositório usando a linha de comando via HTTPS, em "Configuração rápida", clique em . Para clonar o repositório usando uma chave SSH, incluindo um certificado emitido pela autoridade de certificação SSH da sua organização, clique em SSH e em .

Como alternativa, para clonar seu repositório no Desktop, clique em Configurar no Desktop e siga os avisos para concluir o clone.

Abra Git Bash.
Altere o diretório de trabalho atual para o local em que deseja ter o diretório clonado.
Digite git clone e cole a URL já copiada.
git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
Pressione ENTER para criar seu clone local.
$ git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
> Cloning into `Spoon-Knife`...
> remote: Counting objects: 10, done.
> remote: Compressing objects: 100% (8/8), done.

> remove: Total 10 (delta 1), reused 10 (delta 1)
> Unpacking objects: 100% (10/10), done.

11. Uma pull request é uma proposta para mesclar as alterações de um branch em outro. Em uma pull request, os colaboradores podem revisar e discutir o conjunto de alterações proposto antes de integrá-las à base de código principal. As pull requests exibem as diferenças, ou comparações, entre o conteúdo no branch de origem e aquele no branch de destino.
12. Cada linha em um arquivo .gitignore especifica um padrão de pesquisa do arquivo em relação ao caminho do arquivo .gitignore. A sintaxe .gitignore é flexível e dá suporte ao uso de curingas para especificar arquivos individuais ou múltiplos por nome, extensão e caminho. O Git corresponde aos padrões de pesquisa .gitignore com os arquivos em seu projeto para determinar quais arquivos ignorar.
    Normalmente, você apenas adicionará um arquivo .gitignore à pasta raiz do projeto. No entanto, você pode adicionar um arquivo .gitignore a qualquer pasta de projeto para informar ao Git quais arquivos ignorar dentro dessa pasta e suas subpastas em qualquer profundidade aninhada. Para vários arquivos .gitignore, os padrões de pesquisa do arquivo especificados por um arquivo .gitignore dentro de uma pasta têm precedência sobre os padrões especificados por um arquivo .gitignore dentro de uma pasta pai.
13. Um fork é um novo repositório que compartilha configurações de código e visibilidade com o repositório "upstream" original. Os forks geralmente são usados para iterar ideias ou alterações antes de serem propostas de volta para o repositório upstream, como em projetos código aberto ou quando um usuário não tem acesso de gravação ao repositório upstream.
14. Por padrão, sem argumentos, `git log` lista os commits feitos neste repositório em ordem cronológica inversa; isto é, o commit mais recente aparece primeiro.
15. O comando git pull é usado para buscar e baixar conteúdo de repositórios remotos e fazer a atualização imediata ao repositório local para que os conteúdos sejam iguais. Fazer o merge de alterações upstream remotas no repositório local é algo comum em fluxos de trabalho de colaboração baseados em Git. O comando git pull é uma combinação de outros dois comandos, git fetch seguido por git merge. No primeiro estágio da operação, o git pull executa o comando git fetch, que abrange a ramificação local para qual a HEAD aponta. Quando o conteúdo é baixado, o git pull insere o fluxo de trabalho de merge. O commit de merge é criado e a HEAD é atualizada para apontar o novo commit.