GitHub Quick Guide

Quick guide on how to create a repository on GitHub and its basic functions:

Creating a Repository on GitHub

1. Sign in to GitHub:
- Go to [GitHub](*https://github.com/)* and sign in to your account.

2. Navigate to Your Profile:
- Click on your profile icon in the top-right corner of the screen.

3. Select "Repositories" Tab:
- From the dropdown menu, select "Your repositories".

4. Click on "New":
- On the repositories page, click on the green "New" button.

5. Fill in Repository Details:
- Enter a name for your repository.
- Optionally, add a description.
- Choose visibility (public or private).
- Initialize with a README file if needed.
- Add a license and/or .gitignore file if necessary.

6. Click on "Create Repository":
- Once you've filled in the details, click on the green "Create repository" button.

Basic Functions of a GitHub Repository

1. Cloning a Repository
- **Clone with HTTPS**: Allows you to clone the repository using HTTPS URL.

```
git clone https://github.com/username/repository.git
```

- Clone with SSH: Allows you to clone the repository using SSH URL.

```
git clone git@github.com:username/repository.git
```

2. Adding and Committing Changes
-Add Files: Add files to the staging area.

```
git add <file>
```

- Commit Changes: Commit staged changes to the repository.

```
git commit -m "Commit message"
```

3. Pushing Changes

- **Push to Remote:** Push committed changes to the remote repository.

```
git push origin master
```

4. Pulling Changes

- **Pull from Remote:** Fetch and merge changes from the remote repository to your local repository.

```
git pull origin master
```

5. Branching

-Create a Branch:**Create a new branch for development or bug fixing.**

```
git branch <branch_name>
```

- **Switch Branches**: Switch to a different branch.

```
git checkout <branch_name>
```

- **Merge Branches:** Merge changes from one branch to another.

```
git merge <branch_name>
```

6. Forking and Pull Requests

- **Fork Repository:** Create a copy of someone else's repository.
    - Click on the "Fork" button on the repository's page.

- **Pull Request:** Propose changes to the original repository.
    - Click on the "New pull request" button.
    - Compare changes and create a pull request.

Brief description of each function mentioned in the guide:

1. Cloning a Repository:

- **Clone with HTTPS:** Creates a local copy of a repository using the HTTPS URL.
- **Clone with SSH:** Creates a local copy of a repository using the SSH URL. SSH is typically used for authenticated access and is more secure.

2. Adding and Committing Changes:

- **Add Files:** Stages changes for commit by adding files to the staging area.
- **Commit Changes:** Records changes to the repository, accompanied by a commit message describing the changes made.

3. Pushing Changes:

- **Push to Remote:** Uploads committed changes from your local repository to the remote repository on GitHub, allowing others to see your changes.

4. Pulling Changes:

- **Pull from Remote:** Fetches changes from the remote repository and merges them into your local repository, ensuring your local copy is up to date with the remote.

5. Branching:

- **Create a Branch:** Creates a separate line of development, allowing you to work on new features or bug fixes without affecting the main codebase.
- **Switch Branches:** Moves your working directory to a different branch within the repository.
- **Merge Branches:** Combines changes from one branch into another, integrating the work done in separate branches.

6. Forking and Pull Requests:

- **Fork Repository:** Creates a personal copy of someone else's repository, allowing you to freely experiment with changes without affecting the original repository.
- **Pull Request:** Proposes changes made in your forked repository to be integrated into the original repository. Pull requests are used for collaboration and code review.