Question:
Create a file by adding your details as name, fathers name etc push file to remote repository and add your area of specialisation in remote repository. write the steps then perform to move file from
a) remote repo to local repo
b) display the list of git objects in current directory
c)retrieve the blob object for the current version of file
d)move the file from local repo to staging area

Solution:

- Navigated to OneDrive:
- Moved to Desktop Created a New Directory
- Entered the New Directory
- Created a New File
- Added Content to the File: wrote my details
- details ("Name:Ashish Kumar, Father's name: Sk singh, Course :Btech Cse">test.txt) into test.txt.

a) Move from Remote Repo to Local Repo

initialised a new Git repository in your local folder.

1. This command creates a local copy of the entire repository from the remote location.

getting the files from a remote Git repository (like GitHub) to your computer. Doing this using git init

```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test
$ git init
Initialized empty Git repository in C:/Users/singh/OneDrive/Desktop/test/.git/
```

a) Displaying the list of git objects in the current directory

opened the hidden `.git` folder in my project, which stores all the information about your files. Inside the `git` and `objects` folder, you listed what's there. This helps you see how Git manages the files behind the scenes.

- **HEAD**: A pointer to the current branch or commit in a Git repository.
- **config**: A file that contains configuration settings for the Git repository.
- **description**: A brief text description of the repository, primarily used by GitWeb.
- **hooks**/: A directory containing scripts that run on specific Git events (e.g., pre-commit, post-commit).
- **info**/: A directory holding metadata about the repository, including exclude patterns.
- **objects**/: A directory storing all the Git objects (blobs, trees, commits) identified by their SHA-1 hashes.
- **refs**/: A directory containing references to commits, including branches and tags.

```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ ls -a
./  ../  .git/  test.txt

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ cd .git

singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git (GIT_DIR!)
$ ls
HEAD  config  description  hooks/  info/  objects/  refs/
```

```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git (GIT_DIR!)
$ cd objects

singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git/objects (GIT_DIR!)
$ ls
68/  info/  pack/
```

a) Retrieving the blob object of current version of file

Each version of a file in Git is stored as a blob (a type of object). I retrieved the blob to see the content of my `test.txt` file. This shows you the exact text inside your file and confirms it is a blob.

- **68**/: A directory containing specific objects identified by their hash, representing compressed pack files or loose object files.
- **info**/: A subdirectory within `objects/` that holds metadata, such as the `packs` file listing all packs in the repository.
- **pack**/: A directory where Git stores pack files, which are compressed collections of Git objects to optimize storage and access.

```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git/objects (GIT_DIR!)
$ cd 68

singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git/objects/68 (GIT_DIR!)
$ ls
740b2dd437fe7f5a3555e6e2033a6fe690c508
```

```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git/objects/68 (GIT_DIR!)
$ git cat-file -p 68740b2dd437fe7f5a3555e6e2033a6fe690c508
Name:Ashish Kumar, Father's name: Sk singh, Course :Btech Cse

singh@Ashish MINGW64 ~/OneDrive/Desktop/test/.git/objects/68 (GIT_DIR!)
$ git cat-file -t 68740b2dd437fe7f5a3555e6e2033a6fe690c508
blob
```

a) Moving the file from local repo to staging area

Git which files to include in the next snapshot (commit). I do this by using `git add`, which moves your file to the staging area.
And then committed the file to the repo and
Created a repo on github
Added the remote to the local file
And pushed by checking the branch

1. Git Add:
- Use `git add <file>` to include changes to a specific file in the next commit.
- Use `git add .` to stage all modified files in the current directory.

2. Staging Area:
- Staging area (or index) is where you prepare changes before committing them.
- It allows you to review and select which changes to include in the next snapshot.

3. Commit Changes:
- Use `git commit -m "Commit message"` to save your staged changes to the repository.
- The commit creates a snapshot of your current changes and includes a message describing the change.

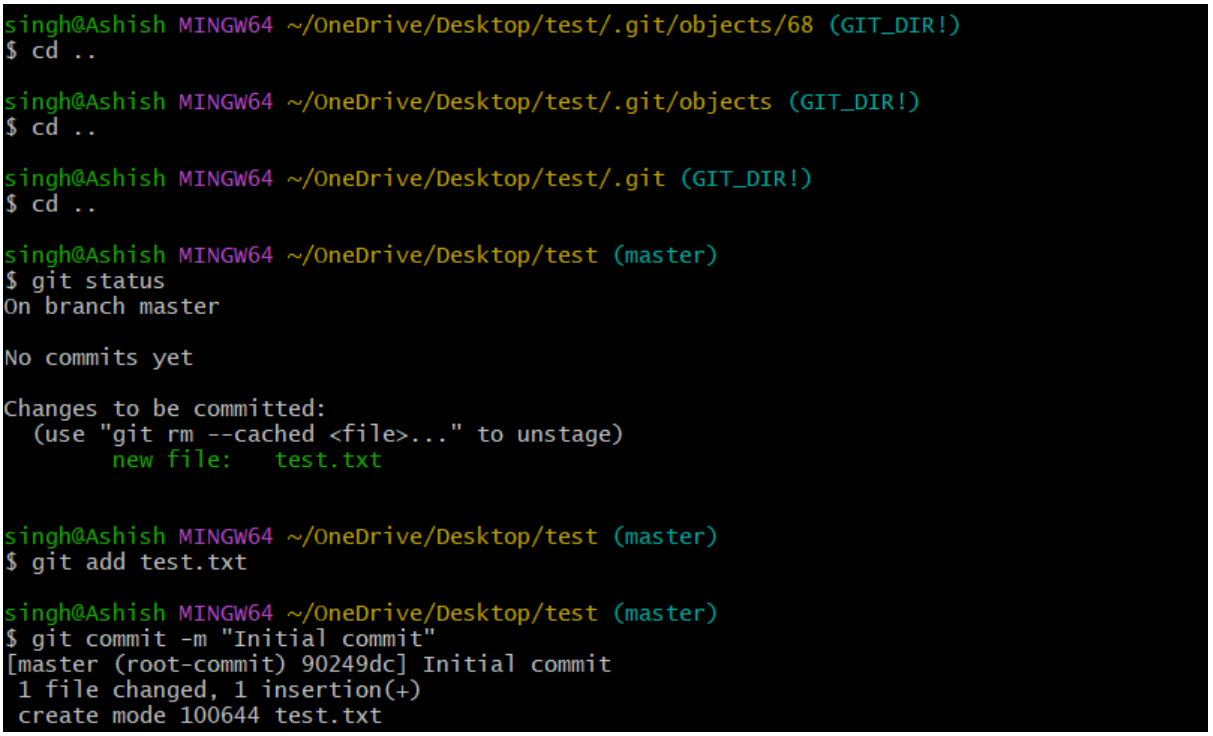
4. Create a Repository on GitHub:
- Log in to GitHub, create a new repository (repo) through the GitHub interface.
- This repository will host your project and its version history.

5. Add Remote to Local Repository:

- Use `git remote add origin <repository-url>` to link your local repository to the GitHub repo.
- This command sets the remote origin to the GitHub repository URL, enabling future pushes and pulls.

6. Check Branch:
- Use `git branch` to view your current branch, ensuring you are on the correct branch to push changes.
- It's essential to be on the correct branch (e.g., main, master, or a feature branch) before pushing.

7. Push Changes:
- Use `git push origin <branch-name>` to upload your local commits to the remote GitHub repository.
- This command syncs your local branch with the corresponding branch on GitHub, making your changes available to others.



```
singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git status
On branch master
nothing to commit, working tree clean

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ ^C

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git remote -v

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git remote add origin https://github.com/codingbyash/devopstestca2.git

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git remote -v
origin  https://github.com/codingbyash/devopstestca2.git (fetch)
origin  https://github.com/codingbyash/devopstestca2.git (push)

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git branch
* master

singh@Ashish MINGW64 ~/OneDrive/Desktop/test (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 275 bytes | 137.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/codingbyash/devopstestca2.git
 * [new branch]      master -> master
```

Result

Github link:

https://github.com/codingbyash/devopstestca2