

Solution 1: clone, delete and init

1. **Create the directory of the main project.** *If you use linux:*

```
mkdir <name_project>
```

2. **Move inside that directory.** *If you use linux:*

```
cd <name_project>
```

3. **Clone a project inside the directory.** Here you can use even a command line or git GUI like github desktop if you are in Windows and clone:

```
git  
clone https://github.com/MeetDineshSharma/<repository_name>.git
```

4. This step is important. You need to **delete the .git directory inside the project <repository_name>**, this is necessary if you want to create a monolith project with lot of git repository. First move inside the directory and then delete the .git. *If you use linux:*

```
cd <repository_name>  
rm -rf .git
```

The **-rf** is necessary because there are some protected files inside it and many folder.

5. **Repeat from point 3** until you put every project you want.
6. Return in the root folder and **use the init command of git** to create the repository. *In a command line or with the git GUI run the init command:*

```
git init
```

This command will create the .git folder with all the infos to use git inside that folder.

7. **Add the remote origin** with the link to the github repository:

```
git remote add  
origin https://github.com/MeetDineshSharma/<main_project_name>.git
```

This assumes that the main project repository already exists in github.

8. Done... Now you can use this as a git repository.

Solution 2: elegant submodules

This is the most elegant solution (maybe).

1. **Create the directory of the main project.** *If you use linux:*

```
mkdir <name_project>
```

2. **Move inside that directory.** *If you use linux:*

```
cd <name_project>
```

3. **Initialize the git repository** using the init command with a console or a git GUI:

```
git init
```

4. Instead of clone the repository you want, you can **create multiple submodules**. You can find the man page here: [Git-Tools-Submodules](#) . Some git GUI doesn't have this functionality, if not use the command line and run this command:

```
git submodule  
add https://github.com/MeetDineshSharma/<repository_name>
```

Run this command for every repository you want to add.

5. **To use a module** you need "clone" it. First of all you need to init:

```
git submodule init
```

Then update:

```
git submodule update
```

6. **Add the remote origin** with the link to the github repository:

```
git remote add  
origin https://github.com/MeetDineshSharma/<main_project_name>.git
```

This assumes that the main project repository already exists in github.

8. **Done...** Now you can use this as a git repository.

WARNING: Is being a long time since I didn't use the submodules so is probably that the related solution may work and may not 😊