```
project (repo)
|-.git
   |-objects
   |-hooks
   |-logs
   |-refs
   |-HEAD
   |-config
```

How does the git stores the sourcecode inside the repository? How does it maintains multiple versions of the sourcecode along with commit logs/history?
GIT is an distributed repository version control system, where the entire repository is distributed across all the developers machines who are working on the project.

GIT stores the
1. SourceCode
2. Commit Logs (author info, Commit Message, #Files Committed)
3. Versions of the Files
4. Commit History (Which File has been modified for how many times, and what changes are made)

inside the repository within .git directory itself. So usually it looks like the size of the repository seems to be huge, since it carries all the information pertaining described above.

But GIT stores this information in an optimized way so that the size of the repository is very small and can be quickly distributed across all the developers in the Team.

How is GIT storing this information inside the repository?
GIT uses hashing technic in storing or organizing the sourcecode and commit log/history inside the repository. Along with that to store the source code it uses Blob files
1. Hashing
2. Blob files

What is Hashing means?
Hashing is an mathematical technic using which an unique number will be generated based on the contents/data we passed.
hashing function: h(data) => unique string sequence

This function works based on the below rules:
1. no #2 different data will have the same hash sequence

hf(“Humpty”) = ax938
hf(“Dumpty”) = ax939

2. always for the same given input, the Hash that is computed will be same
hf(“Humpty”) = ax938
hf(“Humpty”) = ax938

3. using the hash value we cannot compute the original data
hf(“Humpty”) = ax938
by using ax938 = we cannot compute the original data “Humpty”

hf(“ax938”) = will results in a different hash treating it as an input data

What are Blob files?
Blob stands for Binary Large object files. When we commit a file in the GIT repository, the GIT will not store the original file inside the repository.
for eg..

vegetables.txt (newly)
carrot
greens
potato

```
git add .
git commit -m “vegetables”
```

when we commit the vegetables.txt, the GIT will not store this file inside the repository directly. GIT takes the contents of the file vegetables.txt, converts the contents of the file into bytes/binary format and encrypts the contents and stores inside the repository.
This way of storing the contents of a file in binary format is called “Blob” file.