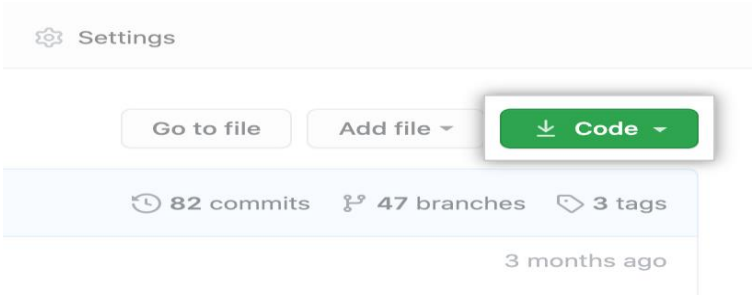
ASSIGNMENT 02

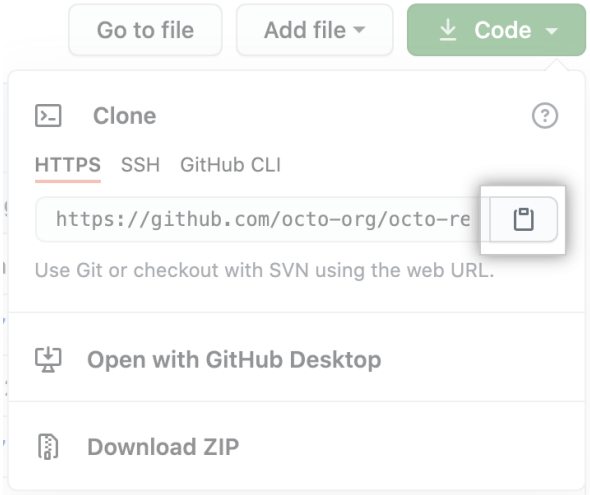
1.Using Github

a: How to clone repository?

- On GitHub.com, navigate to the main page of the repository.
- Above the list of files, click Code.



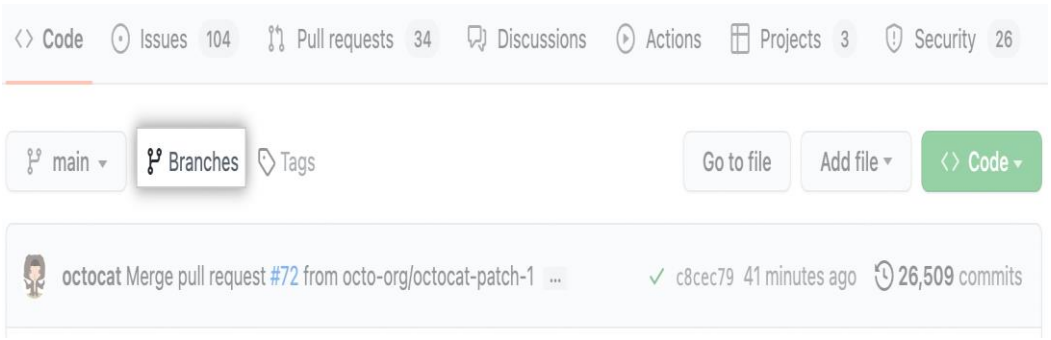
- Copy the URL for the repository.
- To clone the repository using HTTPS, under "HTTPS", click .
- To clone the repository using an SSH key, including a certificate issued by your organization's SSH certificate authority, click SSH, then click .
- To clone a repository using GitHub CLI, click GitHub CLI, then click .



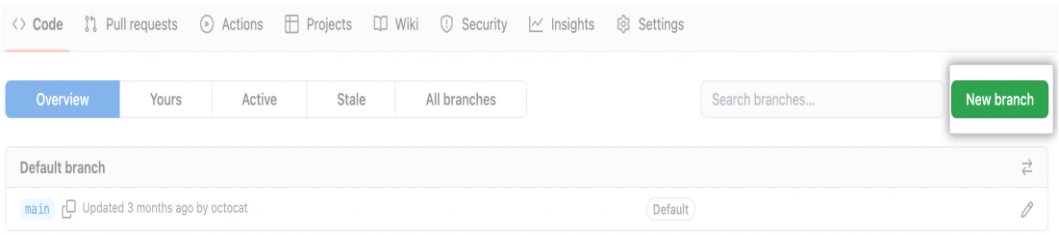
- Open Git Bash.
- Change the current working directory to the location where you want the cloned directory.
- Type git clone, and then paste the URL you copied earlier.
  $ git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY
- Press Enter to create your local clone.

b) How to create branch in github?

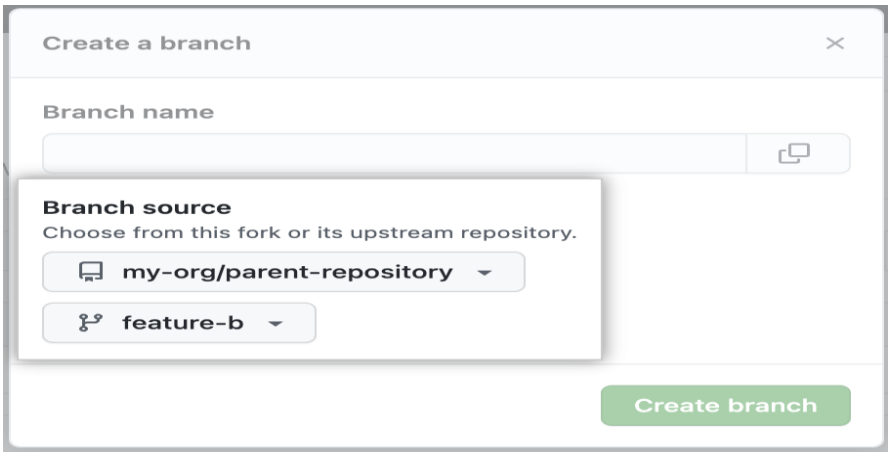
- On GitHub.com, navigate to the main page of the repository.
- Above the list of files, click Branches.



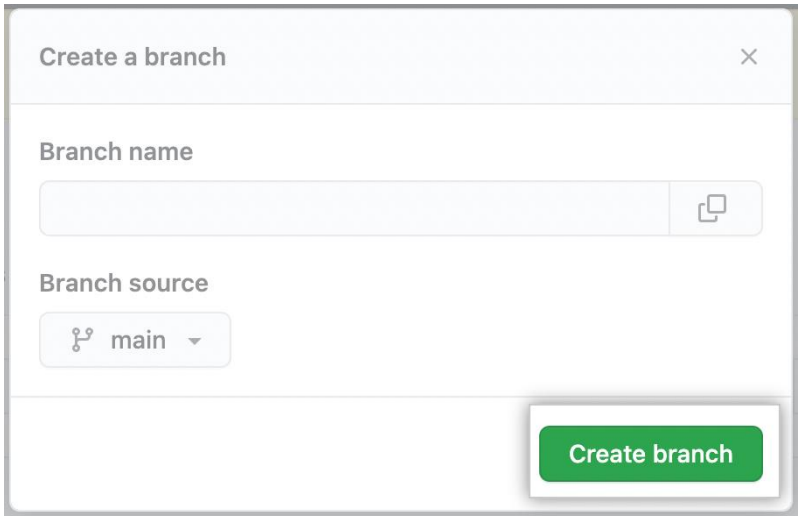
- Click New branch.



- In the dialog box, enter the branch name and optionally change the branch source.
  If the repository is a fork, you also have the option to select the upstream repository as the branch source.



- Click Create branch



c) How to make a commit?

- `git commit` creates a commit, which is like a snapshot of your repository. These commits are snapshots of your entire repository at specific times.
  Commits are the building blocks of "save points" within Git's version control.

  git commit -m "update the README.md with link to contributing guide"

d) How to push changes to the remote repository?

- The git push command takes two arguments:
- A remote name, for example, origin
- A branch name, for example, main
- For example:
  git push REMOTE-NAME BRANCH-NAME
- To rename a branch, you'd use the same git push command, but you would add one more argument: the name of the new branch. For example:
  git push REMOTE-NAME LOCAL-BRANCH-NAME: REMOTE-BRANCH-NAME
- Pushing tags
  By default, and without additional parameters, git push sends all matching branches that have the same names as remote branches.
  To push a single tag, you can issue the same command as pushing a branch:
  git push REMOTE-NAME TAG-NAME
  To push all your tags, you can type the command:
  git push REMOTE-NAME --tags
- Remotes
  When you clone a repository you own, you provide it with a remote URL that tells Git where to fetch and push updates.
  git remote add upstream THEIR_REMOTE_URL

2.What is ommers in blockchain?

It's possible for two blocks to be created simultaneously by a network. When this happens, one block will be left out. This leftover block is called an ommer block.

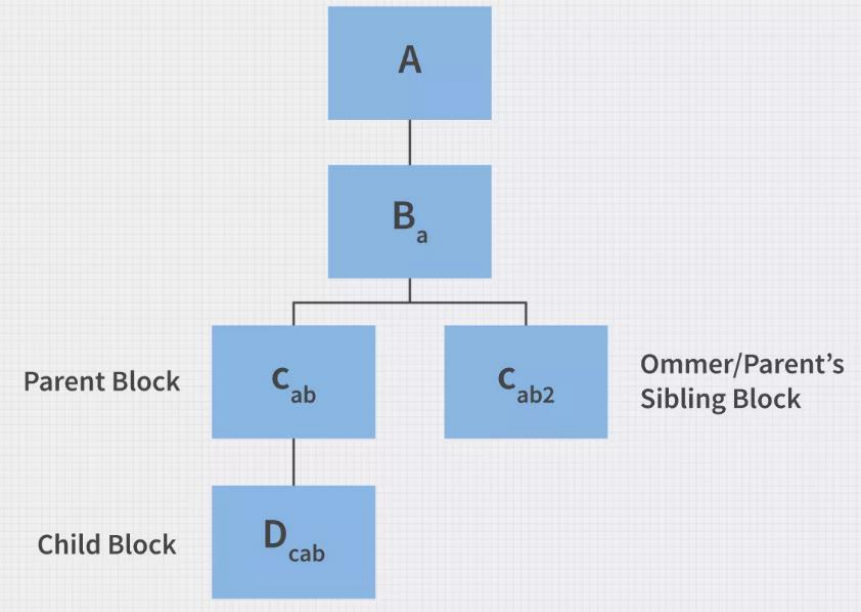
ommer blocks are created in the Ethereum blockchain when two blocks are created and submitted to the ledger at roughly the same time. Only one can enter the ledger.

the first block in a tree could be named block A. The next block created from block A would be considered block A's child and would include A's information plus its own

This block could be called block B but could be represented as Ba. B is the name of the new block, and "a" refers to the data from the parent block. This parent/child relationship continues as more blocks are added with the information from each previous block. This creates a family tree and blockchain.

Now consider if two blocks were validated and created simultaneously from Ba. They are blocks Cab and Cab2, sibling blocks from the same parent block. Only one can be added to the blockchain—so the network chooses

Cab. Cab2 is a fork of the original blockchain but is not added to it or validated. Finally, another block is mined on the blockchain that kept Cab. This is block Dcab. Cab2 is the sibling of Dcab's parent, so Cab2 is an ommer block.



3.What is hard fork and soft fork in blockchain?

Hard fork:

Forks are modifications to the blockchain's network algorithm that leads to the primary blockchain network to split. If there is a scenario where an old blockchain network has crypto working on it, a fork on that blockchain will lead to the formation of a parallel token on the newly forked blockchain network.

The regulations of the blockchain rules are upgraded or altered in a hard fork, creating the previous blockchain and the updated blockchain incompatible with each other.

This implies that the previous nodes will deny the recently upgraded blocks, and the newer blockchain will run under fresh guidelines that will continuously deny blocks from the old blockchain indefinitely.

Soft fork:

A hard fork is a backward-incompatible upgrade to the blockchain, whereas a soft fork is a rule modification that is forward-compatible. The old blockchain will keep accepting blocks from the new advanced blockchain platform since the fork is a forward-compatible alteration, although the regulations have been modified due to the new upgrade.

Broadly said, a soft fork convinces the old blockchain network to accept the altered rules, thus allowing both the upgraded and old blocks of transactions to be accepted at the same moment.

4.What is dificulty bomb?

The term difficulty describes how difficult the computations needed to mine a block in a blockchain for a particular cryptocurrency are. Mining a cryptocurrency is often confused with creating a coin; however, mining is the verification process that involves solving the 64-character hash that encrypts the transaction information. When a miner's machine solves the hash, they are rewarded with a coin.

The original Ethereum blockchain came with an intrinsic feature that increased the difficulty of mining over time—the more blocks that were mined, the more difficult and time-consuming it became to mine the next block.

Ethereum's developers created the difficulty bomb to increase the difficulty of solving the hash exponentially more than before, eventually making it too expensive in time and energy to be worth the cost

5.What is Re-entrancy attack?

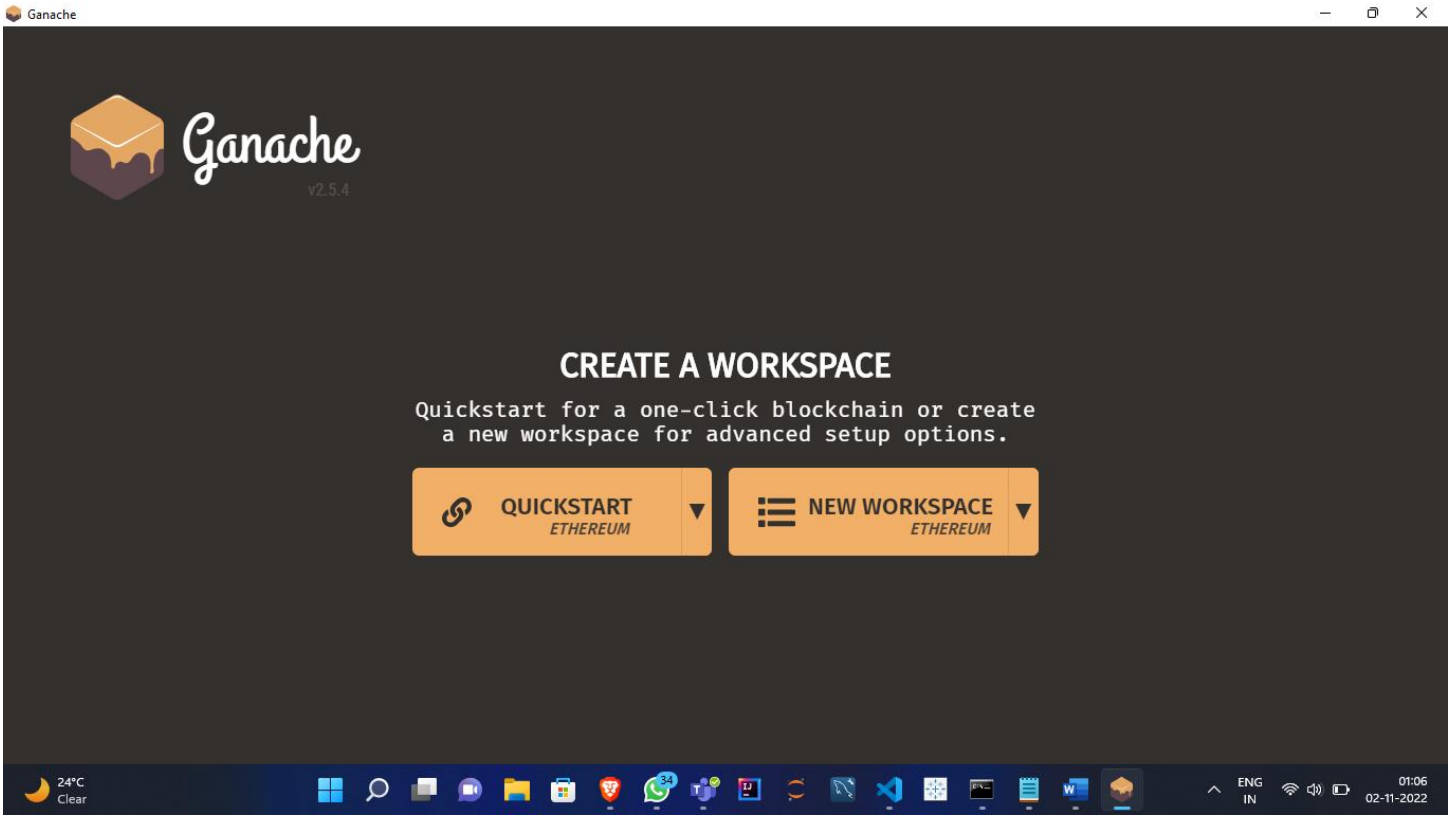
The Reentrancy attack is one of the most destructive attacks in the Solidity smart contract. A reentrancy attack occurs when a function makes an external call to another untrusted contract. Then the untrusted contract makes a recursive call back to the original function in an attempt to drain funds.

When the contract fails to update its state before sending funds, the attacker can continuously call the withdraw function to drain the contract’s funds. A famous real-world Reentrancy attack is the DAO attack which caused a loss of 60 million US dollars.

6. Difference between ERC721 and ERC20?

- The main distinction between ERC20 and ERC721 tokens is that the former is a fungible token, but the latter is a non -fungible token.
- ERC20 tokens are interchangeable and represent a single entity, whereas ERC721 tokens represent a collection of assets. Furthermore, ERC721 is not divisible.
- CrytoKitties is a notable example of ERC721 tokens; gaining complete ownership of virtual cats is one-of-a-kind and cannot be shared with any other player. The game is swiftly gaining traction to the point where blockchain platform gaming may become more generally embraced in the future. However, the same is not possible with ERC20 due to their fungible characteristics.
- ERC20 tokens can be divided in any number of ways. Even sharing 0.1 % of your token is possible. On the other hand, ERC721 tokens are not-divisible.

7. Install and run Ganache and attach the screenshot in the pdf.



8.Write a short note on your understanding of hd wallet.

- A hierarchical deterministic wallet is a digital wallet commonly used to store the keys for holders of cryptocurrencies such as Bitcoin and Ethereum.
- To prevent hacking these keys must be randomly generated and backed up in the wallet.
- HD wallets enable a series of key pairs to be created from one random seed, providing convenience and manageability as well as high-level security.

9. Characteristics of IPFS?

IPFS (Interplanetary file system) is a distributed system for storing and accessing files, websites, applications, and data.

1) Decentralized Protocol

IPFS is a decentralized system that uses content-addressed storage. In this protocol, each piece of data is assigned a unique content identifier (CID). All content-addressed data in IPFS can be found and retrieved based on this unique CID.

IPFS content is housed in several locations in a shared, peer-to-peer network using a distributed hash table (DHT). This decentralized protocol means there is no one point of failure in a single server. And no one entity can censor or eliminate the data.

2) Enhanced Security

This creates huge security advantages over HTTP. The immutable nature of resources in IPFS greatly reduces many cybersecurity threats.

3) High Performance

Storing and distributing data with IPFS saves bandwidth by retrieving data from multiple peers at once. A user requests content based on its unique CID, IPFS retrieves that data based on the CID from multiple nodes at once, and it is then delivered to the user in the quickest, most efficient way possible.

4) Enables Easier Deduplication and Archiving

The CID created with IPFS provides a digital fingerprint that can ensure authenticity and uniqueness. This makes deduplication simpler by creating a single instance of data with an immutable CID. And because there is only one copy of each resource, authentication is also much simpler.

Since there’s no duplication, IPFS minimizes the storage space consumed by data backups and archives. This creates a huge advantage for any organization archiving their data.

5) Allows Improved Content Control

IPFS provides much more control to content creators. Creators can distribute their work themselves without being dependent on a content distributor entity.

Creators also don’t have to spend money on servers to control their content. With IPFS, anyone can make their content available in the network, and anyone in the world can receive that content securely.

10.What is a rainbow table attack?

A rainbow table attack is a password cracking method that uses a special table (a “rainbow table”) to crack the password hashes in a database. Applications don’t store passwords in plaintext, but instead encrypt passwords using hashes. After the user enters their password to login, it is converted to hashes, and the result is compared with the stored hashes on the server to look for a match. If they match, the user is authenticated and able to login to the application.

11.What is payble function in solidity?

When writing a smart contract, you need to ensure that money is being sent to the contract and out of the contract as well. Payable does this for you, any function in Solidity with the modifier Payable ensures that the function can send and receive Ether. It can process transactions with non-zero Ether values and rejects any transactions with a zero Ether value. Additionally, if you want a function to process transactions and have not included the payable keyword in them the transaction will be automatically rejected. An example of this is supposing you have a receive() function with the payable modifier, this means that this function can receive money in the contract and now imagine there is a function send() without a payable modifier it will reject the transaction when you want to send money out of the contract.

You can define a payable function using the following syntax:

function receive() payable {}

function send() payable {}

12.How library contract is different with other contract in solidity.

Libraries are similar to Contracts but are mainly intended for reuse. A Library contains functions which other contracts can call. Solidity have certain restrictions on use of a Library. Following are the key characteristics of a Solidity Library.

Library functions can be called directly if they do not modify the state. That means pure or view functions only can be called from outside the library.

Library can not be destroyed as it is assumed to be stateless.Library cannot have state variables. Library cannot inherit any element. Library cannot be inherited.

13. Create a mist wallet and add the address to the pdf & which will be pushed to github.

14.What is Opensea with respect to NFT?

Opensea is the first ever decentralized NFT marketplace built on the Ethereum blockchain and is currently the largest. You can use it to buy or sell NFTs and create your own NFT collections. As complicated as it sounds, OpenSea is actually a simple platform to navigate, and anyone can use it.

OpenSea uses the ERC-721 and ERC-1155 Ethereum standards for NFTs to confirm ownership of digital collectibles so that users don't claim ownership of what does not belong to them.

While the marketplace facilitates the transfer of NFTs, the transactions are done directly on the Ethereum network between a seller and a buyer.

Because the transaction fees on Ethereum can be outrageous, OpenSea has recently introduced the Polygon blockchain to facilitate faster and cheaper transactions.

15.What is RLP encoding in Ethereum?

Recursive Length Prefix (RLP) serialization is used extensively in Ethereum's execution clients. RLP standardizes the transfer of data between nodes in a space-efficient format. The purpose of RLP is to encode arbitrarily nested arrays of binary data, and RLP is the primary encoding method used to serialize objects in Ethereum's execution layer. The only purpose of RLP is to encode structure; encoding specific data types (e.g. strings, floats) is left up to higher-order protocols; but positive RLP integers must be represented in big-endian binary form with no leading zeroes (thus making the integer value zero equivalent to the empty byte array). Deserialized positive integers with leading zeroes get treated as invalid. The integer representation of string length must also be encoded this way, as well as integers in the payload.

The RLP encoding function takes in an item. An item is defined as follows：

a string (i.e. byte array) is an item , a list of items is an item

data structures like ["cat", ["puppy", "cow"], "horse", [[]], "pig", [""], "sheep"].

16. Read about solidity assembly from solidity.

Assembly or Assembler language indicates a low-level programming language that can be converted to machine code by using assembler. Assembly language is tied to either physical or a virtual machine as their implementation is an instruction set, and these instructions tell the CPU to do that fundamental task like adding two numbers.

Solidity has an option to write an assembly language code inside the smart contract's source code. With the help of Solidity assembly, we can directly interact with the EVM using the opcodes. Assembly provides more control over some logic which cannot be possible using only solidity, like pointing to the specific memory block. One of the main advantages is that it reduces the cost of the gas used to deploy the contract. Solidity has two ways to implement the assembly language:

Inline Assembly: Inline assembly code can be written inside solidity code for more fine-grain control and especially used for enhancing the language via creating new libraries. Inline assembly can be inserted in between solidity statements in a way that EVM can understand. It can also be used when the optimizer is not able to produce efficient code. Solidity becomes easier when features like assembly local variables, functions calls, switch statements, if statements, loops, etc are used.

Syntax:

```
assembly{
    // assembly language statements
}
```

17. Characteristic of decentralised storage network.

Accessible: An ideal distributed system should be accessible. Participation in the network should be easy, allowing as many nodes as possible to store and distribute files on behalf of the network.

Trustless: A trustless system enables cooperation between two parties without them having to know one another or look to a third party. Rather, the incentives of the system push actors towards the behavior necessary for the network to function.

Verifiable: An ideal storage system should make it easy to continuously prove that nodes are storing the exact data they have promised. This type of auditability is key in achieving **Trustlessness.** If you can always establish that data is being stored correctly, you have less need to trust the party providing the storage.

Open: Finally, an ideal distributed storage system is open: its code is open-source and auditable. Furthermore, the storage system should not be monolithic. Instead, it should expose an open protocol that anybody can implement and build upon, rather than encouraging lock-in.

18.What is turing complete?

Turing complete refers to the idea that given infinite time, any program in one language could be written (albeit perhaps inefficiently) in another. In Ethereum, Turing complete means using conditional statements and loops to program smart contracts.