A constructor is a function in Solidity that contains initialization logic for state variables. It is invoked only when the smart contract is deployed, and cannot be invoked again.

If not defined explicitly, the compiler creates a default constructor. To declare a constructor, use the constructor keyword followed by the access specifier, which can be public or internal. A contract with an internal constructor is considered abstract and cannot be deployed.

When inheriting a contract, the child contract must provide the parent contract's constructor parameters. If the child contract doesn't, it's marked as abstract and not deployed. If no explicit constructor is defined, the default constructor will be called.

Key Points:

1. A contract has a single constructor.
2. Executed once to initialize its state.
3. The final code, including public functions and code accessible through public functions, is deployed to the blockchain.
4. Constructors can be public or internal, and if no constructor is defined, a default constructor is present.

YouTube Link:

https://www.youtube.com/embed/dYfuCI7Vao8

Solidity Source File

.....

Bibliography/References

https://cryptomarketpool.com/constructor-in-solidity-smart-contracts/