**Constructor:**

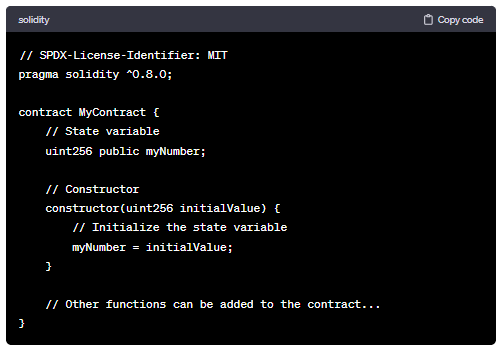
At the time when the smart contract is executed -> constructor is the 1st one that is executed.

In 1 smart contract, we can use only 1 constructor.

Work of constructor to initialize state variable or to set the owner for the contract.

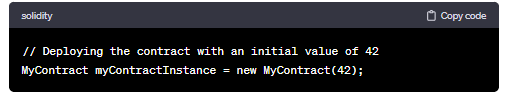
In Solidity, a constructor is a special function that is executed only once when a contract is deployed to the Ethereum blockchain. It is used to initialize the state variables and perform any necessary setup for the contract. The constructor has the same name as the contract and does not have a return type.

Here's a simple example of a Solidity contract with a constructor:



In this example, the **MyContract** contract has a state variable **myNumber** and a constructor that takes an argument **initialValue**. The constructor initializes the **myNumber** state variable with the provided initial value when the contract is deployed.

When deploying this contract, you would provide an argument for the constructor, like this:



The constructor is executed only once during the deployment of the contract, and its purpose is to set up the initial state of the contract. After deployment, the state variables can be modified through other functions defined in the contract.

**Example:**



