* Smart Contracts are immutable.
* Smart Contract runs deterministically in the context of an **Ethereum Virtual Machine** as part of the Ethereum network protocol.
* Smart contracts are typically written in a high-level language, such as Solidity. But in order to run, they must be compiled to the low-level bytecode that runs in the EVM.
* Each contract is identified by an Ethereum address.
* It is also worth noting that smart contracts are not executed "**in parallel**" in any sense—the **Ethereum** world computer can be considered to be a **single-threaded machine**.
* A failed transaction is still recorded as having been attempted
* To delete a contract, you execute an EVM opcode called SELFDESTRUCT (previously called SUICIDE).
* Deleting a contract in this way does not remove the transaction history (past) of the contract, since the blockchain itself is immutable .
* It is also important to note that the SELFDESTRUCT capability will only be available if the contract author programmed the smart contract to have that functionality
* In smart contracts, bugs literally cost money.
* the most widely used language for smart contracts (Solidity) is imperative. Programmers, like most humans, resist change.
* Dr Gideon Greenspan is the founder and CEO of Coin Sciences, the company behind the MultiChain platform for private blockchains.
* Smart contracts are for blockchain use cases which can’t be implemented with transaction constraints.
* Whatever the answer turns out to be, the key to remember is that smart contracts are simply one method for restricting the transactions performed in a database.