Ethereum Accounts

1. **Externally Owned Account (EOA)**
   - Controlled by a private key and identified by an unique address;
   - It has no associated code and someone can generate and send a transaction if he/she posses the private key;
   - Used sending/receiving ETH and for interacting with smart contracts (deployment, calling functions etc);
2. **Contract account**
   - Controlled by the contract code
   - It’s an autonomous agent and it’s code execution is triggered by receiving a transaction or a message (call) from another contract of EOA. Has control of its own Ether balance and state variables;
   - Can hold an Ether balance like an EOA. There must be a payable function defined;
   - Doesn’t have a public or a private key;

Ethereum Accounts

An Ethereum Account has:

1. Nonce - a sequential number tied to every transaction that represents the number of transactions the sender account has made on the network. It's a mechanism used to ensure that the same transaction isn't submitted twice.

2. Ether balance in wei

3. Account address

4. Account private & public key (just for EOA)

5. Code (just for the contract account). This is the immutable EVM bytecode

6. Storage (empty by default, just for the contract account)

Account state contains: nonce, balance, storage root and code hash (Keccak-256 hash of the EVM account code).

Ethereum Address

- **An EOA address is derived from the last 20 bytes (160 bits) of the public key** that are Keccak-256 hashed. It's represented in a hexadecimal format, which is often indicated explicitly by appending 0x to the address;
- The address for an Ethereum Contract is deterministically computed from the address of its creator (sender) and how many transactions the creator has sent (nonce).
- There is a **lower-case** address version and **partial upper-case** version that also contains a **checksum**.