# Smart Contracts can be written in

1. Java, C++, Solidity and JavaScript, because the Ethereum Blockchain is completely language agnostic and cross compilers for every major language exist.
2. **Solidity, Viper, LLL and Serpent, because those are high level languages that are compiled down to bytecode**
3. Solidity and JavaScript, because those are the official first implementations for Distributed applications and the Blockchain fully supports those languages

# Solidity gets compiled to

1. Bytecode that can’t be understood by humans
2. **Bytecodes which are essentially opcodes running instruction by instruction**

# Gas is used

1. **Depending on the instruction/opcode run by the Ethereum Blockchain**
2. Is a fixed amount for the length of your smart contract

# To store almost all data in the Ethereum Blockchain

1. A Linked List with pointers to previous blocks hashes is used
2. **A Merkle Patricia Trie is used**
3. A Radix Trie is used because the Merkle Patricia Trie is too inefficient

# You interact with a smart contract and see a gas usage of 50,000 gas with a gas cost of 15Gwei, how much Ether would you have to pay to the miner?

1. **750,000,000,000,000 Wei**
2. 750,000,000,000 Wei
3. 750,000,000 Wei
4. A flat fee of 1 Ether

# Checking the balance of an address inside a loop of a smart contract constantly

1. Doesn’t cost any gas
2. **Cost gas every time we check the balance**