Smart Contract Compilation

- The Solidity source code is passed to the solidity compiler and the compile returns the EVM bytecode that is deployed and the contract ABI - Abstract Binary Interface.
- There are many solidity compilers available: Remix built-in compiler, solc, solcjs
- Contract bytecode is public. It is saved on the Blockchain and can't be encrypted because it must be run by every Ethereum node;
- Opcodes are the human readable instructions of the program. They can be easily obtained from bytecode;
- Contract source code doesn't have to be public. Most contracts are public to build trust.

Smart Contract Compilation

- Anyone that wants to interact with the contract must have access to the contract ABI. ABI is basically **how you call functions** in a contract and get data back;
- ABI is list of contract's function and arguments and it's in JSON format. ABI is known at compile time.
- ABI is generated from source code through compilation. **If we don't have the source code we can't generate the contract ABI** (or only from the bytecode using reverse engineering);