VYPER

What is Vyper?

Vyper is an open-source programming language that can be used to create smart contracts (self executing programme in blockchain determined by a condition) on the Ethereum blockchain.

Vyper was designed to be as similar to Python as possible while focusing on security and simplicity.

It appears to be logically similar to Solidity and syntactically similar to Python.

After Solidity, Vyper is the second most popular choice for developing your contract.

developed by:
Team led by Taylor Monahan
software developer,
Co-founder of MyEtherWallet

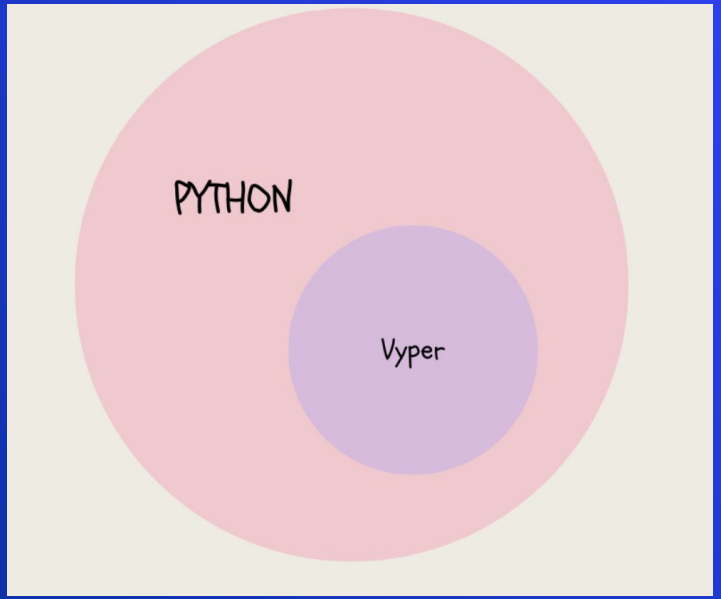
(Monahan started the project in 2016 and the Vyper team is currently actively developing and maintaining the language.)

Advantages:

The main goal was to make contracts more auditable and secure, resulting in less error-prone contracts.It is a strongly typed language, ie that you cannot use one data type as another.

- *It is easier for Python developers to get started with it.*
- *SECURITY: Vyper was created to help smart contracts have a more natural feel. It is a new high-level language for Ethereum smart contract development designed to address security flaws.*
- *SIMPLICITY: The language and compiler have a simple and straightforward user interface.*

- *AUDITABILITY: Vyper code makes an effort to be human-readable.It is aimed at the code reader rather than the programmer.This is one of the differences between it and Python, which is a superset of the language.*



Vyper is simple to set up and use.Nevertheless,Vyper does not aim to be a perfect replacement for everything that can be done in Solidity; it will forbid or make things more difficult if it deems it necessary to increase security.

Which Operating System supports Vyper?

As such, it is not tied to any specific operating system and can be used on any system that supports the necessary development tools and environments. This includes popular operating systems like Windows, MacOS, and Linux.

To use Vyper, you will need to have a development environment set up on your computer, which typically includes a code editor, a command-line interface, and the necessary dependencies and libraries. You will also need to have a local blockchain testnet, such as Geth or OpenEthereum, to test and deploy your contracts.

Has vyper influenced or been influenced by any other languages?

Vyper is a relatively new programming language and it's not clear whether it has directly influenced the development of any other programming languages. However, Vyper's design and development has been influenced by other programming languages, such as Python, C and JavaScript.

Vyper is designed to be a more limited, simpler language than Solidity, the primary language used for writing smart contracts on the Ethereum blockchain. It's possible that Vyper's design principles and approach to smart contract development could influence other programming languages in the future, particularly those used for blockchain and smart contract development.

Real Time Applications:

Vyper is primarily used for writing smart contracts on the Ethereum blockchain ecosystem to create secure, transparent, and tamper-proof smart contracts.
Vyper can be used to create a variety of decentralized applications, such as

- ***TOKEN CONTRACTS**: These contracts are used to create, manage, and distribute digital assets, also known as tokens, that can be traded on various cryptocurrency exchanges.*
- ***CROWDFUNDING PLATFORMS**: Vyper can be used to write smart contracts that govern the rules and mechanics of the crowdfunding campaign, such as how funds are collected, how investors are rewarded, and how the project team can access the funds.*
- ***VOTING SYSTEM**:In a Vyper-based voting system, voters would need to use an Ethereum wallet to interact with the smart contract and cast their vote. The smart contract would then record the vote on the blockchain, providing a tamper-proof and transparent way to track the results of the election.It can enforce rules such as voter identity verification, vote counting, and handling of multiple votes from the same user, as well as the rules of vote counting, such as the majority rule, proportional representation, and so on.*

List of Companies that use or have used vyper:

- ***Gnosis**: A prediction market platform, uses Vyper for their smart contract development.*
- ***Aragon**: A decentralized governance platform, uses Vyper for their smart contracts.*
- ***MolochDAO**:A decentralized autonomous organization (DAO) focused on funding Ethereum infrastructure development, uses Vyper for their smart contracts.*
- ***Kyber Network**: A decentralized exchange platform, has used Vyper for their smart contracts in the past.*

Beginner Level Smart Contract Code Using Vyper

```
contract HelloWorld {

    string public message;

    constructor() public {
        message = "Hello, World!";
    }

    function getMessage() public view returns (string) {
        return message;
    }
}
```

This simple smart contract allows to set a message by calling the setMessage function and passing the message as an argument. The message can be retrieved by calling the getMessage function.
The public keyword in the message variable definition makes it accessible from outside the smart contract while view keyword in the getMessage function makes it read-only function.