

Variables & Functions Visibility

There are **four types of visibilities** for functions and state variables:

1. Public

- The function is part of the contract interface and **can be called both internally** (from within the same contract) **and externally** (from other contracts or by EOA accounts);
- This is the **default for functions**;
- **A getter is automatically created for public variables.** They can be easily accessed from dApps;

2. Private

- Private functions and variables are **available only in the contract they are defined in** (not in other contracts: derived or underived);
- This is the **default for state variables**, they can be accessed only by a getter function;

3. Internal

- Functions are accessible **only from the contract they are defined in and from derived contracts**;
- This is the **default for state variables**, they can be accessed only in the current contract;

4. External

- The function is part of the contract interface, can be **accessed only from other contracts** or by EOA accounts using transactions. It's automatically public;
- **Not-available for state variables**;

Variables & Functions Visibility (cont)

- Everything that is inside a contract **is visible to all external observers**. Making something private only prevents other contracts from accessing and modifying the information, but it will still be visible to the whole world outside of the blockchain;
- Information is **not encrypted by default** on the Ethereum Blockchain;
- The visibility specifier is given after the type for state variables and between parameter list and return parameter list for functions;
- External functions **are more efficient** than public functions in terms of gas consumption;