Lesson 2 - Syntax

Detailed contract structure

- (Review) Contract structure
- What are interfaces
- What are contracts, indeed
- Multiple objects per file
- Libraries

References

https://docs.soliditylang.org/en/latest/grammar.html

https://docs.soliditylang.org/en/latest/structure-of-a-contract.html

Code Reference

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >=0.7.0 <0.9.0; interface helloworldinterface { function
helloworld() external view returns (string memory); settext(string memory
newtext) external; } contract helloworld is string private text;
constructor() text="Hello World" ; public override memory) return }< code>
```

Function definition

- Replacing memory with calldata when stack is enough
- Relation between identifier and parameters and MethodID
- Fallback and receive functions
- Definitions
    - Visibility
    - State mutability
    - Modifiers
    - Virtual
    - Override

References

https://docs.soliditylang.org/en/latest/introduction-to-smart-contracts.html#storage-memory-and-the-stack

https://solidity-by-example.org/function-selector/

Code references

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >=0.7.0 <0.9.0; contract helloworld { string private text;
constructor() text="Hello World" ; } function helloworld() public view
returns (string memory) return settext(string calldata newtext) }< code>
```

Variable declaration and definition

- Elementary types
  - Booleans
  - Integers
  - Fixed
  - Address
  - Bytes
  - Strings
- State Variables
- Constants
- Data locations (again)
- Arrays
- Mappings

References

https://docs.soliditylang.org/en/latest/types.html

Common Solidity Global Variables

- Reserved words and global variables that a programmer should know
- Global variables about blockchain state
- Global variables about the transaction
- Global variables about the transaction message

References

https://docs.soliditylang.org/en/latest/units-and-global-variables.html

Assertion and Modifiers

- How errors are handled on solidity (briefly)
- Assertion
- Require statements
- Modifiers
- Where to use modifiers

References

https://docs.soliditylang.org/en/latest/control-structures.html#error-handling-assert-require-revert-and-exceptions

https://docs.soliditylang.org/en/latest/structure-of-a-contract.html#function-modifiers

Restricting access to functions

- Wrapping up contents
  - Modifier
  - Assertion inside modifiers
  - Message Sender
  - Visibility
  - Mutability
- Implementing basic access control on *setText*

References

https://docs.soliditylang.org/en/latest/common-patterns.html#restricting-access

Homework

- Create Github Issues with your questions about this lesson
- Read the references
- Get to know the Solidity Cheatsheet in depth
- Get familiar with the Solidity Style Guide
- Prepare your environment for next class:
  - Node
  - NPM
  - Yarn
  - Git CLI
  - VS Code
- Create a free account on infura
- Create a free account on alchemy
- Create a free account on etherscan