Solidity Global Variables - block

`blockhash(uint blockNumber) returns (bytes32)`: hash of the given block - only works for 256 most recent, excluding current, blocks

`block.coinbase` (`address payable`): current block miner's address

`block.difficulty` (`uint`): current block difficulty

`block.gaslimit` (`uint`): current block gaslimit

`block.number` (`uint`): current block number

`block.timestamp` (`uint`): current block timestamp as seconds since unix epoch

`now` (`uint`): current block timestamp (alias for `block.timestamp`)

Solidity Global Variables - msg

`msg.data` (`bytes calldata`): complete calldata

`msg.sender` (`address payable`): sender of the message (current call)

`msg.sig` (`bytes4`): first four bytes of the calldata (i.e. function identifier)

`msg.value` (`uint`): number of wei sent with the message

Solidity Global Variables - tx

`tx.gasprice` (`uint`): gas price of the transaction

`tx.origin` (`address payable`): sender of the transaction (full call chain)

Solidity Global Variables - this

`this` **(current contract's type):**

the current contract, explicitly convertible to Address

Solidity Global Functions - assert()

`assert(bool condition)`:

causes an invalid opcode and thus state change reversion if the condition is not met - to be used for internal errors.

Solidity Global Functions - require()

`require(bool condition)`:

reverts if the condition is not met - to be used for errors in inputs or external components.

`require(bool condition, string memory message)`:

reverts if the condition is not met - to be used for errors in inputs or external components. Also provides an error message.

Solidity Global Functions - revert()

`revert()`:

abort execution and revert state changes

`revert(string memory reason)`:

abort execution and revert state changes, providing an explanatory string

Solidity Global Functions - selfdestruct()

`selfdestruct(address payable recipient)`:

destroy the current contract, sending its funds to the given Address