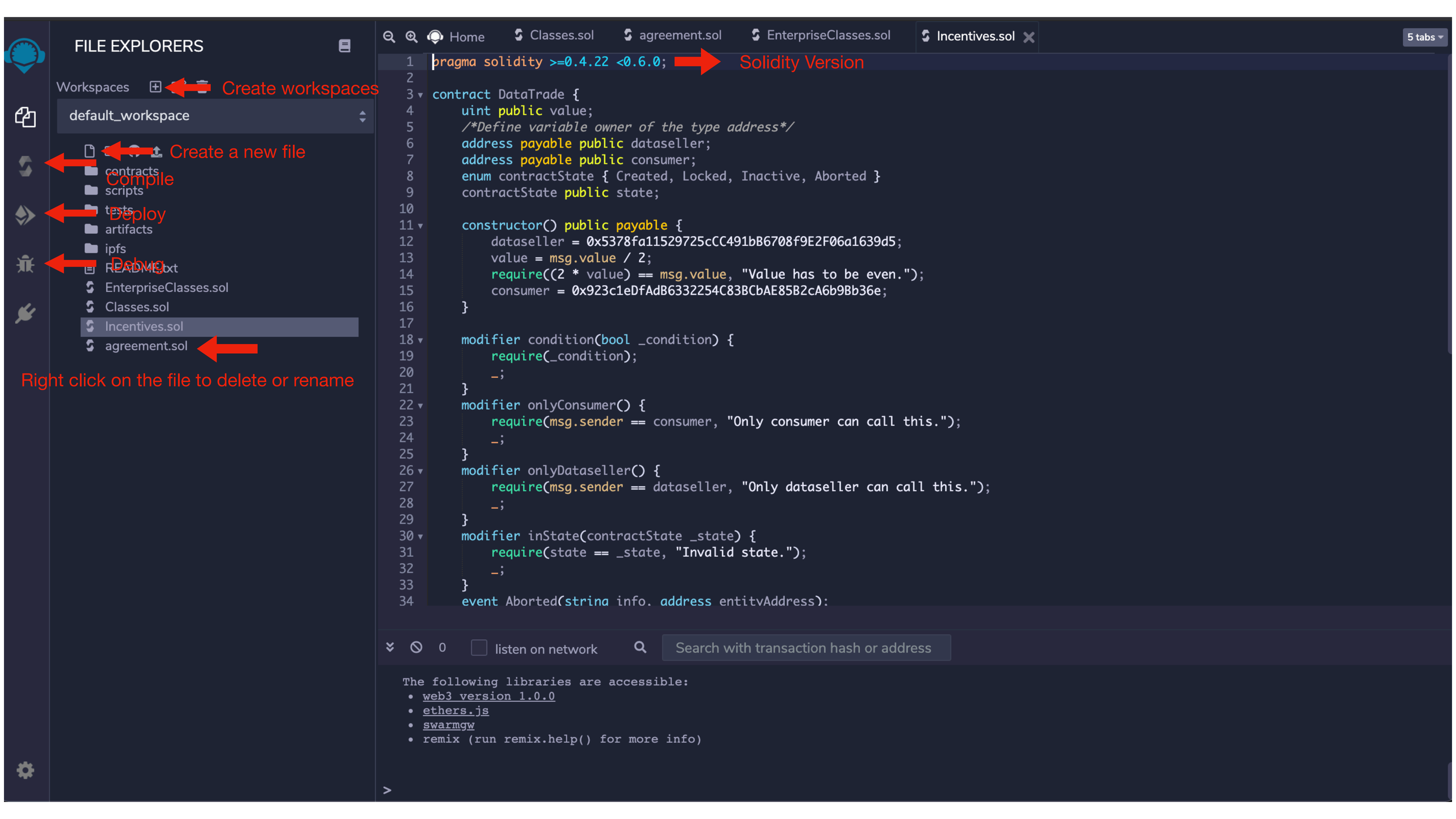
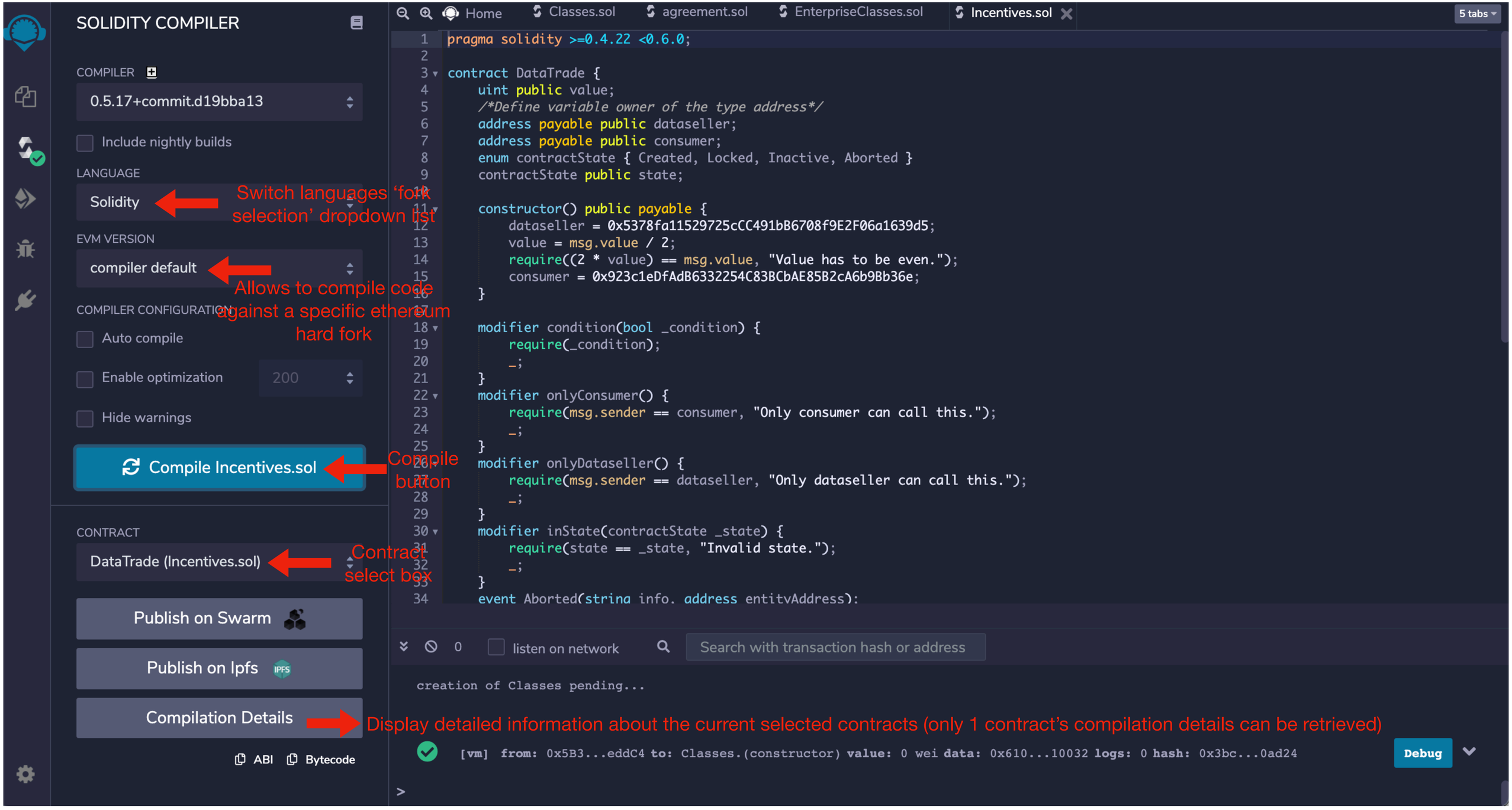
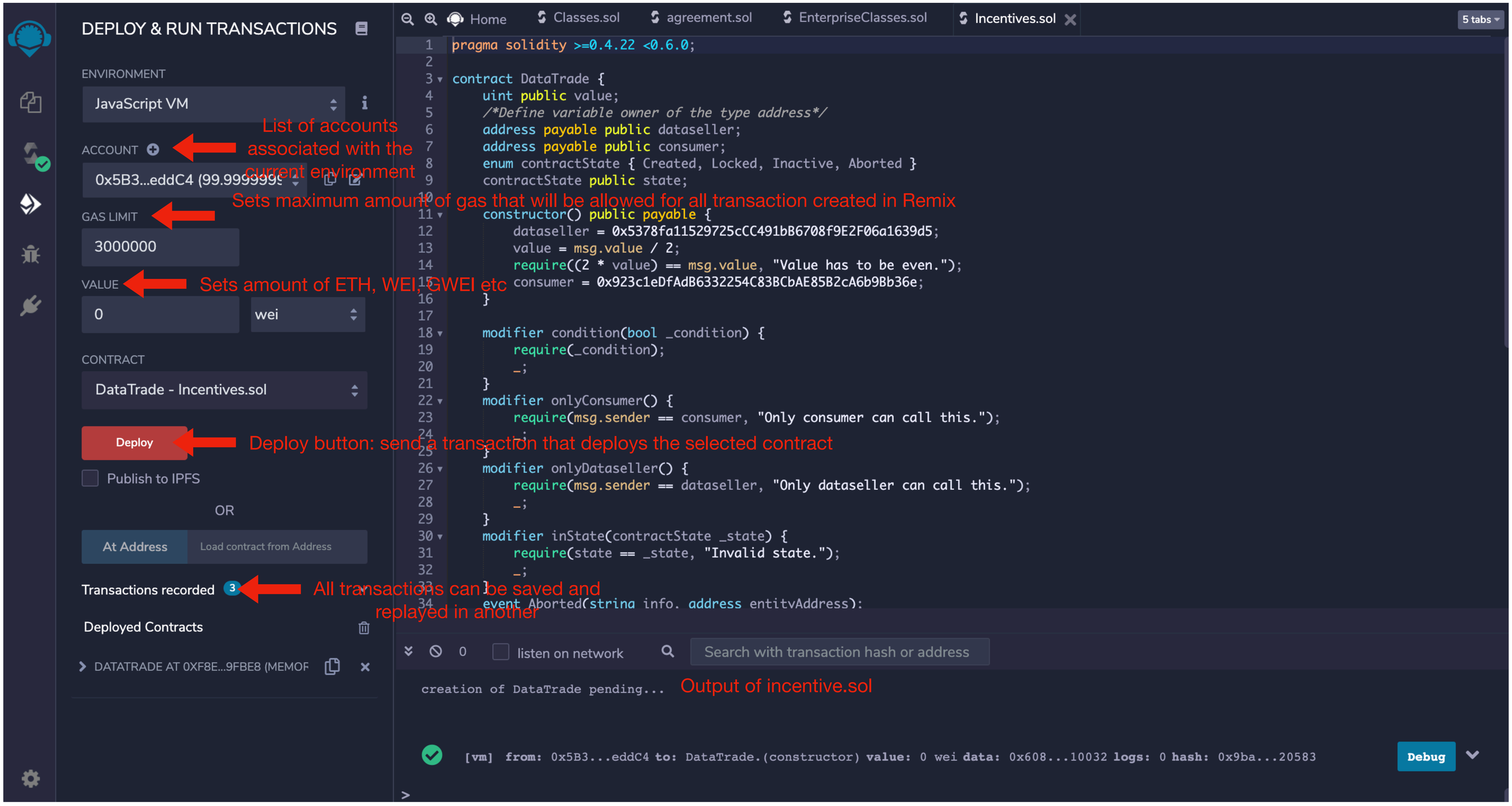
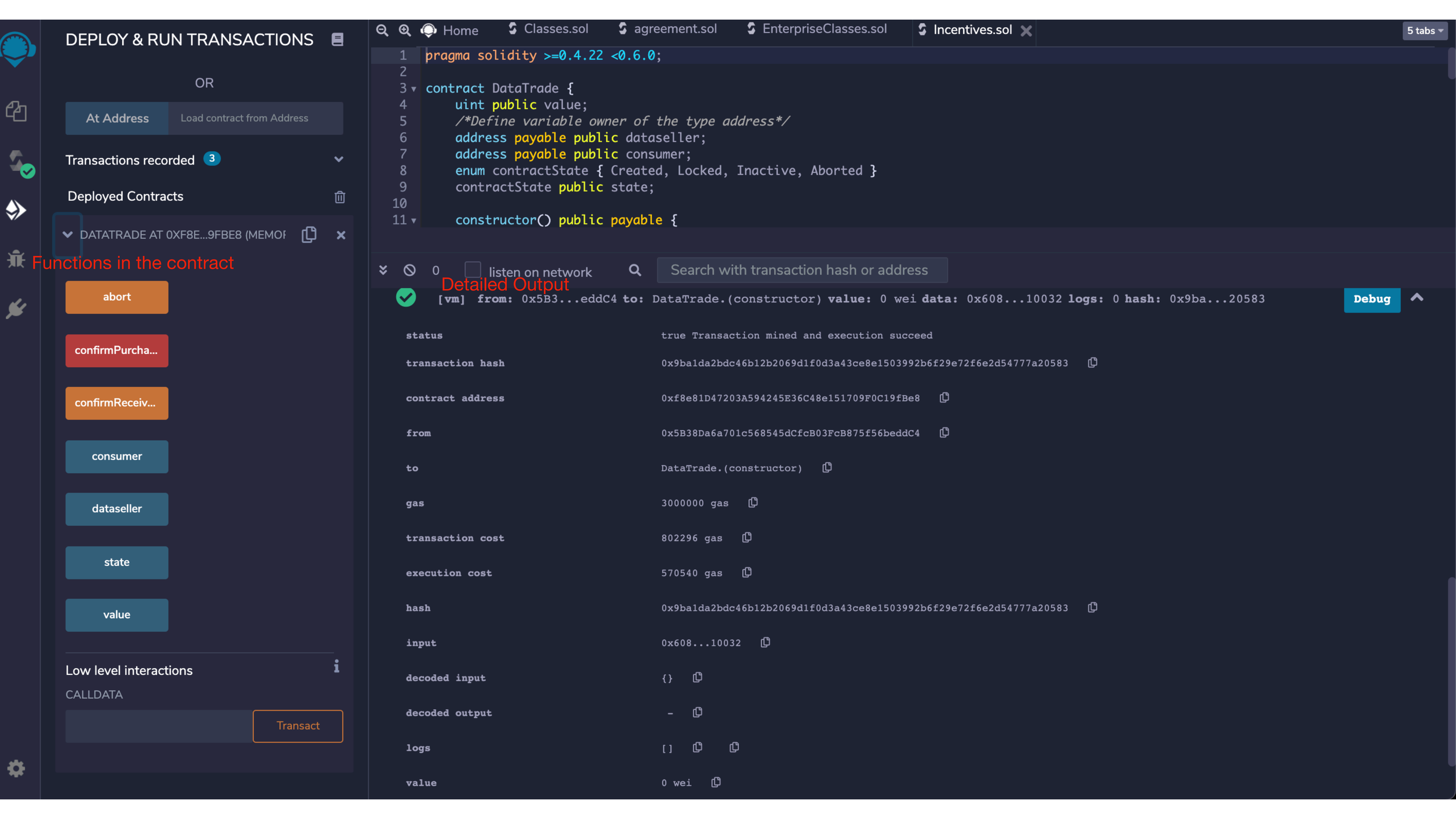
# Guide for Remix IDE (https://remix.ethereum.org/)


FILE EXPLORERS
Workspaces
Create workspaces
default_workspace
Create a new file
Compile
Deploy
Debug
contracts
scripts
tests
artifacts
ipfs
README.txt
EnterpriseClasses.sol
Classes.sol
Incentives.sol
agreement.sol
Right click on the file to delete or rename
Home
Classes.sol
agreement.sol
EnterpriseClasses.sol
Incentives.sol
5 tabs
Solidity Version
pragma solidity >=0.4.22 <0.6.0;
contract DataTrade {
uint public value;
/*Define variable owner of the type address*/
address payable public dataseller;
address payable public consumer;
enum contractState { Created, Locked, Inactive, Aborted }
contractState public state;
constructor() public payable {
dataseller = 0x5378fa11529725cCC491bB6708f9E2F06a1639d5;
value = msg.value / 2;
require((2 * value) == msg.value, "Value has to be even.");
consumer = 0x923c1eDfAdB6332254C83BCbAE85B2cA6b9Bb36e;
}
modifier condition(bool _condition) {
require(_condition);
_;
}
modifier onlyConsumer() {
require(msg.sender == consumer, "Only consumer can call this.");
_;
}
modifier onlyDataseller() {
require(msg.sender == dataseller, "Only dataseller can call this.");
_;
}
modifier inState(contractState _state) {
require(state == _state, "Invalid state.");
_;
}
event Aborted(string info, address entityAddress);
0
listen on network
Search with transaction hash or address
The following libraries are accessible:
web3 version 1.0.0
ethers.js
swarmgw
remix (run remix.help() for more info)
>


SOLIDITY COMPILER
COMPILER
0.5.17+commit.d19bba13
Include nightly builds
LANGUAGE
Solidity
Switch languages 'fork selection' dropdown list
EVM VERSION
compiler default
Allows to compile code against a specific ethereum hard fork
COMPILER CONFIGURATION
Auto compile
Enable optimization
200
Hide warnings
Compile Incentives.sol
Compile button
CONTRACT
DataTrade (Incentives.sol)
Contract select box
Publish on Swarm
Publish on Ipfs
Compilation Details
Display detailed information about the current selected contracts (only 1 contract's compilation details can be retrieved)
ABI
Bytecode
Home
Classes.sol
agreement.sol
EnterpriseClasses.sol
Incentives.sol
5 tabs
pragma solidity >=0.4.22 <0.6.0;
contract DataTrade {
uint public value;
/*Define variable owner of the type address*/
address payable public dataseller;
address payable public consumer;
enum contractState { Created, Locked, Inactive, Aborted }
contractState public state;
constructor() public payable {
dataseller = 0x5378fa11529725cCC491bB6708f9E2F06a1639d5;
value = msg.value / 2;
require((2 * value) == msg.value, "Value has to be even.");
consumer = 0x923c1eDfAdB6332254C83BCbAE85B2cA6b9Bb36e;
}
modifier condition(bool _condition) {
require(_condition);
_;
}
modifier onlyConsumer() {
require(msg.sender == consumer, "Only consumer can call this.");
_;
}
modifier onlyDataseller() {
require(msg.sender == dataseller, "Only dataseller can call this.");
_;
}
modifier inState(contractState _state) {
require(state == _state, "Invalid state.");
_;
}
event Aborted(string info, address entityAddress);
0
listen on network
Search with transaction hash or address
creation of Classes pending...
[vm] from: 0x5B3...eddC4 to: Classes.(constructor) value: 0 wei data: 0x610...10032 logs: 0 hash: 0x3bc...0ad24
Debug
>


DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
JavaScript VM
List of accounts associated with the current environment
ACCOUNT
0x5B3...eddC4 (99.9999999
Sets maximum amount of gas that will be allowed for all transaction created in Remix
GAS LIMIT
3000000
VALUE
Sets amount of ETH, WEI, GWEI etc
0
wei
CONTRACT
DataTrade - Incentives.sol
Deploy
Deploy button: send a transaction that deploys the selected contract
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 3
All transactions can be saved and replayed in another
Deployed Contracts
DATATRADE AT 0XF8E...9FBE8 (MEMOF
Home
Classes.sol
agreement.sol
EnterpriseClasses.sol
Incentives.sol
5 tabs
pragma solidity >=0.4.22 <0.6.0;
contract DataTrade {
uint public value;
/*Define variable owner of the type address*/
address payable public dataseller;
address payable public consumer;
enum contractState { Created, Locked, Inactive, Aborted }
contractState public state;
constructor() public payable {
dataseller = 0x5378fa11529725cCC491bB6708f9E2F06a1639d5;
value = msg.value / 2;
require((2 * value) == msg.value, "Value has to be even.");
consumer = 0x923c1eDfAdB6332254C83BCbAE85B2cA6b9Bb36e;
}
modifier condition(bool _condition) {
require(_condition);
_;
}
modifier onlyConsumer() {
require(msg.sender == consumer, "Only consumer can call this.");
_;
}
modifier onlyDataseller() {
require(msg.sender == dataseller, "Only dataseller can call this.");
_;
}
modifier inState(contractState _state) {
require(state == _state, "Invalid state.");
_;
}
event Aborted(string info, address entityAddress);
0
listen on network
Search with transaction hash or address
creation of DataTrade pending...
Output of incentive.sol
[vm] from: 0x5B3...eddC4 to: DataTrade.(constructor) value: 0 wei data: 0x608...10032 logs: 0 hash: 0x9ba...20583
Debug
>


DEPLOY & RUN TRANSACTIONS
OR
At Address
Load contract from Address
Transactions recorded 3
Deployed Contracts
DATATRADE AT 0XF8E...9FBE8 (MEMOI
Functions in the contract
abort
confirmPurcha...
confirmReceiv...
consumer
dataseller
state
value
Low level interactions
CALLDATA
Transact
Home
Classes.sol
agreement.sol
EnterpriseClasses.sol
Incentives.sol
5 tabs
pragma solidity >=0.4.22 <0.6.0;
contract DataTrade {
uint public value;
/*Define variable owner of the type address*/
address payable public dataseller;
address payable public consumer;
enum contractState { Created, Locked, Inactive, Aborted }
contractState public state;
constructor() public payable {
0
listen on network
Search with transaction hash or address
Detailed Output
[vm] from: 0x5B3...eddC4 to: DataTrade.(constructor) value: 0 wei data: 0x608...10032 logs: 0 hash: 0x9ba...20583
Debug
status
true Transaction mined and execution succeed
transaction hash
0x9ba1da2bdc46b12b2069d1f0d3a43ce8e1503992b6f29e72f6e2d54777a20583
contract address
0xf8e81D47203A594245E36C48e151709F0C19fBe8
from
0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
to
DataTrade.(constructor)
gas
3000000 gas
transaction cost
802296 gas
execution cost
570540 gas
hash
0x9ba1da2bdc46b12b2069d1f0d3a43ce8e1503992b6f29e72f6e2d54777a20583
input
0x608...10032
decoded input
{}
decoded output
-
logs
[]
value
0 wei