

Remix - Ethereum IDE

remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js

GmailYouTubeMapsNewsTranslateNew TabAll Bookmarks

FILE EXPLORER

WORKSPACES

default_workspace

contracts

1_Storage.sol

2_Owner.sol

3_Ballot.sol

scripts

tests

.prettierrc.json

README.txt

REMIX

The Native IDE for Web3 Development.

WebsiteDocumentationRemix PluginRemix Desktop

Search Documentation

Files

New FileOpen FileAccess File System

Load from

GitHubGistIPFSHTTPS

Learn

Remix Basics

An introduction to Remix's interface and basic operations.

Get Started

Intro to Solidity

0

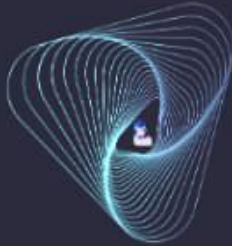
☐ listen on all transactions

Search with transaction hash or address

remix

Type the library name to see available commands.

Featured



WE NEED YOUR HELP

Remixers... Have a spare moment? Please help us improve your Remix experience with this one-minute survey.

[Go to survey](#)

Get Started - Project Templates

GNOSIS SAFE MULTISIG

Create Multi-Signature wallets using this template.


0XPROJECT ERC20

Create an ERC20 token by importing 0xProject contract.

OPENZEPPELIN ERC20


Create an ERC20 token by importing OpenZeppelin librar

Featured Plugins




SOLIDITY ANALYZERS

Analyze your code using Remix



COOKBOOK


Find smart contracts, solidity



SOLIDITY

Compile, test, and analyze con

Type here to search



ENG

19:30

02-10-2023

DEPLOY & RUN TRANSACTIONS

☐ Publish to IPFS

At Address

Transactions recorded **1** ⓘ

Deployed Contracts

COLLEGE CERTIFICATE AT 0XD91...

Balance: 0 ETH

IssueCertificate

certificates

getCertificate

owner

totalCertificates

Low level interactions

CALLDATA

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract collegeCertificate {
5     address public owner;
6
7     struct Certificate {
8         string studentName;
9         string courseName;
10        uint256 DateOfGraduation;
11        uint256 issueDate;
12        address issuer;
13    }
14
15    uint256 public totalCertificates;
16    mapping(uint256 => Certificate) public certificates;
17
18    event CertificateIssued(
19        uint256 indexed certificateId,
20        string studentName,
21        string courseName,
22        uint256 issueDate,
23        address indexed issuer
24    );
```


☐ listen on all transactions

[vm] from: 0x583...eddC4 to: collegeCertificate.(constructor) value: 0 wei data: 0x608...20033 logs: 0 hash: 0x88a...9f358 Debug

DEPLOY & RUN TRANSACTIONS

ENVIRONMENT

Injected Provider - MetaMask

Main (1) network

ACCOUNT

0xE76...D99dD (0 ether)

GAS LIMIT

3000000

VALUE

0 Wei

CONTRACT

collegeCertificate - college.sol

evm version: paris

Deploy

☐ Publish to IPFS

At Address Load contract from Address

Transactions recorded 1 1

Deployed Contracts

Currently you have no contract instances to

```
57
58
59
60
61
62
63
64
65
66
67
68
69
```

Gas estimation failed

Gas estimation errored with the following message (see below). The transaction execution will likely fail. Do you want to force sending?
Returned error: [{"jsonrpc":"2.0","error":{"insufficient funds for transfer","id":"8564970888014016}}

Send Transaction Cancel Transaction

```
uint256, uint256, address) {
    Graduation, cert.issueDate, cert.issuer)
}
```

0 ☐ listen on all transactions 🔍 Search with transaction hash or address

creation of collegeCertificate pending...

✓

At Address	Load contract from Address.
------------	-----------------------------

Transactions recorded **1** ⓘ

Deployed Contracts

COLLEGE CERTIFICATE AT 0XD91...

Balance: 0 ETH

IssueCertificate string studentName, string

certificates ulint256

`getCertificate` uint256 certificateId

owner

totalCertificates

Low level interactions

CALLDATA

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract collegeCertificate {
5     address public owner;
6
7     struct Certificate {
8         string studentName;
9         string courseName;
10        uint256 DateOfGraduation;
11        uint256 issueDate;
12        address issuer;
13    }
14
15    uint256 public totalCertificates;
16    mapping(uint256 => Certificate) public certificates;
17
18    event CertificateIssued(
19        uint256 indexed certificateId,
20        string studentName,
21        string courseName,
22        uint256 issueDate,
23        address indexed issuer
24    );
```

listen on all transactions ☐ Search with transaction hash or address

[vm] from: 0x583...eddC4 to: collegeCertificate.(constructor) value: 0 wei data: 0x608...20033 logs: 0 hash: 0x88a...9f358 Debug