PRACTICAL 5

[CS601] – Cryptography and Blockchain

Date – 27/03/2023 | *By* Aishwarya Suryakant Waghmare, PRN – 2001106059

Title/Aim of the practical :

To Build and Deploy a Modern Web 3.0 Blockchain App.

Apparatus/Tools/ Resources used :

- Lecture Notes
- E-Resources
- E-Book
- Laptop

Theory of the practical :

- ✓ Web 3.0 is the next generation of the internet, which is based on decentralization, open-source, and transparency.
- ✓ Blockchain technology plays a significant role in Web 3.0 as it enables decentralized applications (DApps) to operate without intermediaries.
- ✓ Here are some features that a modern Web 3.0 blockchain app could have :
  - ➢ Decentralized :
    - A Web 3.0 app should be built on a blockchain network to ensure that it is decentralized.
    - This means that the app should be able to operate without a central authority or intermediary.
  - ➢ Smart Contracts :
    - A Web 3.0 app could use smart contracts to automate processes and ensure that transactions are executed according to predefined rules.
  - ➢ Tokenization :
    - Tokenization is the process of representing real-world assets on a blockchain as digital tokens.
    - A Web 3.0 app could use tokens to represent assets such as real estate, stocks, and commodities.
  - ➢ Interoperability :
    - A Web 3.0 app should be able to communicate and exchange data with other apps and networks seamlessly.
  - ➢ Privacy and Security :
    - A Web 3.0 app should provide privacy and security for its users.
    - This could be achieved through the use of cryptographic techniques and zero-knowledge proofs.
  - ➢ User-Friendly Interface :
    - A Web 3.0 app should have a user-friendly interface that makes it easy for users to interact with the app and access its features.
  - ➢ Community-Driven :

- A Web 3.0 app should be community-driven, meaning that it should be developed and maintained by a community of developers and users.

✓ Examples of modern Web 3.0 blockchain apps include decentralized finance (DeFi) applications, non-fungible token (NFT) marketplaces, and social media platforms.

Procedure of the Practical/ Codes:

PickupLines.sol in the picklines/contracts – directory

```
/*Use a license depending on your project.*/
// SPDX-License-Identifier: UNLICENSED

/*Code is written for Solidity version 0.4.16, or a newer version*/
pragma solidity ^0.8.0;

/*Built-in Hardhat interactive JavaScript console*/
import "hardhat/console.sol";

/*Main Solidity Contract*/
contract PickupLines {
  /*Constructor function for our contract*/
   constructor() {
      console.log("I am the Cheesy PickUp Lines' smart contract.");
   }
}
```

run.js in the picklines/scripts directory folder

```
/*The `main` function to run contract locally for an instance.*/
const main = async () => {
 /*Helper function to get the contract `PickupLines`*/
 const contracts = await hre.ethers.getContractFactory("PickupLines");

 /*Deploying the contract for an 'instance'*/
 const contract = await contracts.deploy();
 await contract.deployed();

 /*Address of the deployed contract.*/
 console.log("Contract deployed to:", contract.address);
};
```

```
/*A try-catch block for our `main` function*/
const runMain = async () => {
  try {
    await main();
    process.exit(0); // exit Node process without error
  } catch (error) {
    console.log(error);
    process.exit(1); // exit Node process while indicating 'Uncaught Fatal Exception' error
  }
};

/*Running the `runMain` function.*/
runMain();

updated pickuplines.sol smart contract
contract PickUpLines {
    /*Solidity event, that fires when a new line is submitted.*/
    event NewPickUpLine(address indexed from, uint256 timestamp, string line);

    /*Data members*/
    uint256 private seed; /*Seed data*/
    uint256 totalLines; /*Total lines data*/
    mapping(address => bool) hasWrote; /*Map of all addresses with a line submitted*/

    /*A composite data member for a pick up line*/
    struct PickUpLine {
        address writer;
        string line;
        uint256 timestamp;
    }
    /*Array of all pick up lines submitted.*/
    PickUpLine[] pickuplines;

    constructor() payable {
      console.log("I am the Cheesy PickUp Lines' smart contract!");
    }
```

```
    /*Function for adding a new line to the contract.*/
    function newLine(string memory _line) public {

        /*Adding a new Pickup Line to our blockchain.*/
        totalLines += 1;
        pickuplines.push(PickUpLine(msg.sender, _line, block.timestamp));
        hasWrote[msg.sender] = true;
        emit NewPickUpLine(msg.sender, block.timestamp, _line);
    }

    /*Function to get all the lines submitted to the contract.*/
    function getTotalLines() public view returns (uint256) {
        console.log("We have %s total PickUpLines.", totalLines);
        return totalLines;
    }
}
```

deploy.js in the pickuplines/scripts – directory folder

```
  /*The `main` function to deploy contract locally*/
  const main = async () => {
  /*Getting deployer's address.*/
  const [deployer] = await hre.ethers.getSigners();

  /*Getting deployer's ETH balance*/
  const accountBalance = await deployer.getBalance();

  /*Logging the Deployer's address and the balance.*/
  console.log("Deploying contracts with account: ", deployer.address);
  console.log("Account balance: ", accountBalance.toString());

  /*Deploying the contract.*/
  const contracts = await hre.ethers.getContractFactory("PickupLines");
  const contract = await contracts.deploy();
  await contract.deployed();

  /*Logging the address of the deployed contract.*/
  console.log("PickupLines address: ", contract.address);
```

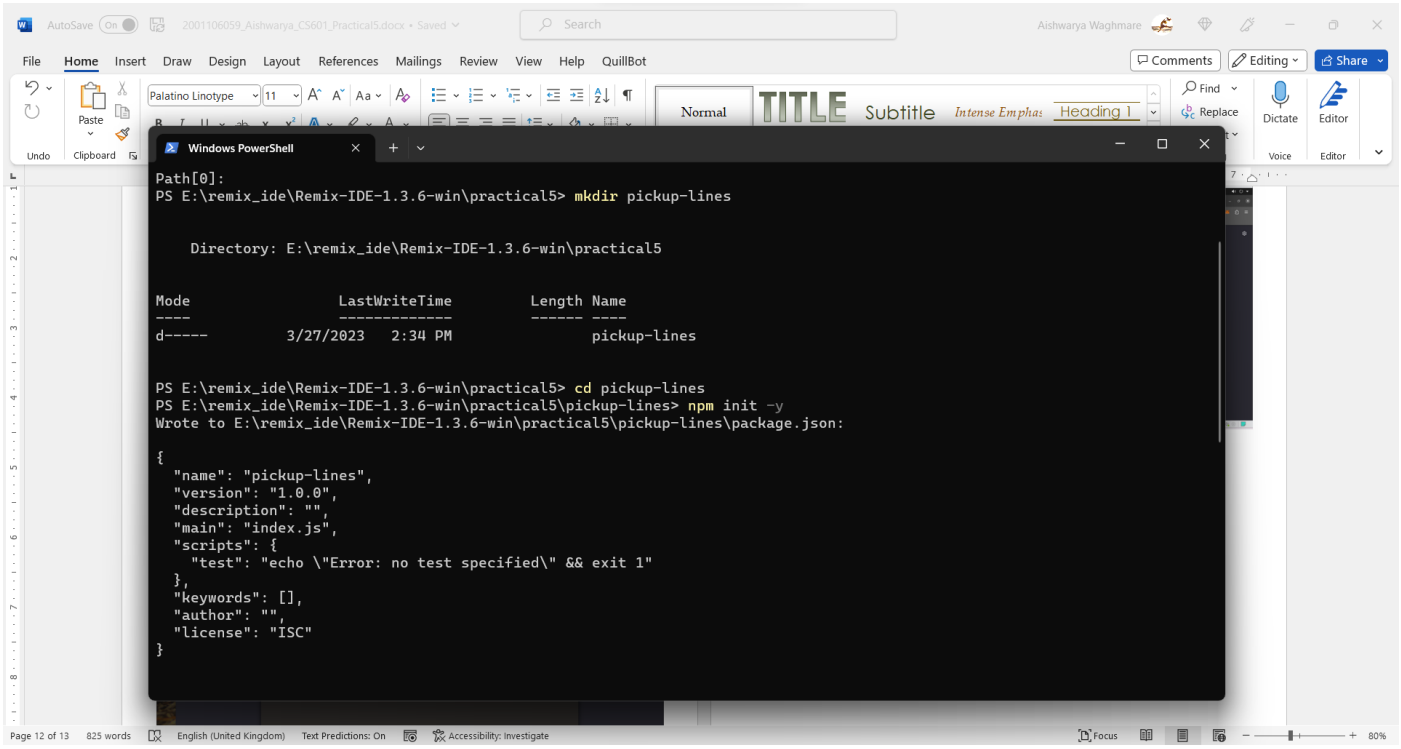
```
};

/*A try-catch block for our `main` function*/
const runMain = async () => {
  try {
    await main();
    process.exit(0);
  } catch (error) {
    console.log(error);
    process.exit(1);
  }
};

/*Running the `runMain` function.*/
runMain();
```

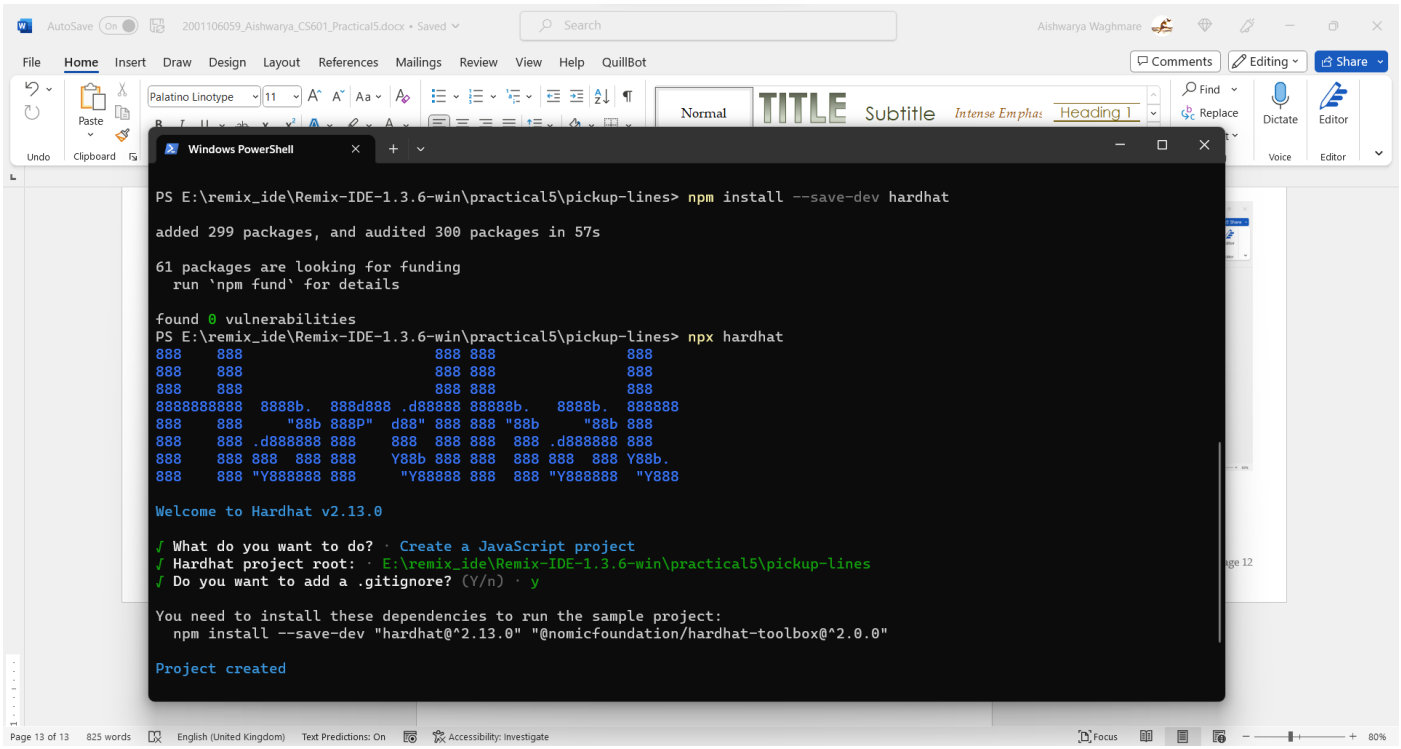
Making changes to hardhat.config.js

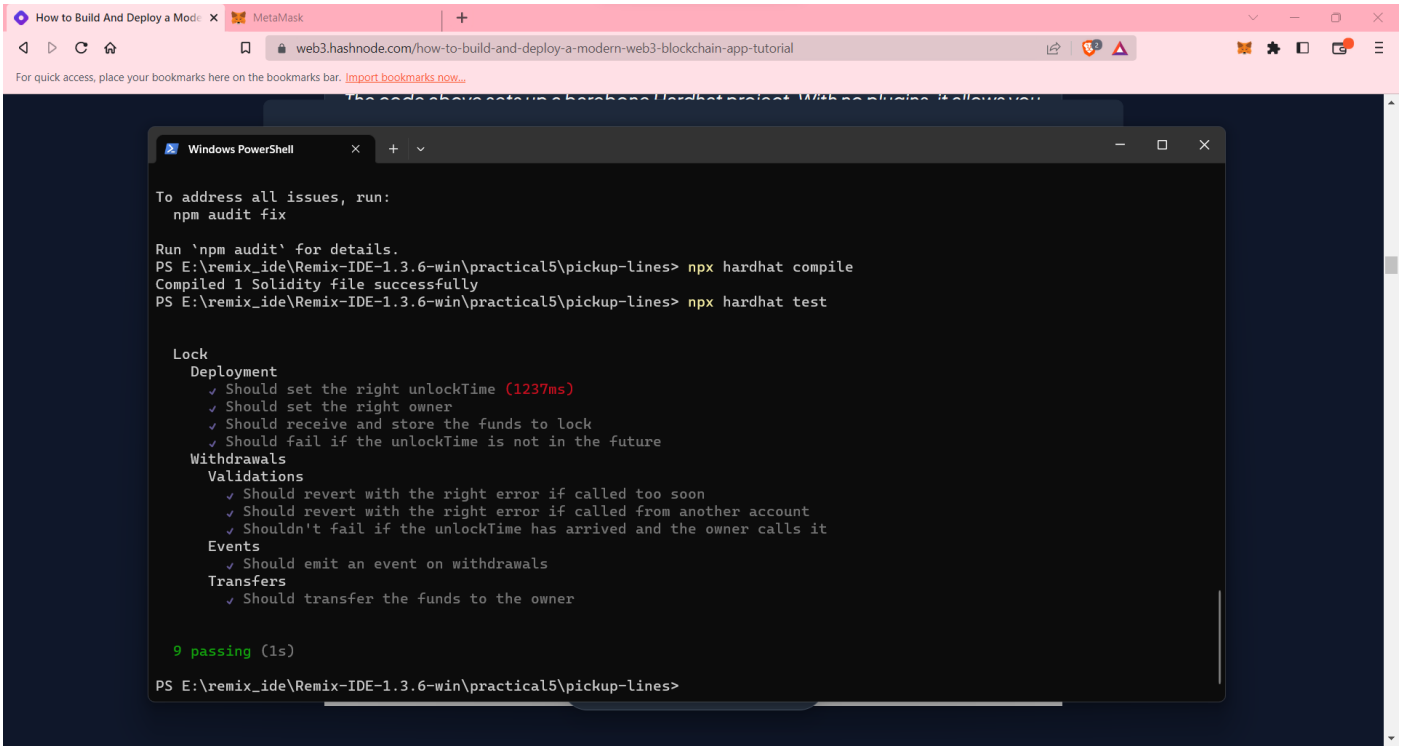
```
//Find YOUR_ALCHEMY_API_URL in the alchemy dashboard.
require("@nomiclabs/hardhat-waffle");

module.exports = {
  solidity: "0.8.0",
  networks: {
    rinkeby: {
      url: "YOUR_ALCHEMY_API_URL",
      accounts: ["YOUR_WALLET_ACCOUNT_KEY"]
    },
  },
};
```

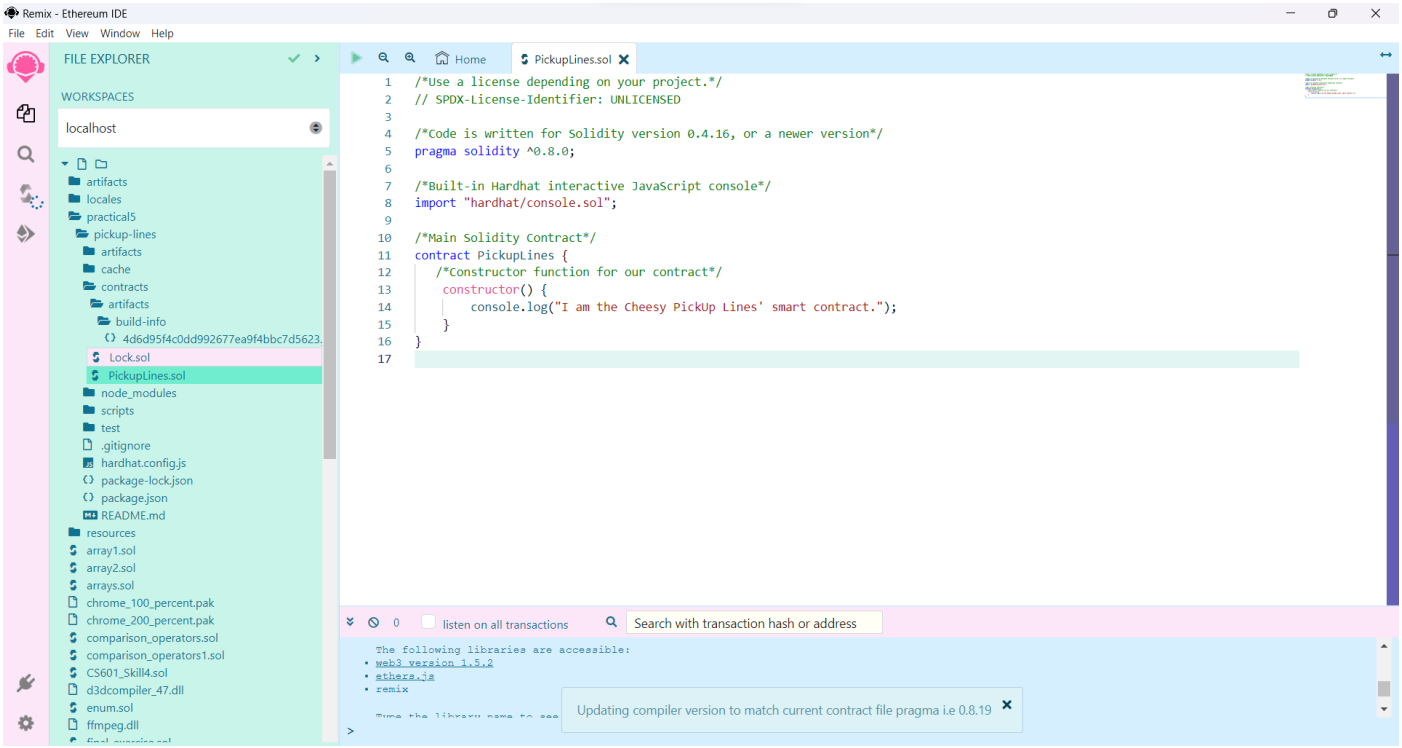
Result/ Output/ Screenshots of the Practical :

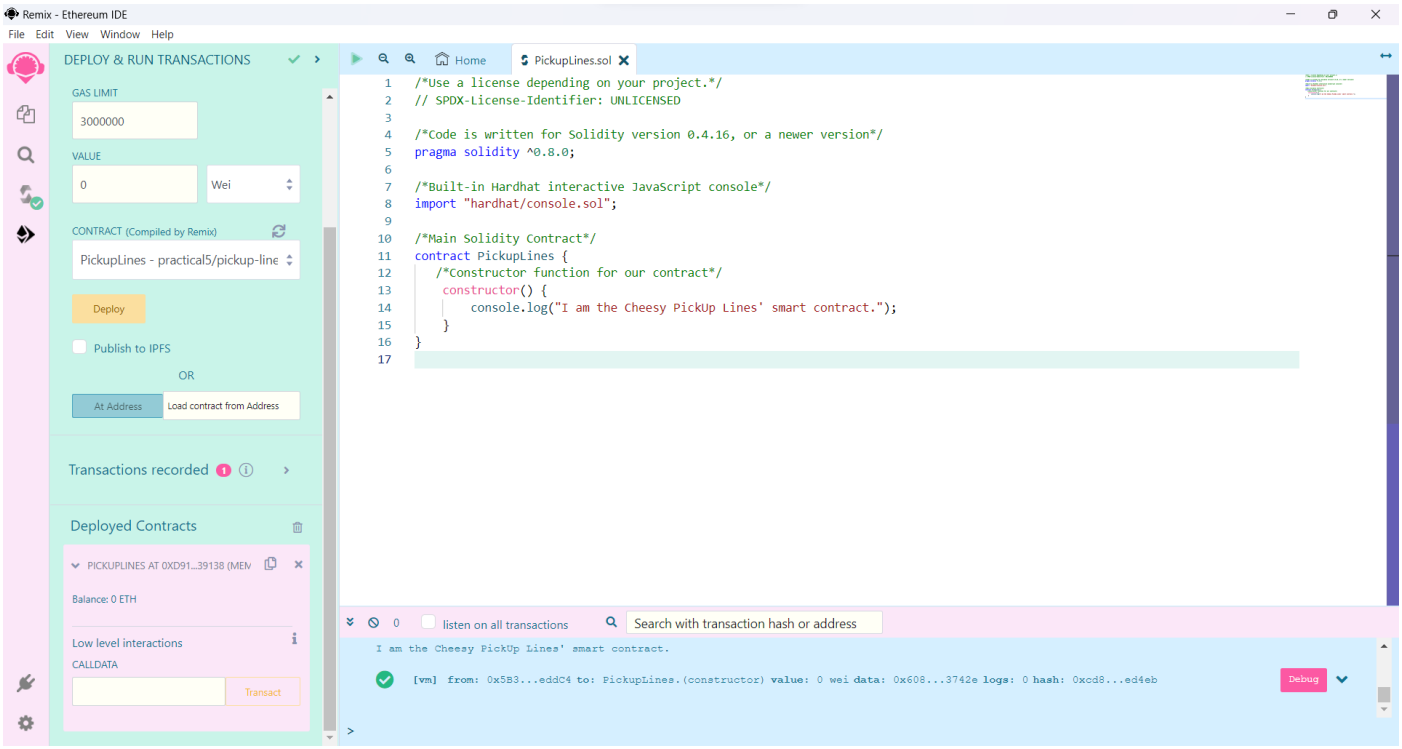
Therefore, built and deployed a modern web3.0 blockchain app on windows.

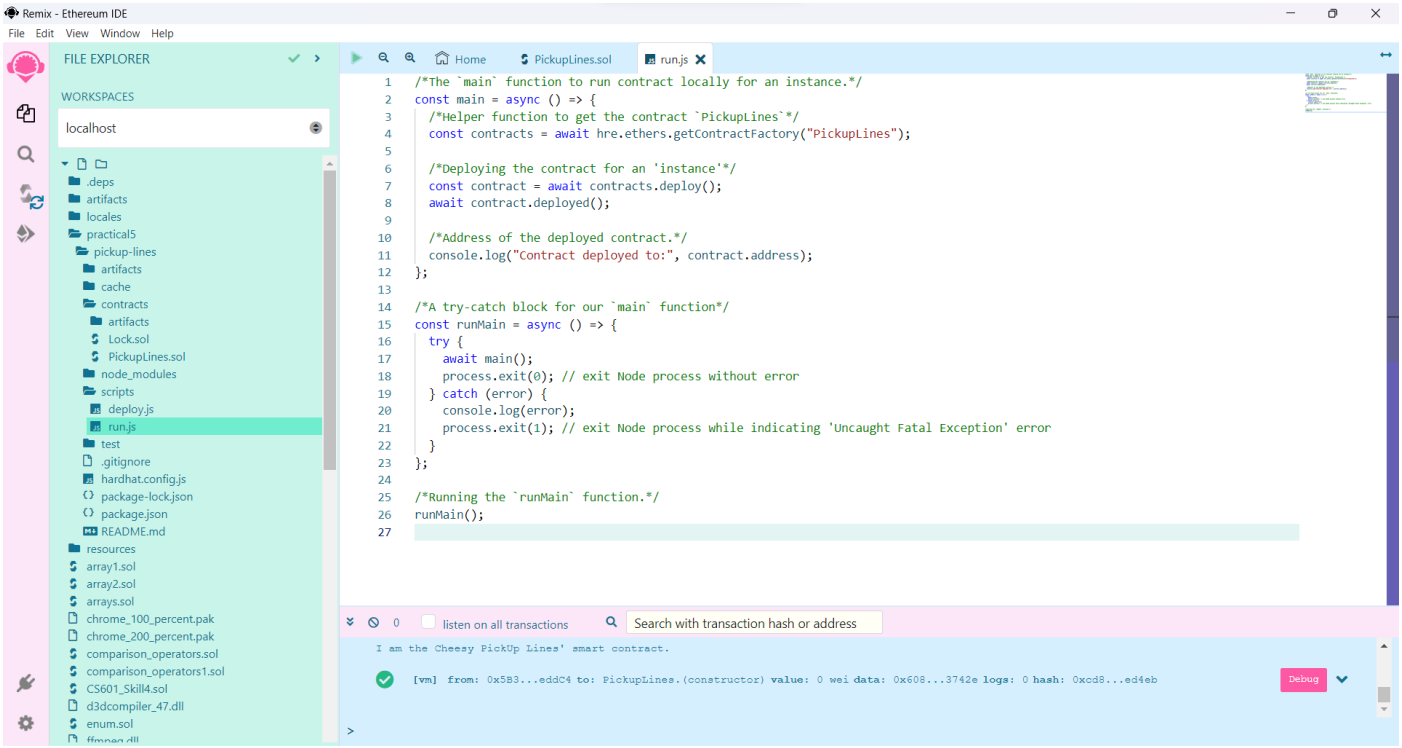


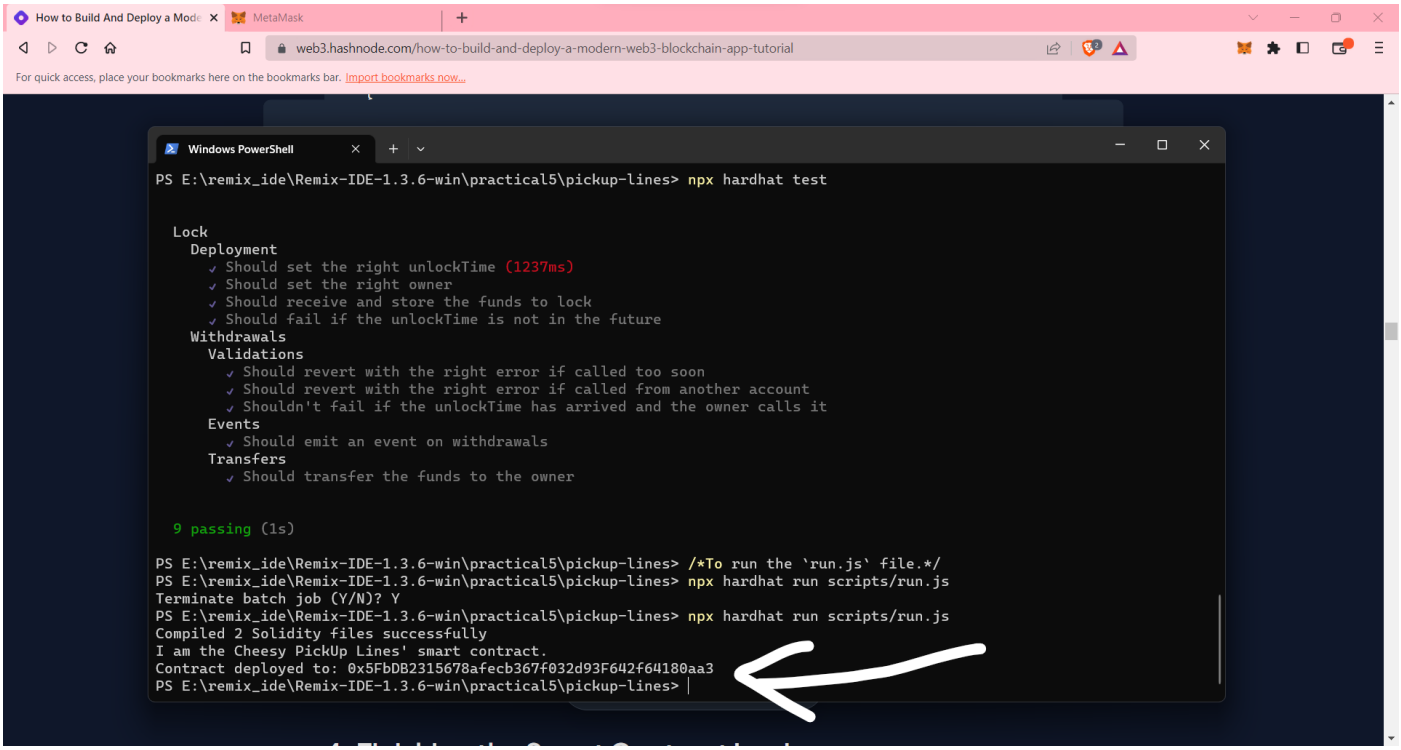


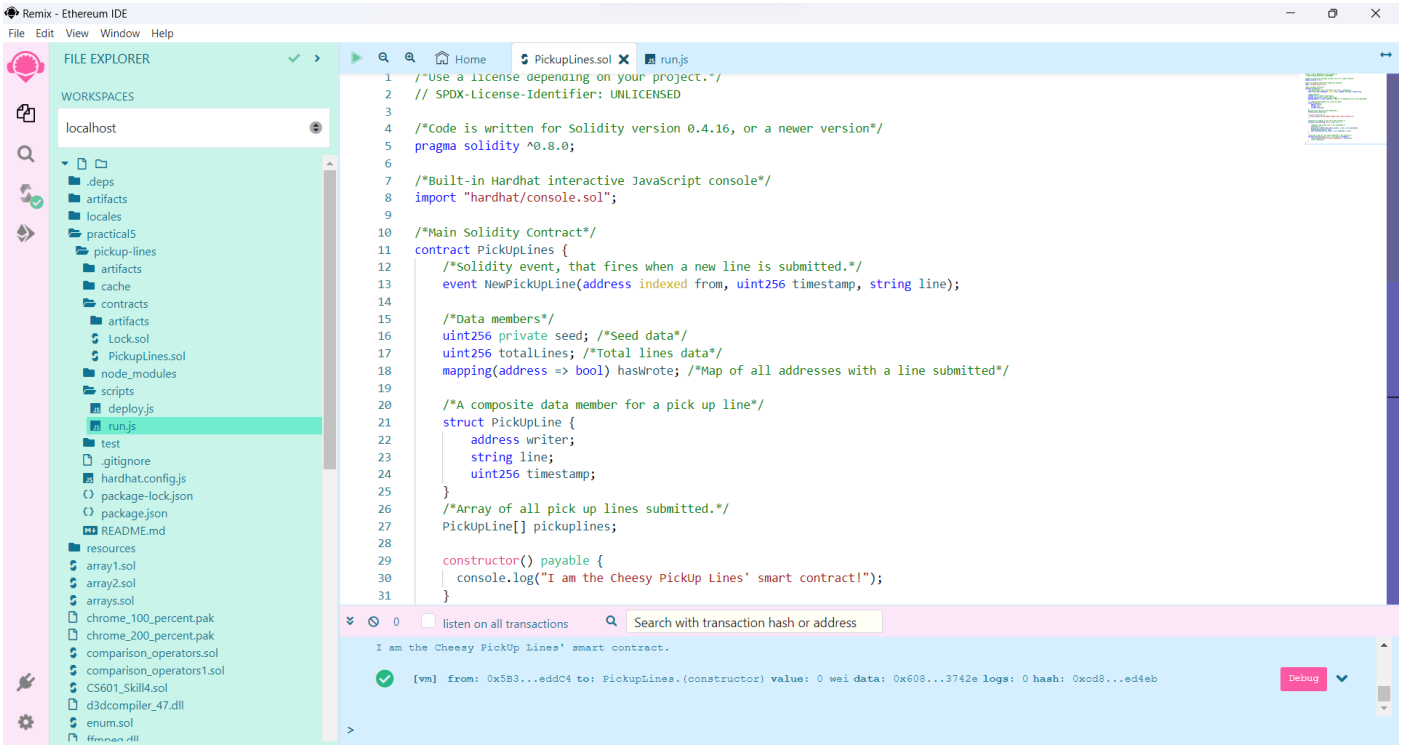

How to Build And Deploy a Mode
MetaMask
web3.hashnode.com/how-to-build-and-deploy-a-modern-web3-blockchain-app-tutorial
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Windows PowerShell
To address all issues, run:
npm audit fix
Run `npm audit` for details.
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat compile
Compiled 1 Solidity file successfully
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat test
Lock
Deployment
Should set the right unlockTime (1237ms)
Should set the right owner
Should receive and store the funds to lock
Should fail if the unlockTime is not in the future
Withdrawals
Validations
Should revert with the right error if called too soon
Should revert with the right error if called from another account
Shouldn't fail if the unlockTime has arrived and the owner calls it
Events
Should emit an event on withdrawals
Transfers
Should transfer the funds to the owner
9 passing (1s)
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines>

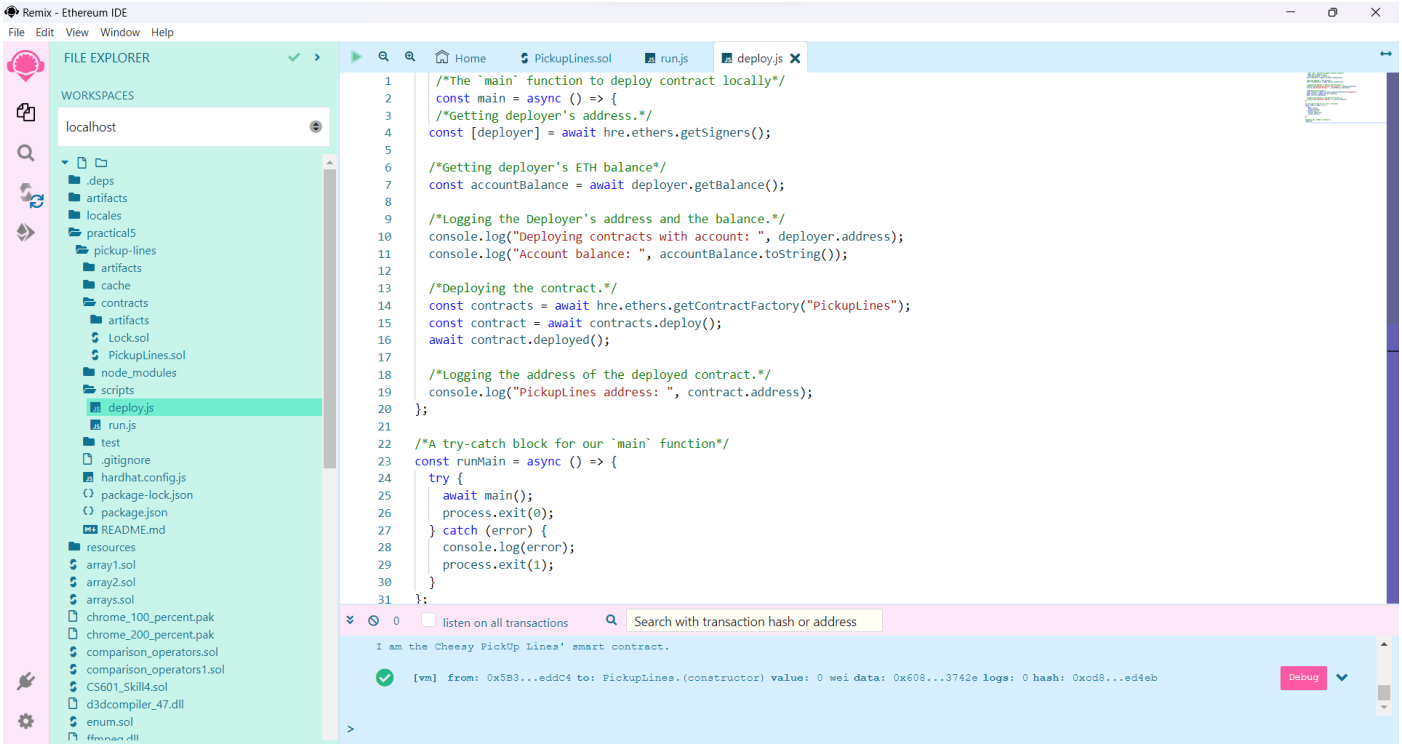

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
artifacts
locales
practical5
pickup-lines
artifacts
cache
contracts
artifacts
build-info
Lock.sol
PickupLines.sol
node_modules
scripts
test
.gitignore
hardhat.config.js
package-lock.json
package.json
README.md
resources
array1.sol
array2.sol
arrays.sol
chrome_100_percent.pak
chrome_200_percent.pak
comparison_operators.sol
comparison_operators1.sol
CS601_Skill4.sol
d3dcompiler_47.dll
enum.sol
ffmpeg.dll
Home
PickupLines.sol
/*Use a license depending on your project.*/
// SPDX-License-Identifier: UNLICENSED
/*Code is written for Solidity version 0.4.16, or a newer version*/
pragma solidity ^0.8.0;
/*Built-in Hardhat interactive JavaScript console*/
import "hardhat/console.sol";
/*Main Solidity Contract*/
contract PickupLines {
/*Constructor function for our contract*/
constructor() {
console.log("I am the Cheesy PickUp Lines' smart contract.");
}
}
listen on all transactions
Search with transaction hash or address
The following libraries are accessible:
web3 version 1.5.2
ethers.js
remix
Updating compiler version to match current contract file pragma i.e 0.8.19

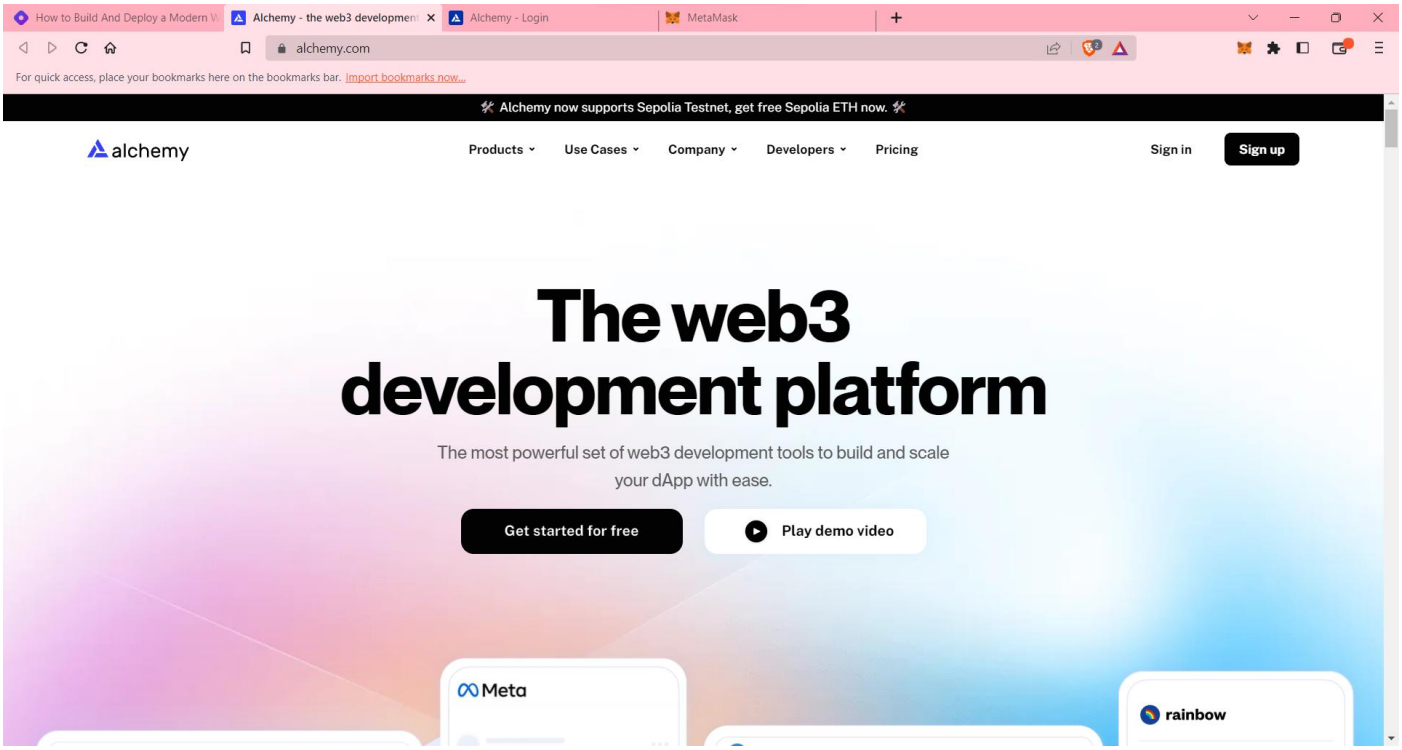

Remix - Ethereum IDE
File Edit View Window Help
DEPLOY & RUN TRANSACTIONS
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT (Compiled by Remix)
PickupLines - practical5/pickup-line
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded
Deployed Contracts
PICKUPLINES AT 0XD91...39138 (MEM
Balance: 0 ETH
Low level interactions
CALLDATA
Transact
Home
PickupLines.sol
/*Use a license depending on your project.*/
// SPDX-License-Identifier: UNLICENSED
/*Code is written for Solidity version 0.4.16, or a newer version*/
pragma solidity ^0.8.0;
/*Built-in Hardhat interactive JavaScript console*/
import "hardhat/console.sol";
/*Main Solidity Contract*/
contract PickupLines {
/*Constructor function for our contract*/
constructor() {
console.log("I am the Cheesy PickUp Lines' smart contract.");
}
}
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug

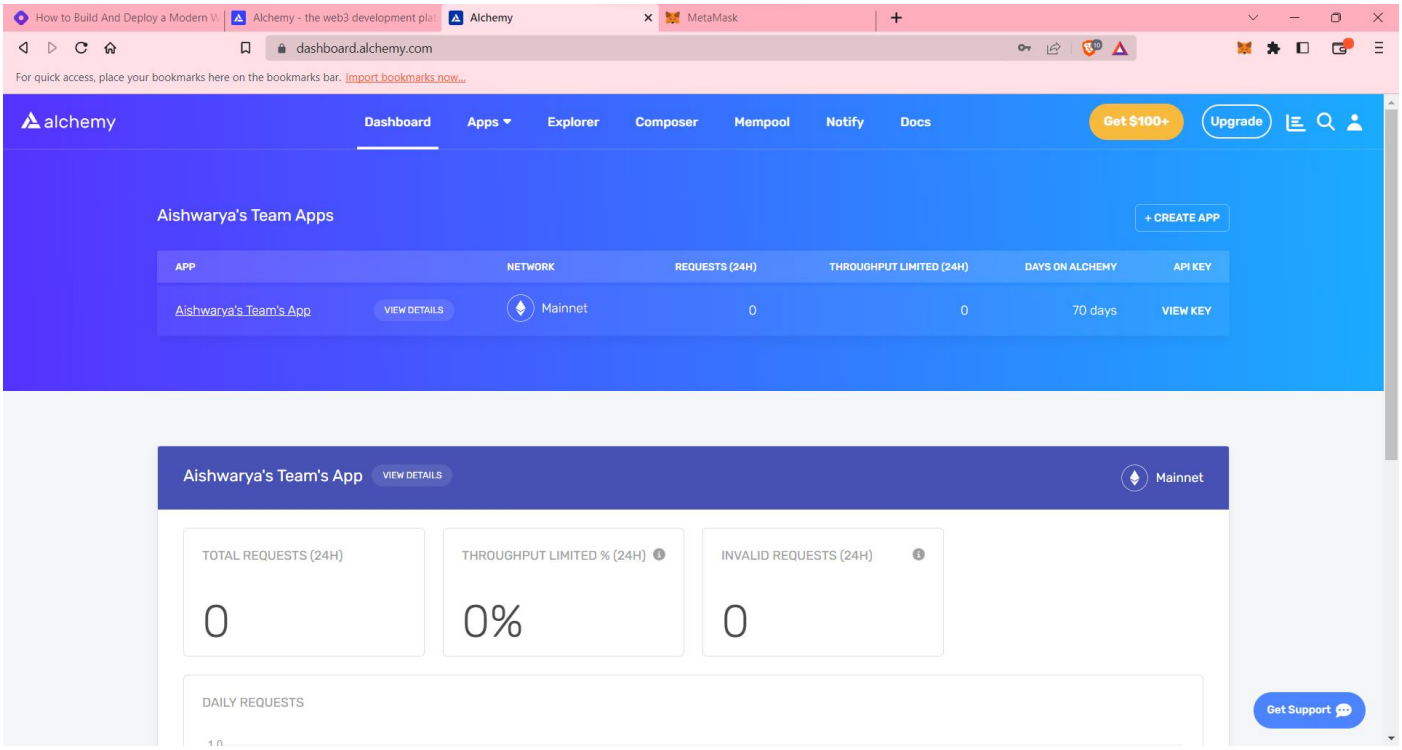

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
.deps
artifacts
locales
practical5
pickup-lines
artifacts
cache
contracts
artifacts
Lock.sol
PickupLines.sol
node_modules
scripts
deploy.js
run.js
test
.gitignore
hardhat.config.js
package-lock.json
package.json
README.md
resources
array1.sol
array2.sol
arrays.sol
chrome_100_percent.pak
chrome_200_percent.pak
comparison_operators.sol
comparison_operators1.sol
CS601_Skill4.sol
d3dcompiler_47.dll
enum.sol
Home
PickupLines.sol
run.js
/*The `main` function to run contract locally for an instance.*/
const main = async () => {
/*Helper function to get the contract `PickupLines`*/
const contracts = await hre.ethers.getContractFactory("PickupLines");
/*Deploying the contract for an 'instance'*/
const contract = await contracts.deploy();
await contract.deployed();
/*Address of the deployed contract.*/
console.log("Contract deployed to:", contract.address);
};
/*A try-catch block for our `main` function*/
const runMain = async () => {
try {
await main();
process.exit(0); // exit Node process without error
} catch (error) {
console.log(error);
process.exit(1); // exit Node process while indicating 'Uncaught Fatal Exception' error
}
};
/*Running the `runMain` function.*/
runMain();
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug

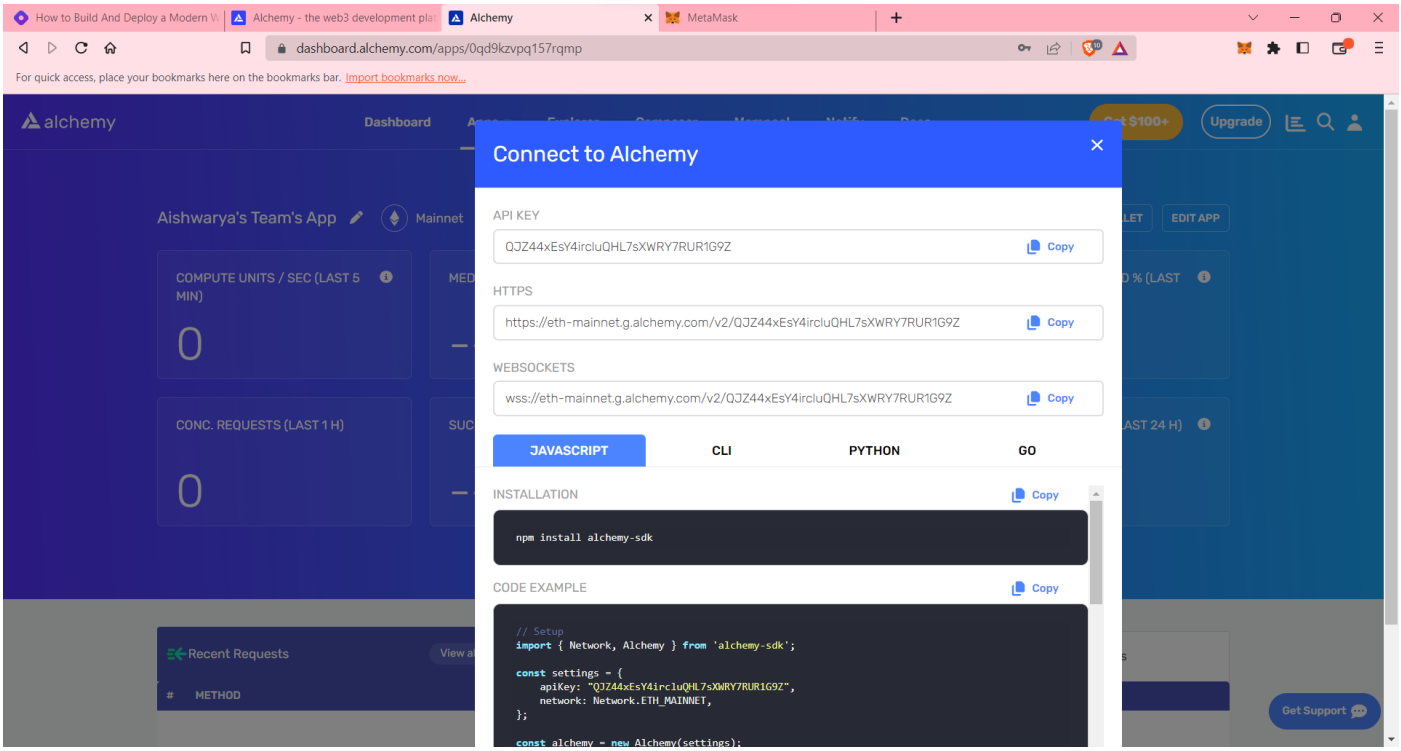

How to Build And Deploy a Mode
MetaMask
web3.hashnode.com/how-to-build-and-deploy-a-modern-web3-blockchain-app-tutorial
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Windows PowerShell
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat test
Lock
Deployment
✓ Should set the right unlockTime (1237ms)
✓ Should set the right owner
✓ Should receive and store the funds to lock
✓ Should fail if the unlockTime is not in the future
Withdrawals
Validations
✓ Should revert with the right error if called too soon
✓ Should revert with the right error if called from another account
✓ Shouldn't fail if the unlockTime has arrived and the owner calls it
Events
✓ Should emit an event on withdrawals
Transfers
✓ Should transfer the funds to the owner
9 passing (1s)
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> /*To run the `run.js` file.*/
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/run.js
Terminate batch job (Y/N)? Y
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/run.js
Compiled 2 Solidity files successfully
I am the Cheesy PickUp Lines' smart contract.
Contract deployed to: 0x5FbDB2315678afecb367f032d93F642f64180aa3
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines>

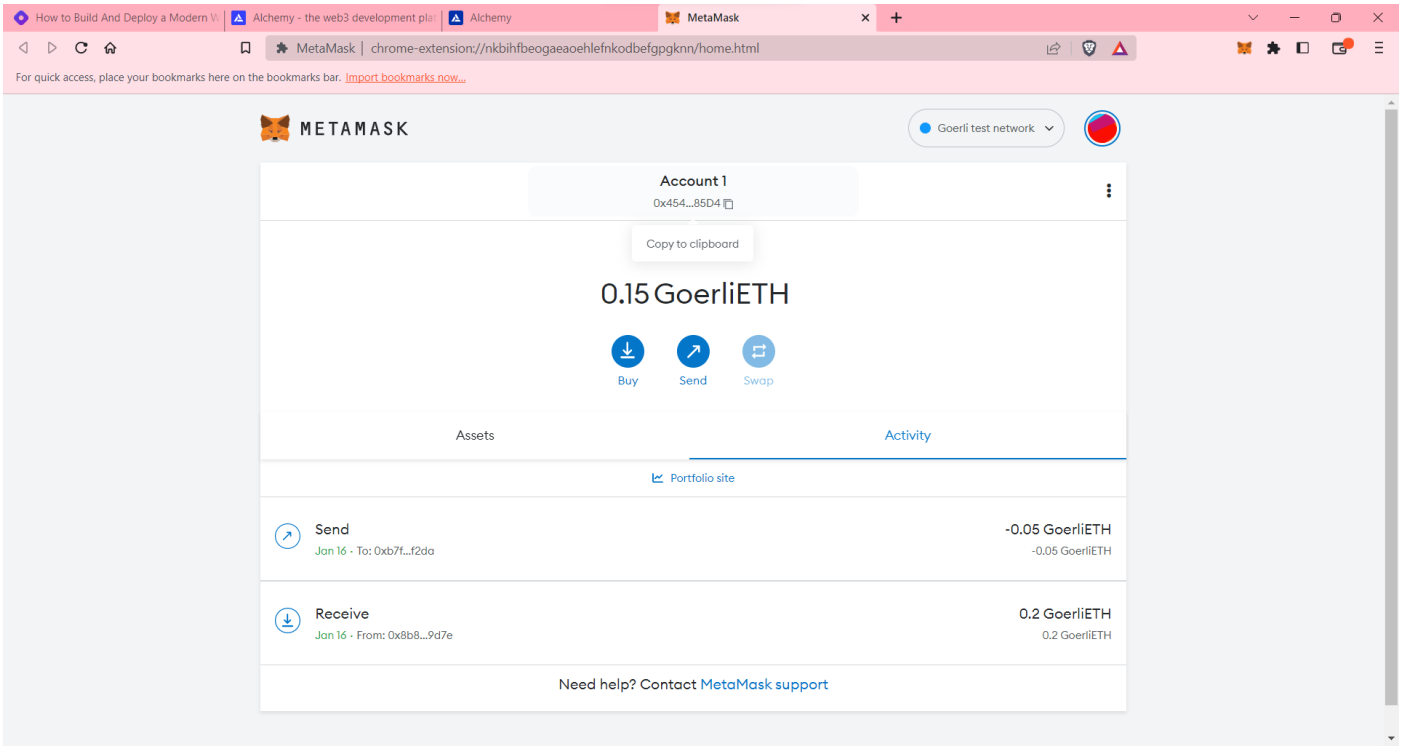

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
.deps
artifacts
locales
practical5
pickup-lines
artifacts
cache
contracts
artifacts
Lock.sol
PickupLines.sol
node_modules
scripts
deploy.js
run.js
test
.gitignore
hardhat.config.js
package-lock.json
package.json
README.md
resources
array1.sol
array2.sol
arrays.sol
chrome_100_percent.pak
chrome_200_percent.pak
comparison_operators.sol
comparison_operators1.sol
CS601_Skill4.sol
d3dcompiler_47.dll
enum.sol
Home
PickupLines.sol
run.js
/*Use a license depending on your project.*/
// SPDX-License-Identifier: UNLICENSED
/*Code is written for Solidity version 0.4.16, or a newer version*/
pragma solidity ^0.8.0;
/*Built-in Hardhat interactive JavaScript console*/
import "hardhat/console.sol";
/*Main Solidity Contract*/
contract PickUpLines {
/*Solidity event, that fires when a new line is submitted.*/
event NewPickUpLine(address indexed from, uint256 timestamp, string line);
/*Data members*/
uint256 private seed; /*Seed data*/
uint256 totalLines; /*Total lines data*/
mapping(address => bool) hasWrote; /*Map of all addresses with a line submitted*/
/*A composite data member for a pick up line*/
struct PickUpLine {
address writer;
string line;
uint256 timestamp;
}
/*Array of all pick up lines submitted.*/
PickUpLine[] pickuplines;
constructor() payable {
console.log("I am the Cheesy PickUp Lines' smart contract!");
}
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug

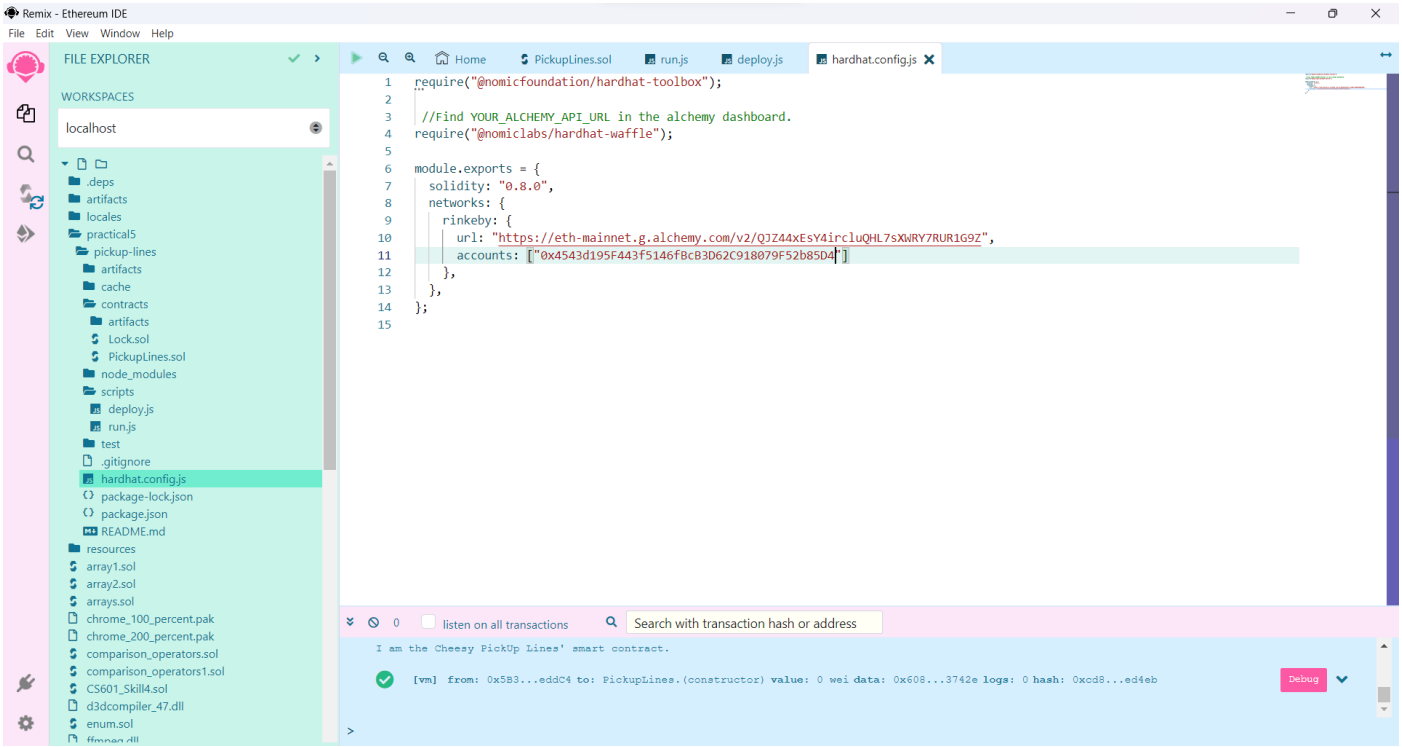

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
.deps
artifacts
locales
practical5
pickup-lines
artifacts
cache
contracts
artifacts
Lock.sol
PickupLines.sol
node_modules
scripts
deploy.js
run.js
test
.gitignore
hardhat.config.js
package-lock.json
package.json
README.md
resources
array1.sol
array2.sol
arrays.sol
chrome_100_percent.pak
chrome_200_percent.pak
comparison_operators.sol
comparison_operators1.sol
CS601_Skill4.sol
d3dcompiler_47.dll
enum.sol
Home
PickupLines.sol
run.js
deploy.js
/*The `main` function to deploy contract locally*/
const main = async () => {
/*Getting deployer's address.*/
const [deployer] = await hre.ethers.getSigners();
/*Getting deployer's ETH balance*/
const accountBalance = await deployer.getBalance();
/*Logging the Deployer's address and the balance.*/
console.log("Deploying contracts with account: ", deployer.address);
console.log("Account balance: ", accountBalance.toString());
/*Deploying the contract.*/
const contracts = await hre.ethers.getContractFactory("PickupLines");
const contract = await contracts.deploy();
await contract.deployed();
/*Logging the address of the deployed contract.*/
console.log("PickupLines address: ", contract.address);
};
/*A try-catch block for our `main` function*/
const runMain = async () => {
try {
await main();
process.exit(0);
} catch (error) {
console.log(error);
process.exit(1);
}
};
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug

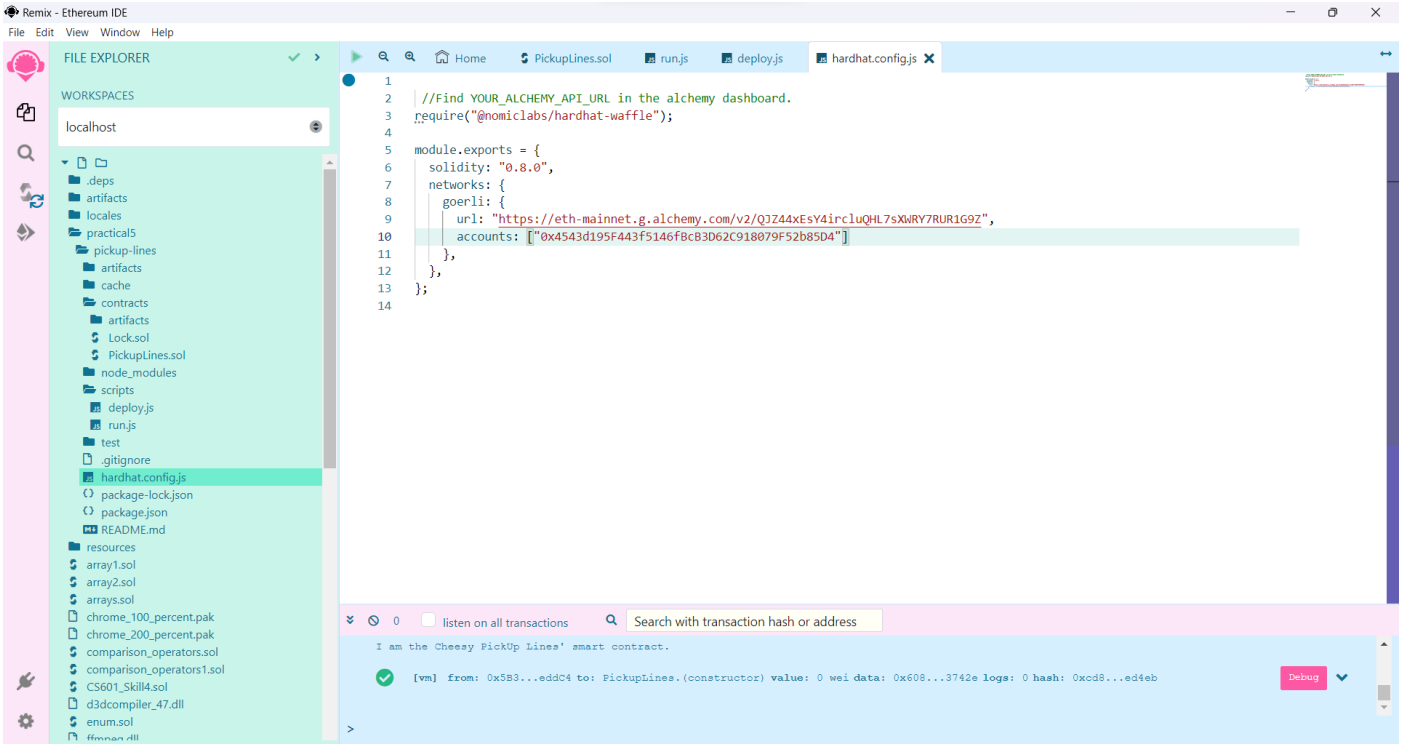

How to Build And Deploy a Modern W
Alchemy - the web3 developmen
Alchemy - Login
MetaMask
alchemy.com
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Alchemy now supports Sepolia Testnet, get free Sepolia ETH now.
alchemy
Products
Use Cases
Company
Developers
Pricing
Sign in
Sign up
The web3 development platform
The most powerful set of web3 development tools to build and scale your dApp with ease.
Get started for free
Play demo video
Meta
rainbow

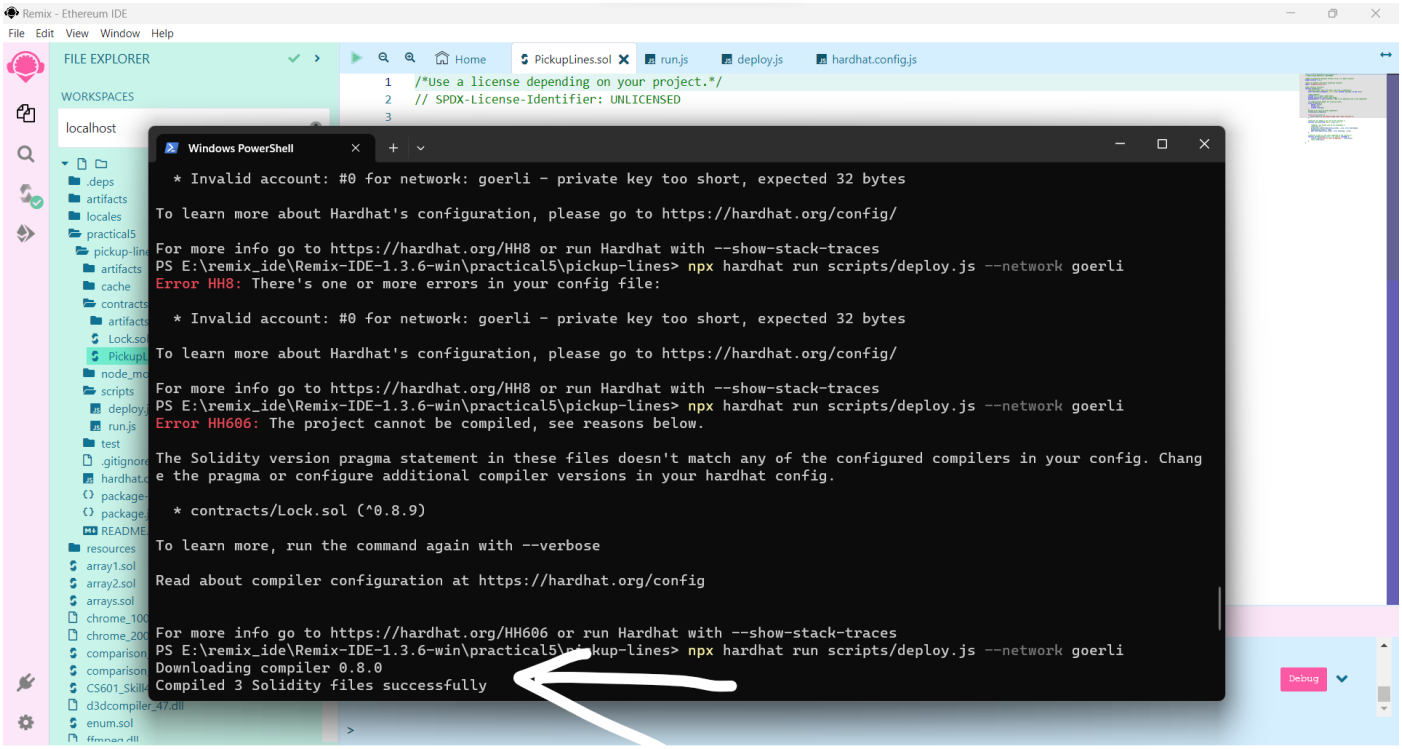

How to Build And Deploy a Modern W
Alchemy - the web3 development pla
Alchemy
MetaMask
dashboard.alchemy.com
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
alchemy
Dashboard
Apps
Explorer
Composer
Mempool
Notify
Docs
Get $100+
Upgrade
Aishwarya's Team Apps
+ CREATE APP
APP
NETWORK
REQUESTS (24H)
THROUGHPUT LIMITED (24H)
DAYS ON ALCHEMY
API KEY
Aishwarya's Team's App
VIEW DETAILS
Mainnet
0
0
70 days
VIEW KEY
Aishwarya's Team's App
VIEW DETAILS
Mainnet
TOTAL REQUESTS (24H)
0
THROUGHPUT LIMITED % (24H)
0%
INVALID REQUESTS (24H)
0
DAILY REQUESTS
Get Support

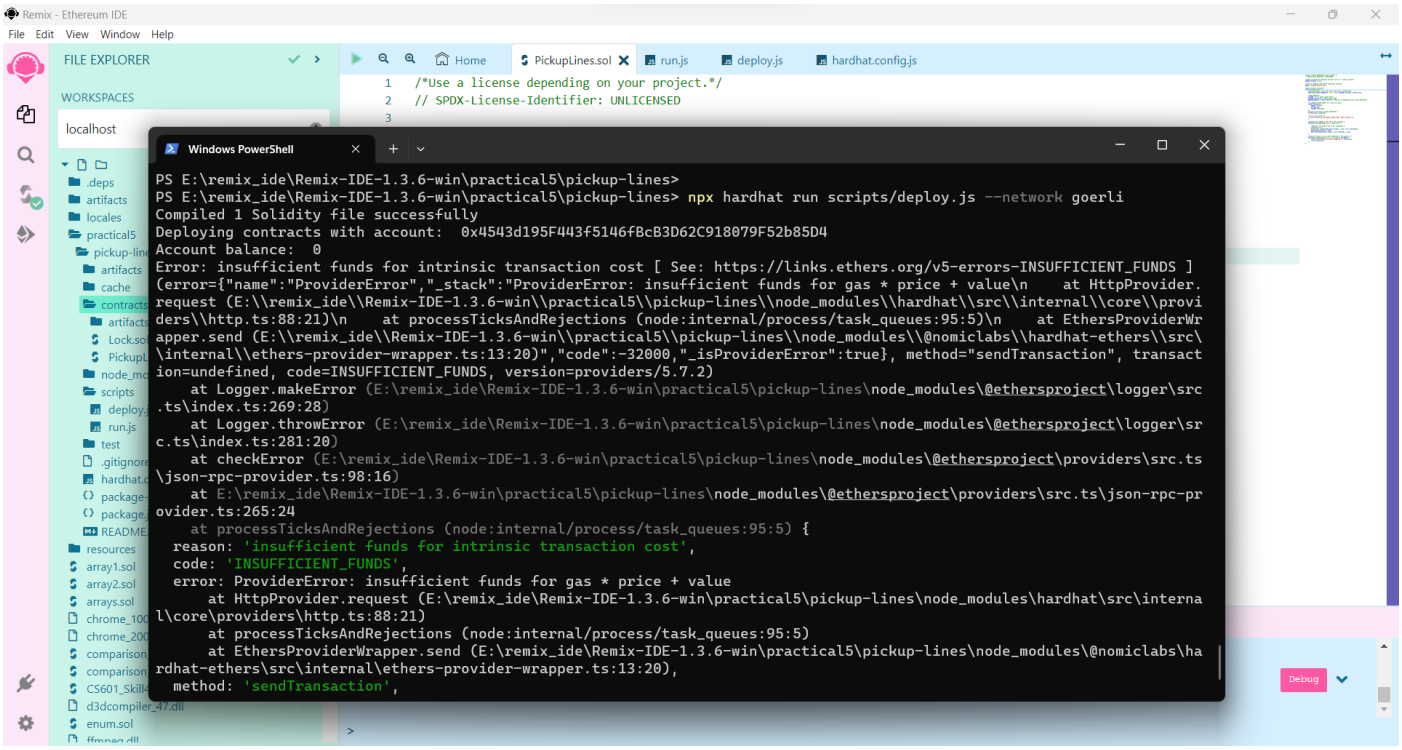

How to Build And Deploy a Modern W
Alchemy - the web3 development pla
Alchemy
MetaMask
dashboard.alchemy.com/apps/0qd9kzvpq157rqmp
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
alchemy
Dashboard
Upgrade
Aishwarya's Team's App
Mainnet
EDIT APP
COMPUTE UNITS / SEC (LAST 5 MIN)
0
CONC. REQUESTS (LAST 1 H)
0
Recent Requests
METHOD
Connect to Alchemy
API KEY
QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z
Copy
HTTPS
https://eth-mainnet.g.alchemy.com/v2/QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z
Copy
WEBSOCKETS
wss://eth-mainnet.g.alchemy.com/v2/QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z
Copy
JAVASCRIPT
CLI
PYTHON
GO
INSTALLATION
Copy
npm install alchemy-sdk
CODE EXAMPLE
Copy
// Setup
import { Network, Alchemy } from 'alchemy-sdk';
const settings = {
apiKey: "QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z",
network: Network.ETH_MAINNET,
};
const alchemy = new Alchemy(settings);
Get Support

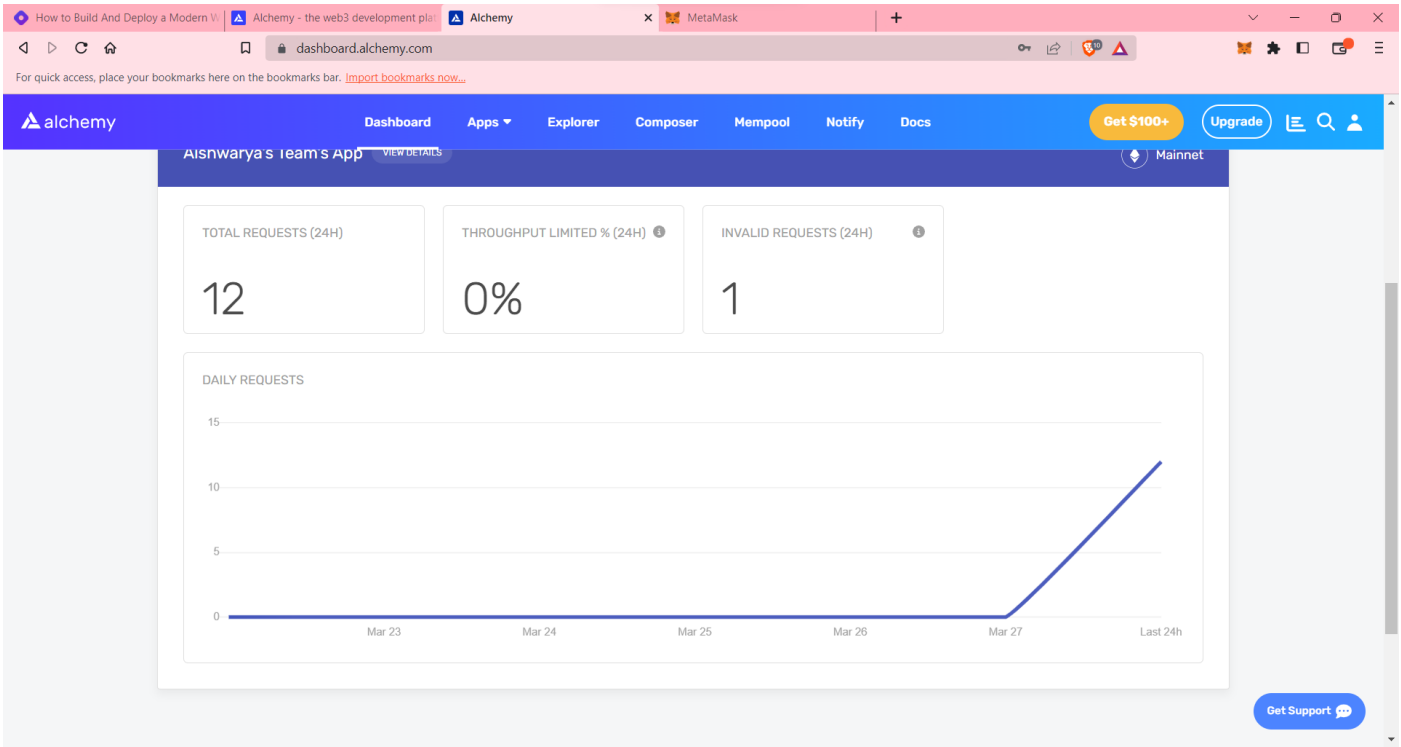

How to Build And Deploy a Modern W
Alchemy - the web3 development pla
Alchemy
MetaMask
MetaMask | chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
METAMASK
Goerli test network
Account 1
0x454...85D4
Copy to clipboard
0.15 GoerliETH
Buy
Send
Swap
Assets
Activity
Portfolio site
Send
Jan 16 · To: 0xb7f...f2da
-0.05 GoerliETH
-0.05 GoerliETH
Receive
Jan 16 · From: 0x8b8...9d7e
0.2 GoerliETH
0.2 GoerliETH
Need help? Contact MetaMask support

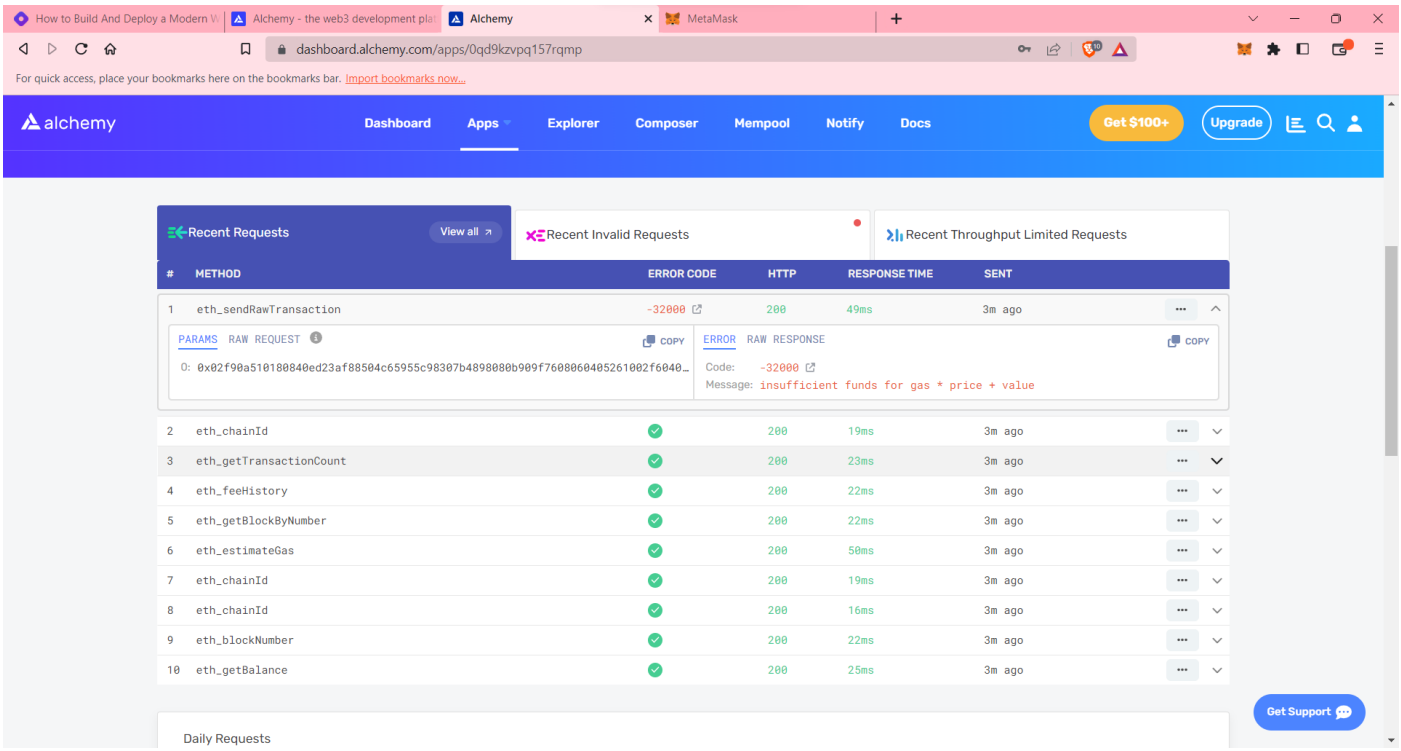

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
.deps
artifacts
locales
practical5
pickup-lines
artifacts
cache
contracts
artifacts
Lock.sol
PickupLines.sol
node_modules
scripts
deploy.js
run.js
test
.gitignore
hardhat.config.js
package-lock.json
package.json
README.md
resources
array1.sol
array2.sol
arrays.sol
chrome_100_percent.pak
chrome_200_percent.pak
comparison_operators.sol
comparison_operators1.sol
CS601_Skill4.sol
d3dcompiler_47.dll
enum.sol
Home
PickupLines.sol
run.js
deploy.js
hardhat.config.js
require("@nomicfoundation/hardhat-toolbox");
//Find YOUR_ALCHEMY_API_URL in the alchemy dashboard.
require("@nomiclabs/hardhat-waffle");
module.exports = {
solidity: "0.8.0",
networks: {
rinkeby: {
url: "https://eth-mainnet.g.alchemy.com/v2/QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z",
accounts: ["0x4543d195F443f5146fBcB3D62C918079F52b85D4"]
},
},
};
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug

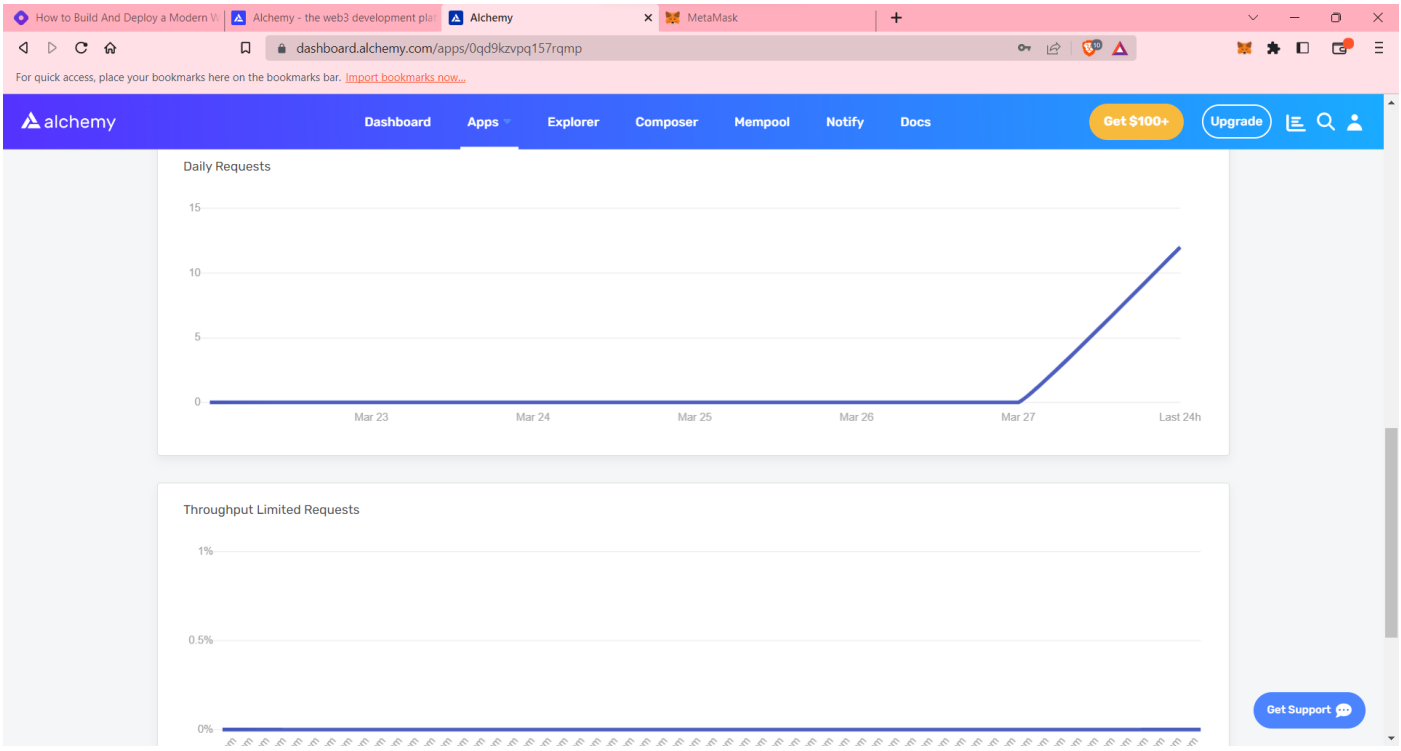

Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
Home
PickupLines.sol
run.js
deploy.js
hardhat.config.js
//Find YOUR_ALCHEMY_API_URL in the alchemy dashboard.
require("@nomiclabs/hardhat-waffle");
module.exports = {
solidity: "0.8.0",
networks: {
goerli: {
url: "https://eth-mainnet.g.alchemy.com/v2/QJZ44xEsY4ircluQHL7sXWRY7RUR1G9Z",
accounts: ["0x4543d195F443f5146fBcB3D62C918079F52b85D4"]
},
},
};
listen on all transactions
Search with transaction hash or address
I am the Cheesy PickUp Lines' smart contract.
[vm] from: 0x5B3...eddC4 to: PickupLines.(constructor) value: 0 wei data: 0x608...3742e logs: 0 hash: 0xcd8...ed4eb
Debug


Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
Home
PickupLines.sol
run.js
deploy.js
hardhat.config.js
/*Use a license depending on your project.*/
// SPDX-License-Identifier: UNLICENSED
Windows PowerShell
* Invalid account: #0 for network: goerli - private key too short, expected 32 bytes
To learn more about Hardhat's configuration, please go to https://hardhat.org/config/
For more info go to https://hardhat.org/HH8 or run Hardhat with --show-stack-traces
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/deploy.js --network goerli
Error HH8: There's one or more errors in your config file:
* Invalid account: #0 for network: goerli - private key too short, expected 32 bytes
To learn more about Hardhat's configuration, please go to https://hardhat.org/config/
For more info go to https://hardhat.org/HH8 or run Hardhat with --show-stack-traces
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/deploy.js --network goerli
Error HH606: The project cannot be compiled, see reasons below.
The Solidity version pragma statement in these files doesn't match any of the configured compilers in your config. Chang
e the pragma or configure additional compiler versions in your hardhat config.
* contracts/Lock.sol (^0.8.9)
To learn more, run the command again with --verbose
Read about compiler configuration at https://hardhat.org/config
For more info go to https://hardhat.org/HH606 or run Hardhat with --show-stack-traces
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/deploy.js --network goerli
Downloading compiler 0.8.0
Compiled 3 Solidity files successfully
Debug


Remix - Ethereum IDE
File Edit View Window Help
FILE EXPLORER
WORKSPACES
localhost
Home
PickupLines.sol
run.js
deploy.js
hardhat.config.js
1 /*Use a license depending on your project.*/
2 // SPDX-License-Identifier: UNLICENSED
3
Windows PowerShell
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines>
PS E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines> npx hardhat run scripts/deploy.js --network goerli
Compiled 1 Solidity file successfully
Deploying contracts with account: 0x4543d195F443f5146fBcB3D62C918079F52b85D4
Account balance: 0
Error: insufficient funds for intrinsic transaction cost [See: https://links.ethers.org/v5-errors-INSUFFICIENT_FUNDS]
(error={"name":"ProviderError","_stack":"ProviderError: insufficient funds for gas * price + value\n at HttpProvider.
request (E:\\remix_ide\\Remix-IDE-1.3.6-win\\practical5\\pickup-lines\\node_modules\\hardhat\\src\\internal\\core\\provi
ders\\http.ts:88:21)\n at processTicksAndRejections (node:internal/process/task_queues:95:5)\n at EthersProviderWr
apper.send (E:\\remix_ide\\Remix-IDE-1.3.6-win\\practical5\\pickup-lines\\node_modules\\@nomiclabs\\hardhat-ethers\\src\
\internal\\ethers-provider-wrapper.ts:13:20)","code":-32000,"_isProviderError":true}, method="sendTransaction", transact
ion=undefined, code=INSUFFICIENT_FUNDS, version=providers/5.7.2)
at Logger.makeError (E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\@ethersproject\logger\src
.ts\index.ts:269:28)
at Logger.throwError (E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\@ethersproject\logger\sr
c.ts\index.ts:281:20)
at checkError (E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\@ethersproject\providers\src.ts
\json-rpc-provider.ts:98:16)
at E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\@ethersproject\providers\src.ts\json-rpc-pr
ovider.ts:265:24
at processTicksAndRejections (node:internal/process/task_queues:95:5) {
reason: 'insufficient funds for intrinsic transaction cost',
code: 'INSUFFICIENT_FUNDS',
error: ProviderError: insufficient funds for gas * price + value
at HttpProvider.request (E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\hardhat\src\interna
l\core\providers\http.ts:88:21)
at processTicksAndRejections (node:internal/process/task_queues:95:5)
at EthersProviderWrapper.send (E:\remix_ide\Remix-IDE-1.3.6-win\practical5\pickup-lines\node_modules\@nomiclabs\ha
rdhat-ethers\src\internal\ethers-provider-wrapper.ts:13:20),
method: 'sendTransaction',
Debug


How to Build And Deploy a Modern W
Alchemy - the web3 development plat
Alchemy
MetaMask
dashboard.alchemy.com
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
alchemy
Dashboard
Apps
Explorer
Composer
Mempool
Notify
Docs
Get $100+
Upgrade
Aishwarya's Team's App
VIEW DETAILS
Mainnet
TOTAL REQUESTS (24H)
12
THROUGHPUT LIMITED % (24H)
0%
INVALID REQUESTS (24H)
1
DAILY REQUESTS
15
10
5
0
Mar 23
Mar 24
Mar 25
Mar 26
Mar 27
Last 24h
Get Support





Parameters achieved/ Conclusion :

Therefore, understood – implemented and successfully built and deployed a modern web3.0 blockchain app.