

The screenshot shows the Microsoft Visual Studio Code (VS Code) interface. The top bar includes standard menu items like File, Edit, Selection, View, Go, Run, etc., followed by navigation icons (back, forward, search). A search bar labeled "Audit" is positioned at the top right. The left sidebar features a "File Explorer" icon, a "Search" icon, a "Problems" icon, a "File" icon, a "Run" icon, a "Terminal" icon, and a "Help" icon. The main workspace displays the "package.json" file for an "audit" project. The file content is as follows:

```
1  {
2    "name": "audit",
3    "version": "1.0.0",
4    "main": "index.js",
5    "scripts": {
6      "test": "echo \\\"Error: no test specified\\\" && exit 1"
7    },
8    "keywords": [],
9    "author": "",
10   "license": "ISC",
11   "description": "",
12   "devDependencies": {
13     "@nomicfoundation/hardhat-chai-matchers": "^2.1.0",
14     "@nomicfoundation/hardhat-ethers": "^3.1.3",
15     "@nomicfoundation/hardhat-ignition": "^0.15.16",
16     "@nomicfoundation/hardhat-ignition-ethers": "^0.15.17",
17     "@nomicfoundation/hardhat-network-helpers": "^1.1.2",
18     "@nomicfoundation/hardhat-toolbox": "^5.0.0",
19     "@nomicfoundation/hardhat-verify": "^2.1.3",
20     "@nomicfoundation/ignition-core": "^0.15.15",
21     "@typechain/ethers-v6": "^0.5.1",
22     "@typechain/hardhat": "^9.1.0",
23     "@types/chai": "^4.3.20",
24     "@types/mocha": "^10.0.10",
25     "chai": "^4.5.0",
26     "ethers": "^6.16.0",
27     "hardhat": "^2.22.10",
28     "hardhat-gas-reporter": "^1.0.10",
29     "solidity-coverage": "^0.8.17",
30   }
31 }
```

The status bar at the bottom provides information about the file: "Ln 35, Col 1" and "Spaces: 2". It also indicates the file is in "JSON" format. Other status indicators include "17°C Light rain" in the bottom-left corner and system status icons in the bottom-right corner.

A screenshot of the Microsoft Visual Studio Code (VS Code) interface. The top bar shows the menu: File, Edit, Selection, View, Go, Run, ... followed by navigation icons (back, forward, search). A search bar contains the text "Audit". On the right are window control buttons (minimize, maximize, close) and a status bar showing battery level, signal strength, and system time.

The left sidebar (Explorer) lists files and folders:

- OPEN EDITORS: hardhat.config.js
- OUTLINE
- AUDIT:
 - artifacts
 - cache
 - contracts
 - node_modules
 - scripts
 - test
- hardhat.config.js
- package-lock.json
- package.json

The main editor area displays the content of the "hardhat.config.js" file:

```
// hardhat.config.js
require("@nomicfoundation/hardhat-toolbox");

/** @type import('hardhat/config').HardhatUserConfig */
module.exports = {
  solidity: "0.8.24", // Use a stable Solidity version
};
```

The status bar at the bottom provides build information: 0 errors, 0 warnings, and 0 critical errors. It also shows file statistics: Line 8, Column 1, Spaces: 2, UTF-8, CRLF, and a JavaScript icon. Other status icons include a weather icon (7°C, Light rain), a search icon, a file icon, a calendar icon, a browser icon, a messaging icon (16 notifications), and a video icon. The bottom right shows the system status: ENG IN, battery level, signal, volume, and the date/time: 14:35, 31-12-2025.

The screenshot shows a Microsoft Visual Studio Code (VS Code) window with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** Audit
- Icons:** Explorer, Open Editors, Outline, Audit, Artifacts, Cache, Contracts, Node_modules, Scripts, Test, hardhat.config.js, package-lock.json, package.json.
- Editor Area:** The active editor is titled "Counter.sol X" and is located at "contracts > Counter.sol".

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.24;
3
4 contract Counter {
5     uint256 public count;
6
7     event Incremented(uint256 newCount);
8     event Decremented(uint256 newCount);
9
10    function increment() public {
11        count += 1;
12        emit Incremented(count);
13    }
14
15    function decrement() public {
16        require(count > 0, "Counter: cannot decrement below zero");
17        count -= 1;
18        emit Decremented(count);
19    }
20
21    function getCount() public view returns (uint256) {
22        return count;
23    }
24 }
```
- Bottom Status Bar:** Shows file status (0 changes), search results (0), audit findings (0), line (Ln 24, Col 2), spaces (Spaces: 4), encoding (UTF-8), line endings (CRLF), plain text, finish setup, go live, and a bell icon.
- System Icons:** Weather (7°C, Light rain), Search, Taskbar icons (calculator, file, browser, terminal, messaging, video player), system status (signal, battery, volume), and date/time (14:35, 31-12-2025).

EXPLORER

OPEN EDITORS

JS Counter.js test

OUTLINE

AUDIT

artifacts

cache

contracts

Counter.sol

node_modules

scripts

test

JS Counter.js

hardhat.config.js

package-lock.json

package.json

JS Counter.js X

```
test > JS Counter.js > ⚠️ describe("Counter Contract") callback
1  const { expect } = require("chai");
2  const { ethers } = require("hardhat");
3
4  describe("Counter Contract", function () {
5    let counter;
6
7    beforeEach(async function () {
8      // Deploy the contract before each test
9      const Counter = await ethers.getContractFactory("Counter");
10     counter = await Counter.deploy();
11   });
12
13   it("Should initialize with a count of 0", async function () {
14     expect(await counter.getCount()).to.equal(0);
15   });
16
17   it("Should increment the count", async function () {
18     await counter.increment();
19     expect(await counter.getCount()).to.equal(1);
20   });
21
22   it("Should decrement the count", async function () {
23     await counter.increment();
24     await counter.decrement();
25     expect(await counter.getCount()).to.equal(0);
26   });
27
28   it("Should fail when decrementing below zero", async function () {
29     await expect(counter.decrement()).to.be.revertedWith("Counter: cannot decrement below zero");
30   });
31
32   it("Should emit an event on increment", async function () {
33     await expect(counter.increment())
34       .to.emit(counter, "Incremented")
35       .withArgs(1);
36   });
37 });
```



File Edit Selection View Go Run ... ← → Q Audit

EXPLORER ... PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

powershell + ×

OPEN EDITORS

OUTLINE

AUDIT

artifacts

cache

contracts

Counter.sol

node_modules

scripts

test

hardhat.config.js

package-lock.json

package.json

PS C:\codes\Audit> npx hardhat test

PS C:\codes\Audit> npx hardhat compile

Compiled 1 Solidity file successfully (evm target: paris).

PS C:\codes\Audit> npx hardhat test

Counter Contract

- ✓ Should initialize with a count of 0
- ✓ Should increment the count
- ✓ Should decrement the count
- ✓ Should fail when decrementing below zero (47ms)
- ✓ Should emit an event on increment

5 passing (2s)

Counter Contract

- ✓ Should initialize with a count of 0
- ✓ Should increment the count
- ✓ Should decrement the count
- ✓ Should fail when decrementing below zero (47ms)
- ✓ Should emit an event on increment

5 passing (2s)

✓ Should emit an event on increment

5 passing (2s)

5 passing (2s)

PS C:\codes\Audit>

Finish Setup Go Live

17°C Light rain

Search

14:34 31-12-2025

Hardhat Installation And Counter Example

This document provides a minimal, working example of a Hardhat project with a simple Solidity `Counter` contract and a JavaScript test file that runs without errors.

The setup is configured to use stable versions of Hardhat and its dependencies to ensure compatibility and avoid common installation issues encountered in the sandbox environment.

Setup and Installation

Follow these steps to set up the project environment:

1. Create the project directory and initialize Node.js:**

```
mkdir hardhat-counter
cd hardhat-counter
npm init -y
```

2. Install Hardhat and dependencies:

Due to recent changes in Hardhat's dependency tree, specific stable versions are used, and the `"--legacy-peer-deps` flag is required for a clean install.

```
npm install --save-dev hardhat@2.22.10
@nomicfoundation/hardhat-toolbox@5.0.0 --legacy-peer-deps

# Install remaining dependencies for the toolbox

npm install --save-dev "@nomicfoundation/hardhat-chai-matchers@^2.0.0"
"@nomicfoundation/hardhat-ethers@^3.0.0"
"@nomicfoundation/hardhat-ignition-ethers@^0.15.0"
"@nomicfoundation/hardhat-network-helpers@^1.0.0"
"@nomicfoundation/hardhat-verify@^2.0.0" "@typechain/ethers-v6@^0.5.0"
"@typechain/hardhat@^9.0.0" "@types/chai@^4.2.0" "@types/mocha@>=9.1.0"
"chai@^4.2.0" "ethers@^6.4.0" "hardhat-gas-reporter@^1.0.8"
"solidity-coverage@^0.8.1" "ts-node@>=8.0.0" "typechain@^8.3.0"
"typescript@>=4.5.0" "@nomicfoundation/hardhat-ignition@^0.15.16"
"@nomicfoundation/ignition-core@^0.15.15" --legacy-peer-deps
```

3. Create the Hardhat Configuration File (`hardhat.config.js`):**

This file uses the CommonJS format for maximum compatibility.

```
\``\``\`javascript
// hardhat.config.js
```

```
require("@nomicfoundation/hardhat-toolbox");

/** @type import('hardhat/config').HardhatUserConfig */
module.exports = {
  solidity: "0.8.24", // Use a stable Solidity version
};

```

```

4. Create the necessary directories:

```
mkdir - contracts
mkdir - test
mkdir - scripts
```

```
Contract Code (`contracts/Counter.sol`)
```

This is a simple Solidity contract that allows incrementing and decrementing a counter, with a check to prevent decrementing below zero.

```
\``\``solidity
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.24;

contract Counter {
 uint256 public count;

 event Incremented(uint256 newCount);
 event Decremented(uint256 newCount);

 function increment() public {
 count += 1;
 emit Incremented(count);
 }

 function decrement() public {
 require(count > 0, "Counter: cannot decrement below zero");
 count -= 1;
 emit Decremented(count);
 }

 function getCount() public view returns (uint256) {
 return count;
 }
}
```

```

}

```
## Test Code (`test/Counter.js`)

This JavaScript test file uses Hardhat's built-in environment, Ethers.js, and Chai for assertions. It tests all functionalities of the `Counter` contract, including deployment, state changes, error handling, and event emission.

```
```
javascript
const { expect } = require("chai");
const { ethers } = require("hardhat");

describe("Counter Contract", function () {
  let counter;

  beforeEach(async function () {
    // Deploy the contract before each test
    const Counter = await ethers.getContractFactory("Counter");
    counter = await Counter.deploy();
  });

  it("Should initialize with a count of 0", async function () {
    expect(await counter.getCount()).to.equal(0);
  });

  it("Should increment the count", async function () {
    await counter.increment();
    expect(await counter.getCount()).to.equal(1);
  });

  it("Should decrement the count", async function () {
    await counter.increment();
    await counter.decrement();
    expect(await counter.getCount()).to.equal(0);
  });

  it("Should fail when decrementing below zero", async function () {
    await expect(counter.decrement()).to.be.revertedWith("Counter: cannot decrement below zero");
  });

  it("Should emit an event on increment", async function () {
    await expect(counter.increment())
      .to.emit(counter, "Incremented")
      .withArgs(1);
  });
}
```

```

```
Running the Tests
```

To compile the tests

```
\```\`bash
npx hardhat compile
\``\`
```

```
Running the Tests
```

To run the tests and see the successful output:

```
\```\`bash
npx hardhat test
\``\`
```

```
Expected Output
```

The output confirms that all 5 tests passed successfully:

```
\``\``
Counter Contract
✓ Should initialize with a count of 0
✓ Should increment the count
✓ Should decrement the count
✓ Should fail when decrementing below zero
✓ Should emit an event on increment
5 passing (XXXms)
\``\`
```