

Fund Me

The screenshot displays the Remix Ethereum IDE interface. The top bar shows the browser address: `remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js`. The left sidebar contains a 'FILE EXPLORERS' panel with a 'default_workspace' and a list of files including `boolean.sol`, `stringexample.sol`, `sharedwallet.sol`, `ExtraStorage.sol`, `StorageFactory.sol`, and `FundMe.sol`. The main editor area shows the `FundMe.sol` contract code, which includes a `getVersion()` function, a `withdraw()` function, and a `fallback()` function. The code is written in Solidity and includes comments and imports.

```
function getVersion() public view returns (uint256){
    AggregatorV3Interface pricefeed = AggregatorV3Interface(0x8A753747A1Fa494EC906cE90E9F37563ABAF630e);
    return pricefeed.version();
}

modifier onlyOwner {
    // require(msg.sender == owner);
    if (msg.sender != i_owner) revert NotOwner();
    _;
}

function withdraw() payable onlyOwner public {
    for (uint256 funderIndex=0; funderIndex < funders.length; funderIndex++){
        address funder = funders[funderIndex];
        addressToAmountFunded[funder] = 0;
    }
    funders = new address[](0);
    // // transfer
    // payable(msg.sender).transfer(address(this).balance);
    // // send
    // bool sendSuccess = payable(msg.sender).send(address(this).balance);
    // require(sendSuccess, "Send failed");
    // call
    (bool callSuccess, ) = payable(msg.sender).call{value: address(this).balance}("");
    require(callSuccess, "call failed");
}

// Explainer from: https://solidity-by-example.org/fallback/
// Ether is sent to contract
// is msg.data empty?
// / \
// yes no
// / \
// receive()? fallback()
```

