

Tests fort insert currency

1. Screen shot of terminal output of tests

```

PROBLEMS OUTPUT DEBUG CONSOLE PORTS TERMINAL
PS C:\Users\matto\OneDrive\Cplusplus\project\week7> ./merkelrex
Tests for insert Currency
Start with an empty wallet
Print wallet:
    your wallet is empty

If currency 'type' is not in the wallet yet, set the amount to amount
Add 10 BTC and 10000 USDT
Print wallet:
BTC : 10.000000
USDT : 10000.000000

If currency 'type' is already in the wallet and has > 0, add amount to the current amount
Add some more currency, 5 BTC and 5000 USDT
Print wallet:
BTC : 15.000000
USDT : 15000.000000

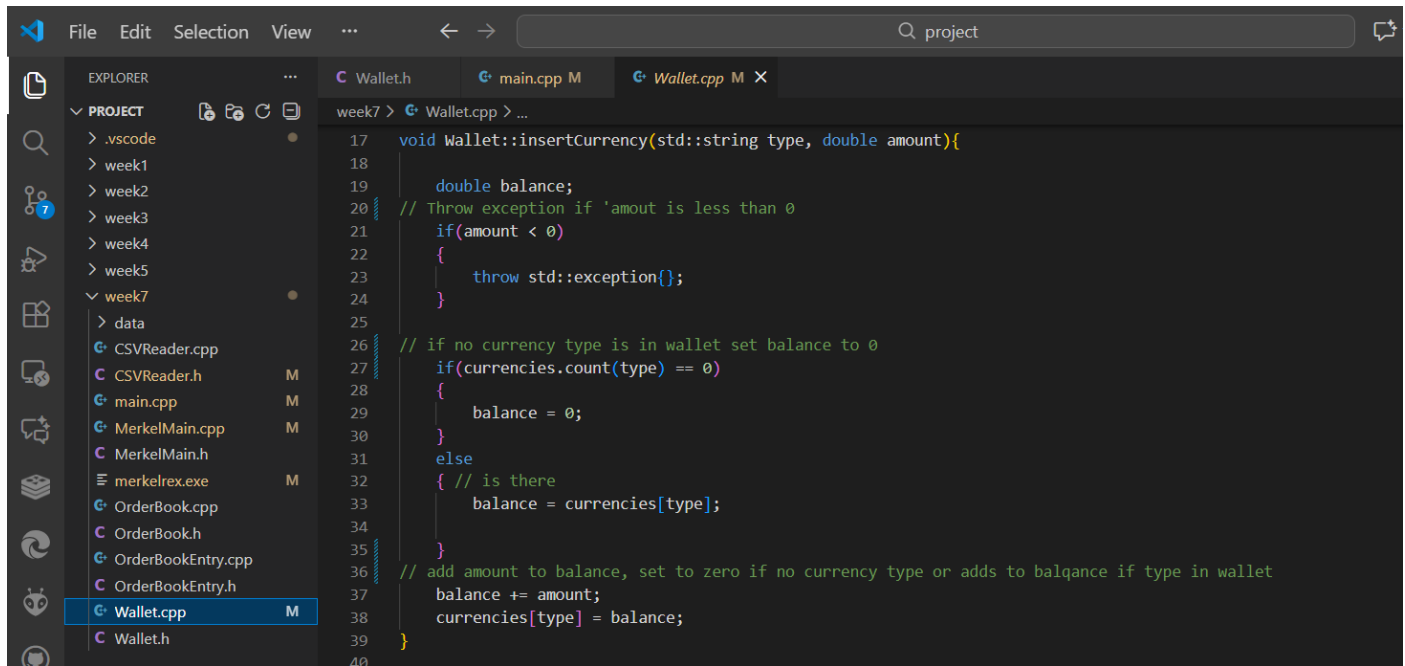
Throw an exception if 'amount' is less than 0
Lets try to add -10 ETH
terminate called after throwing an instance of 'std::exception'
    what():  std::exception
PS C:\Users\matto\OneDrive\Cplusplus\project\week7>

```

- ## 2. Test script in main() function

```
25 // Prototypes and declarations for functions below
26
27
28 std::string orderBookTypeToString(OrderBookType type);
29
30 int main()
31 {
32     // MerkelMain app;
33     //app.init();
34     //CSVReader::readCSV("crypto_dataset.csv");
35
36     Wallet wallet;
37
38     // Tests for insert Currency
39     std::cout << "Tests for insert Currency" << std::endl;
40     std::cout << "Start with an empty wallet" << std::endl;
41     std::cout << "Print wallet:" << std::endl;
42     std::cout << wallet.toString() << std::endl;
43     std::cout << "" << std::endl;
44     std::cout << "If currency 'type' is not in the wallet yet, set the amount to amount"<< std::endl;
45     std::cout << "Add 10 BTC and 10000 USDT " << std::endl;
46     wallet.insertCurrency("BTC", 10);
47     wallet.insertCurrency("USDT", 10000);
48     std::cout << "Print wallet:" << std::endl;
49     std::cout << wallet.toString() << std::endl;
50     std::cout << "If currency 'type' is already in the wallet and has > 0, add amount to the current amount" << std::endl;
51     std::cout << "Add some more currency, 5 BTC and 5000 USDT" << std::endl;
52     wallet.insertCurrency("BTC", 5);
53     wallet.insertCurrency("USDT", 5000);
54     std::cout << "Print wallet:" << std::endl;
55     std::cout << wallet.toString() << std::endl;
56     std::cout << "Throw an exception if 'amount' is less than 0" << std::endl;
57     std::cout << "Lets try to add -10 ETH" << std::endl;
58     wallet.insertCurrency("ETH", -10);
59
60 }
```

3. Screen shot of insertCurrency() function from Wallet class



The screenshot displays the Visual Studio Code editor interface. The Explorer panel on the left shows a project structure with folders 'week1' through 'week7' and various source files. The file 'Wallet.cpp' is selected and open in the editor. The editor shows the implementation of the `insertCurrency` function for the `Wallet` class. The function takes a `std::string type` and a `double amount` as parameters. It declares a `double balance;` and includes a comment: `// Throw exception if 'amount is less than 0`. It then checks `if(amount < 0)` and throws a `std::exception()` if true. Next, it checks `if(currencies.count(type) == 0)` to see if the currency type is already in the wallet. If not, it sets `balance = 0;`. Otherwise, it sets `balance = currencies[type];`. Finally, it adds the amount to the balance: `balance += amount;` and updates the wallet's currency list: `currencies[type] = balance;`. The function ends with a closing brace.

```
17 void Wallet::insertCurrency(std::string type, double amount){
18
19     double balance;
20     // Throw exception if 'amount is less than 0
21     if(amount < 0)
22     {
23         throw std::exception();
24     }
25
26     // if no currency type is in wallet set balance to 0
27     if(currencies.count(type) == 0)
28     {
29         balance = 0;
30     }
31     else
32     { // is there
33         balance = currencies[type];
34     }
35
36     // add amount to balance, set to zero if no currency type or adds to balqance if type in wallet
37     balance += amount;
38     currencies[type] = balance;
39 }
40
```