Test for Bank

TEST NAME	TEST DESC	EXPECTED	ACTUAL	COMMENTS
Test for **Bank**	Check registered Bank name	Bank name equal to the name entered	Same as expected	

Test for BankManager

Test for **BankManager**	Test **addBank(Name)** function with Bank name entered	**addBank()** be able to add the Bank into the BankList	Same as expected	
Test for **BankManager**	Test **getBank(Name)** function with Bank name entered	**getBank()** can return a list of all Bank	Same as expected	

Test for Stock

Test for **Stock**	Test **Stock()** function with the stockCode, price and volume entered	When new stock is created, **Stock()** can set the stockCode, price and volume of class Stock equal to that new stock.	Same as expected	

Test for StockManager

Test for **StockManager**	Test **addStock()** function	Created a new stock and if that stock does not exist, or already exist but have no price, **addStock()** wil be able to add the stock into the StockList	Same as expected	
Test for **StockManager**	Test **addStock()** function	Created a new stock and if that stock already exists with name and price, **addStock()** will add the volume to that stock.	Same as expected	

Test for **StockManager**	Test **getStock(stockCode)** function	**getStock()** can return a list of all stock which has the stockCode equal to the stockCode entered	Same as expected	

Test for Exchange

Test for **Exchange**	Create and check stock manager, bank manager and log manager is empty or not	BankManager, StockManager, LogManager exists	Same as expected	
Test for **Exchange**	Check it belong for their class or not	BankManager, StockManager, LogManager belong with their class	Same as expected	

Test for LogManager

Test for **addlog()** in logList	Compare with another log already have in log list	Return new logList	Same as expected	
Test for **getbanktransaction()** in logList	Test **getBankTransaction()**	List of bank transactions is returned	Same as expected	

Test for Log

Test for log	Compare new log in log list	**(String origin, String dest, String type, String stockCode, int volume, Decimal price, DateTime timeStamp)** matches input	Same as expected	

Test for WindowManager

Test for signup functionality

1. Bank access sign up screen
   Test method: open the program, input nothing at the login prompt
   Expected: Program displayed prompt for user to login
   Actual: Program displayed prompt for user to login
2. Bank can sign up with a username
   Test method: at sign up screen, input a username
   Expected: Program displayed signed up successfully
   Actual: Program displayed signed up successfully
3. Bank cannot sign up with an existing username
   Test method: use an already signed up username upon signing up
   Expected: Program displayed that bank already exist
   Actual: Program displayed that bank already exist
4. Bank can go back to login screen from signup screen
   Test method: input nothing at the signup screen
   Expected: Login screen is displayed
   Actual: Login screen is displayed
5. Bank can login using the new account
   Test method: login using the signed-up username
   Expected: login successfully
   Actual: login successfully

Test for login functionality

1. Bank can login
   Test method: login using an existing account
   Expected: login successfully
   Actual: login successfully
2. Ask for retry if wrong username is input
   Test method: try logging in using a new username that is not registered
   Expected: prompt that username does not match any registered bank and ask for login again
   Actual: prompt that username does not match any registered bank and ask for login again

Test for main menu functionality

1. Trade option working
   Test method: input 1 at the main menu screen
   Expected: trade menu is displayed
   Actual: trade menu is displayed
2. Your asset option working
   Test method: input 2 at the main menu screen
   Expected: Asset screen is displayed
   Actual: asset screen is displayed
3. Your invoice option working
   Test method: input 3 at the main menu screen
   Expected: invoice screen is displayed
   Actual: invoice screen is displayed
4. Transaction log option working
   Test method: input 4 at the main menu screen
   Expected: transaction log screen is displayed
   Actual: transaction log screen is displayed
5. Financial report option working
   Test method: input 5 at the main menu screen
   Expected: financial report screen is displayed
   Actual: financial report screen is displayed
6. Exchange inventory option working
   Test method: input 9 at the main menu screen
   Expected: inventory screen is displayed
   Actual: inventory screen is displayed
7. Exit option working
   Test method: input 0 at the main menu screen
   Expected: the program terminates and clear the screen
   Actual: the program terminates and clear the screen

Test for Trade Menu functionality

1. Back option working
   Test method: input 0 at trade menu screen
   Expected: main menu screen is displayed
   Actual: main menu screen is displayed
2. Stock code displaying best prices
   Test method: insert different prices for the same stock inside jsonsaved.json file
   Expected: only the lowest price is displayed
   Actual: only the lowest price is displayed
3. Only unique stock code is displayed
   Test method: insert in multiple different stock codes inside jsonsaved.json along with existing stock codes
   Expected: no same stock code is displayed
   Actual: no same stock code is displayed
4. User can choose a stock code
   Test method: input in a stock code displayed in the menu
   Expected: trade screen for the chosen stock is displayed
   Actual: trade screen for the chosen stock is displayed
5. User chose a non-existing stock
   Test method: choose a stock not displayed in the menu
   Expected: user prompted to choose from the listed option
   Actual: user prompted to choose from the listed option

Test for Buy Menu

1. Buy option working
   Test method: input 1 at buy menu
   Expected: program prompt user to enter a price
   Actual: program prompt user to enter price
2. 10 best prices is displayed in ascending order
   Test method: insert in different prices for the same stock inside jsonsaved.json file
   Expected: 10 lowest price is displayed in an ascending order
   Actual: 10 lowest price is displayed in an ascending order
3. User can buy
   Test method: input a suitable price and volume
   Expected: program promted for successful transaction
   Actual: program promted for successful transaction
4. Quote option working
   Test method: input 2 at the buy menu
   Expected: program prompt user to enter a price and return a quote accordingly
   Actual: program prompt user to enter a price and return a quote accordingly
5. Back option working
   Test method: input 0 at buy menu
   Expected: trade menu is displayed
   Actual: trade menu is displayed

Test for Assets screen

1. Assets are displayed correctly
   Test method: insert stock for the bank inside jsonsaved.json
   Expected: the inserted stock is displayed
   Actual: the inserted stock is displayed

Test for Invoice screen

1. All stock purchases are displayed and fees are calculated
   Test method: use the buy menu to buy stock multiple times
   Expected: the invoice screen shows all successful buys along with fees that matches the calculations
   Actual: the invoice screen shows all successful buys along with fees that matches the calculations

Test for Transaction Logs

1. All logs are saved and displayed
   Test method: try to buy and quote stock using the trade menu
   Expected: the transaction logs show all logs that corresponds to what was bought and quoted
   Actual: the transaction logs show all logs that corresponds to what was bought and quoted

Test for Exchange Report

1. All banks are displayed and values are displayed according the saved data
   Test method: login using different banks and buy multiple stocks
   Expected: all banks are displayed and credit score is displayed according to what was purchased, fees are calculated accordingly
   Actual: all banks are displayed and credit score is displayed according to what was purchased, fees are calculated accordingly

Test for Exchange inventory

1. All stocks inside the exchange are displayed
   Test method: insert stock data inside jsonsaved.json
   Expected: the data is displayed
   Actual: the data is displayed