You need to implement 00P structure (basic classes and methods) to represent customer ' s e - wallet functionality .

- The system will consist of the customers with their own unique customer IDs and their e - wallet .
- Each e - wallet has at least 2 accounts - 1 with virtual currency (credits , 1 credit = 1 USD) and 1 with USD account .
- The user can set any of account except virtual one as a default account .
- The user can freeze any of account except the virtual one .
- The user can add more accounts in any of the currency .
- It should be possible to list all of the accounts of the user and get a balance of any account .
- The user can top up any account with any amount which is not higher than customer ' s top up limit (per day) .
- The user can transfer amounts from one account to another .
- The user can withdraw money from any account not exceeding customers withdrawal limit (per day) .
- It is not possible to do money withdrawal from virtual currency account .

- You need to implement the business logic of each method described above .
- You don ' t need to implement data storage , except the log file .

- Each action of the customer should be logged into the file , each log record should mention the time and name of the action executed .

- Your code should be clean , independent , easy to understand and well - documented It will be run from command - line . Based on it , the person verifying your task will write some sample scenario and verify the log file .

Example of scenario:

1. Create new customer with id test customer _ 12
2. Get list of accounts of the user

3. Set a daily top up limit to 100 USD .
4. Top up USD account with 113 USD
5. Top up USD account with 57 USD
6. Add EUR account
7. Transfer 40 USD to EUR account .
8. Get balance of the EUR account .
9. Freeze EUR account .

Etc