CASH DESK MODULE

Description:

You are developing a cash operations module in an internal information system. Cashiers should be able to deposit and withdraw money in BGN and EUR. After 2 withdrawals and 2 deposits, the cashier should be able to check his balance and denominations. The cashier starts the day with set amount in BGN and EUR in the cash desk. All cash operations will be executed in the same working day. The user name of the cashier within the system is MARTINA.

Starting Amounts:

1000 BGN, denomination: 50x10, 10x50

2000 EUR, denomination: 100x10, 20x50

Cash Operations:

Withdrawal 1: 100 BGN, denomination: 5x10 BGN, 1x50 BGN

Withdrawal 2: 500 EUR, denomination:10x50 EUR

Deposit 1: 600 BGN, denomination: 10x10 BGN, 10x50 BGN

Deposit 2: 200 EUR, denomination: 5x20 EUR, 2x50 EUR

Requirements & Outputs:

- **Answer Deadline:** 96 hours after receiving the email with the assignment
- **Answer Format:** Link to GitHub repository
- **Answer Recipient:** Send the answer formatted as stated above before the deadline to the following recipients: daniel.baykov@fibank.bg and bilyana.a.ivanova@fibank.bg
- **Clarifications will not be provided. Conduct the development based solely on this document.**
- **Output: After cloning the project from the repository, building it, running it locally and executing the authenticated 5 Postman requests in the order provided, the output of the last request must have the right balances and denominations for BGN and EUR for the cashier.**
- Use GitHub for project storage
- Use Java 8
- Use Maven
- Use Spring Boot
- Use validation for the API requests
- Use JSON format for requests and responses
- User jar as packaging format
- Use embedded Tomcat for local running of the application
- Use a custom request header named FIB-X-AUTH with API key = f9Uie8nNf112hx8s for all API calls
- Create controllers for cash operations as well as balance check:
    - /api/v1/cash-operation. Deposits and withdrawals must be part of one and the same API method
    - /api/v1/cash-balance. Balance and denominations are returned from one and the same API method
- Store the API key in a suitable place and suitable format within the spring project
- Create a Postman collection with individual requests (total of 2+2+1) for the cash operations and balance check. Add the postman collection (and environment, if any) within the git project. Order the requests for execution numerically.
- Use separate TXT file for transaction history, Find the most simple, fast and reliable way to format and structure the file. Update the file with each API method call.
- Use separate TXT file for cash balances and denominations. Find the most simple, fast and reliable way to format and structure the file. Update the file with each API method call.