**Reference Material – Folio3 Senior QA / Lead QA Hiring Test**

*Please refer to the below question for solving Q. No. 9 to 18 in the online hiring test.*

Given below is the function that validates the Card Number, Customer Name, Pin Code, Maximum Amount Limit, Available Balance and Transaction Amount for a transaction on an E-Commerce Web Site.

You are required to ensure the effectiveness of this code. Verify the outputs against the provided input parameters to ensure all possible work flows are working correctly.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **S.No** | **cardNo** | **UserName** | **Pin** | **Transaction** **Amount** | **Available****Balance** | **Output** | **Reason** |
| 9 | 16 | Saba | 1240 | 8000 |  |  |  |
| 10 | 17 | Ali | 1241 | 20000 |  |  |  |

**int makeTransaction (unsigned int cardNo, string userName, int pin, int transactionAmount)
{
    //date is a dd/mm/yyyy
    if ( cardNo < 10 )
    return 0;

    if ( userName == NULL )
    return 0;**

**if ( pin < 1000 )
    return 0;**

**//The following function returns all the account information for this given card number.
    accountInformation = getAccountInfo (cardNo);

    if (accountInformation == NULL)
    {
        return 0;
    }
    else
    {
        if((accountInformation.userName == userName) && (accountInformation.availableBalance >= transactionAmount)
            && (accountInforma.maxlimit >= transactionAmount) && (accountInformation.pin == pin))
            {
                // The following function deduct amount from user account
                deduce (cardNo, transactionAmount);

                cout << accountInformation.availableBalance << endl;
                return 1;
            }
        else
            return 0;
    }
}**

**Dataset for the program: (Get accountInformation for card#)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **cardNo** | **userName** | **Pin** | **maxLimit** | **Available Balance** |
| 06 | Kashif | 1234 | 25000 | 50000 |
| 50 |  | 1235 | 20000 | 15000 |
| 12 | Samad | 0948 | 5000 | 4000 |
| 13 | Nasir | 1237 | 6000 | 10000 |
| 14 | Ijaz | 1238 | 15000 | 5000 |
| 15 | Owais | 1239 | 4000 | 5000 |
| 16 | Saba | 1240 | 8000 | 10000 |
| 17 | Ali | 1241 | 20000 | 20000 |

**Input Parameters:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **S.No** | **cardNo** | **UserName** | **Pin** | **Transaction** **Amount** | **Available****Balance** | **Output** | **Reason** |
| 1 | 06 | Kashif | 1234 | 2000 |  | 0 | Card # < 10 |
| 2 | 50 |  | 1235 | 3000 |  | 0 | **userName == NULL** |
| 3 | 12 | Samad | 0948 | 4000 |  | 0 | **pin < 1000** |
| 4 | 20 | Nasir | 1236 | 5000 |  |  | **accountInformation == NULL** |
| 5 | 13 | Owais | 1237 | 1000 |  | 0 | **accountInformation.userName == username Does not match** |
| 6 | 14 | Ijaz | 1238 | 3000 | 2000 | 1 |  |
| 7 | 15 | Owais | 1239 | 5000 | 0 | 1 |  |
| 8 | 16 | Saba | 1242 | 6000 |  | 1 | Pin |
| 9 | 16 | Saba | 1240 | 8000 | 2000 | 1 |  |
| 10 | 17 | Ali | 1241 | 20000 | 0 | 1 |  |