

Date: 16/09/2020

Course: TY BCA

Enrollment No.: 201803100110002

Subject: 030010517 CC12 Software Testing Techniques

Practical-2**➤ Refer the code and do the following:**

- Add deposit and withdrawal dates.
- Date format must be dd-mm-yyyy.
- Bank account number must be in the format 20200001 and must be between 20200001 to 20209999.
- Amount deposited must be in multiples of 500.
- Develop test cases for Dates, Acno, Deposit, Withdrawal, Overdraw, balance.
- Each test case must focus on parameters like length of input, datatype, business logic etc.
- For the above design regular expressions.
- Change the code in such a way that assertions can be applied.

➤ Code:

```
import java.util.Scanner;
```

```
import java.text.SimpleDateFormat;
```

```
import java.util.Date;
```

```
import java.util.regex.*;
```

```
class BankInternal {
```

```
    int account_no;
```

```
    float balance = 0;
```

```
    Date depositDate, withdrawDate;
```

```
    Scanner sc = new Scanner(System.in);
```

```
bankInternal()
{
    System.out.println("Enter Account Number ");
    while(true){
        account_no = sc.nextInt();
        if(Pattern.matches("^(2020)[0-9]{3}[1-9]{1}$", account_no))
            break;
        else
            System.out.println("Enter 8 digit in length account no.: ");
    }

    System.out.println("Enter Initial balance ");
    balance = sc.nextFloat();
}

void deposit()
{
    float amount;
    String depositDate;

    System.out.println("Enter Amount to be Deposited ");
    while(true){
        amount = sc.nextFloat();
        if(amount % 500 == 0)
            break;
```

```
        else

            System.out.println("Enter amount which is multiple of 500: ");

    }

    System.out.println("Enter Date of Deposited (dd/mm/yyyy): ");


    while(true){

        depositDate = sc.nextLine();

        if(Pattern.matches("^(1[0-2]|0[1-9])/(3[01]|[12][0-9]|0[1-9])/[0-9]{4}$", depositDate))

            break;

        else

            System.out.println("Enter date in given format.");

    }


    dDate = new SimpleDateFormat("dd/MM/yyyy").parse(depositDate);

    balance = balance + amount;

    System.out.println("Deposited! Account balance is " + balance);

}


void withdraw()

{

    float amount;

    String withdrawDateStr;
```

```
System.out.println("Enter Amount to be Withdrawn ");

amount = sc.nextFloat();

System.out.println("Enter Date of Withdrawn (dd/mm/yyyy) ");

while(true){

    withdrawDateStr = sc.nextLine();

    if(Pattern.matches("(^(1[0-2]|0[1-9])/(3[01]| [12][0-9]|0[1-9])/[0-9]{4}$",
withdrawDateStr))

        break;

    else

        System.out.println("Incorrect date format. Enter again.");

}

withdrawDate = new SimpleDateFormat("dd/MM/yyyy").parse(withdrawDateStr);

if(amount<balance)

{

    balance = balance - amount;

    System.out.println("Amount Withdrawn! Available balance: " + balance);

}

else

{

    balance = balance - amount;

    System.out.println("Overdraws!!!");
```

```

    }
}
}

```

```

public class Bank{

    public static void main(String[] args)

    {

        bankInternal myObj = new bankInternal();


        myObj.deposit();

        myObj.withdraw();

    }

}

```

➤ **Test Case:**

Input Variable	Length			Datatype			Range/Criteria			Special Validation			
	Invalid	Valid	Invalid	Valid	Invalid		Invalid	Valid	Invalid	Valid	Invalid	Valid	Invalid
Account Number	<8	=8	>8	numeric/integer	string, float, boolean		<20200001	inclusive 20200001-20209999	null, >20209999, others	prefix 2020	other prefix		other prefix
Deposits			3	numeric/integer				500, 1000, 1500, 2000					
Withdraw	<3	>=3<=5	>5	numeric/integer	string, float, boolean		<100	inclusive 100-25,000	>25,000	N/A	N/A		N/A
Date	<10	=10	>10	string	numeric/integer/float, boolean	Date	<1	inclusive 1-31	>31	28 When month is February	>28	29 When month is February and it is leap year	>29
						Month	<1	inclusive 1-12	>12	N/A	N/A	N/A	N/A
						Year (assumption: range begin from 1999)	<1999	inclusive 1999- 2020	>2020	N/A	N/A	N/A	N/A
Overdraw				numeric/integer									
Balance			3	numeric/integer									

➤ **Enhance your learning:**

1. Find and describe how bank account numbers numbered in India are generated? Consider any Bank of your choice and list all the validation criteria for numbering. Prepare test case table with actual input values for each validation criteria.

Solution: Finacle generates account numbers in a serial format, including various details, such as branch code, scheme code, serial number, etc. When you open an account with Indian Bank, you will be issued an account number with 9 digits. As Indian Bank uses BaNCS core banking, account numbers are issued on a random basis.

- Bank: SBI Bank
 - Validation Criteria:
 - Mostly SBI uses eleven numeric (11) digit account no at every where to open the Saving, Current, FD/RD, Loan, KCC and CC/OD accounts. Even uses eleven digit in CBS to access the customers transactions.
 - SBI has 17 digit complete account number but SBI uses 11 digit in CBS. The six digit are zero before eleven digit account number. We can see the complete SBI Account number is 0000001111111111 where 0 is starting six digit and 1 is 11 digit account number.

○ Test Case:

	Length			Datatype			Range/Criteria			Special Validation			
Input Variable	Invalid	Valid	Invalid	Valid	Invalid		Invalid	Valid	Invalid	Valid	Invalid	Valid	Invalid
Account Number	<8	=8	>8	numeric/integer	string, float, boolean		<20200001	inclusive 20200001-20209999	null, >20209999, others	prefix 2020	other prefix		other prefix
Deposits			3	numeric/integer				500, 1000, 1500, 2000					
Withdraw	<3	>3<=5	>5	numeric/integer	string, float, boolean		<100	inclusive 100-25,000	>25,000	N/A	N/A		N/A
Date	<10	=10	>10	string	numeric/integer float, boolean	Date	<1	inclusive 1-31	>31	28 When month is February	>28	29 When month is February and it is leap year	>29
						Month	<1	inclusive 1-12	>12	N/A	N/A	N/A	N/A
						Year (assumption: range begin from 1999)	<1999	inclusive 1999-2020	>2020	N/A	N/A	N/A	N/A
Overdraw				numeric/integer									
Balance			3	numeric/integer									

2. Visit any banking site. Enter the customer id and check the virtual keyboard online to enter the password. How is the keyboard different from the laptop/desktop keyboard? Write at least two differences observed on that keyboard in four to five sentences each.

Solution:

- Virtual keypad is an online application, which substitutes the actual physical keyboard with a mouse.

- When you click on the virtual keyboard option at the time of net banking, the monitor flashes a keyboard on your screen.
- You have to use the mouse to click on the relevant keys to sign into your net banking ID.
- Banks have come with what's called 'Virtual Keyboard' to protect your account or card information and password (IPINs) from such fraudsters' hands.
- Once you enter your log in ID, you have to choose the option of virtual keyboard by clicking on the appropriate option.
- Then you can use the mouse to key in your password.