dab - 5

Develop a Java program to create a class Bank that maintains 2 kinds of account for its customer.

```java
import java.util.scanner;

class Account
{
    string customerName;
    long accountNumber;
    string accountType;
    double balance;

    public Account (String customerName,
            long accountNumber, String accountType,
                    double balance)
    {
        this.customerName = customerName.
        this.accountNumber = accountNumber.
        this.accountType = accountType;
        this.balance = balance;
    }
    public void deposit (double amount)
    {
        balance + = amount;
        System.out.println ("Deposit
                successful. updated balance:"+
                        balance);

    }
```

```java
public void displayBalance()
{
    System.out.println("Deposit Successful.
                        updated balance:" +bala
}

{
    System.out.println("Account Balance:" +bala
}

}

class currAcct extends Account
{
    double minBalance;
    double service charge;

    public currAcct (String customer Name,
                     long accountNumber, doubl

    super(customer Name, accountNumber, "current", balan

    this.minBalance = 100;
    this.servicecharge = 50;

}
```

```java
private void checkMinBalance()
{
    if (balance < minBalance)
    {
        balance -= servicecharge;
        system.out.println("service charges
                            imposed. updated
                            balance: " + balance);
    }
}

public void withdraw (double amount)
{
    if (amount <= balance)
    {
        balance -= amount;
        system.out.println("withdrawal
                            successful. updated
                            balance: " + balance);
        checkMinBalance();
    }
    else
    {
        system.out.println("Insufficient
                            fund for withdrawal.");
    }
}
}
```

```java
class SavAcc extends Account
{
    double interestRate;

    public SavAcc(String customerName,
                  long accountnumber, double balance)
    {
        super(customerName, accountNumber,
                        "saving", balance);
        this.interestRate = 0.05;
    }

    public void depositInterest()
    {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("interest deposited
                        updated balance:" + balance);
    }

    public void withdraw(double amount)
    {
        if(amount <= balance)
        {
            balance -= amount;
            System.out.println("withdrawal success
                        updated balance:" + balance);
        }
        else
        {
            System.out.println("Insufficient fund
                        for withdrawal");
        }
    }
}
```

```java
public class Bank
{
    public static void main (String[] args)
    {
        Scanner scanner = new Scanner (System.in);

        CurrAcct currentAccount = new CurrAcct
                   ("John Doe", 1234 , 1500 );
        currentAccount. deposit(500);
        currentAccount. displayBalance();
        currentAccount. withdraw (1000);


        SavAcct SavingsAccount = new SavAcct

                        ("John Doe", 9345, 2000);
        SavingsAccount. deposit(1000);
        SavingsAccount. displaybalance();
        SavingsAccount. depositInterest();
        SavingsAccount. displayBalance();
        SavingsAccount. withdraw (500);


        scanner.close();
    }
}
```

output:

Deposit successful
update balance : 2000

account Balance : 2000
withdrawal succesful

update balance = 1000

Deposit successful

9/1/24