Account Class uml Diagram

|  |
| --- |
| ACCOUNT |
| * id: int * balance: double * annualInterestRate: double * dateCreated: Date |
| + Account()  + Account(currentId: int , currentBalance: double)  + getId(): int  + getBalance(): double  + getMonthlylInterestRate(): double  + getMonthlyInterest(): double  + getDateCreated(): Date  + setId(id: int) : void  + setBalance(balance: double) : void  + setAnnualInterestRate(annualInterestRate: double) : void  + withDraw(amountToWithdraw: double) : void  + deposit(amountToBeDeposited: double) : void |

* Private

+ Public

|  |
| --- |
| myAccount: Account (Object) |
| Id=1122  Balance = 20,000 |

import java.util.Date;

public class Account {

private int id = 0; // declaration of data field id of type int with default value 0;

// declaration of data field balance of type double with default value 0.0

private double balance = 0.0;

// declaration of data field annual interest rate of type double with default value 0.0;

private double annualInterestRate = 0.0;

// declaration of data field dateCreated of type Date;

private Date dateCreated ;

// no argument constructor

public Account(){

}

// constructor with two arguments of type int and double

public Account (int currentId, double newBalance) {

id = currentId;

balance = newBalance;

}

// method getId that returns the current account id

public int getId() {

return id;

}

// method setId that sets a new id

public void setId (int newId) {

id = newId;

}

// method getBalance that returns the current balance

public double getBalance() {

return balance;

}

// method that sets a new balance

public void setBalance(double newBalance) {

balance = balance;

}

// method getAnnualInterestRate that returns the annual interest rate

public double getAnnualInterestRate () {

return annualInterestRate;

}

//method that sets the annual interest rate

public void setAnnualInterestRate (double newAnnualInterestRate) {

annualInterestRate = newAnnualInterestRate;

}

// method that returns the current date

public Date getDateCreated () {

return dateCreated;

}

// method that calculate and returns the montyly interest rate (annualInterestRate/100)/12

public double getMonthlyInterestRate () {

return annualInterestRate / 1200.0 ;

}

// method that calculate and returns the montlylyInterest

public double getMonthlyInterest() {

double monthlyInterestRate = getMonthlyInterestRate();

return balance \* monthlyInterestRate;

}

/\*method that check if account has enough money then withdraw

\* if true else it tells the user he/she doesn't have enough money

\*/

public void withdraw (double amountToWithdraw) {

if (balance >= amountToWithdraw)

balance -= amountToWithdraw;

else

System.out.println ("Sorry, but the requested amount is greater than your available balance!!!");

}

// method to deposit money to your account

public void deposit (double amountToBeDeposited) {

balance += amountToBeDeposited;

}

}

// TestAccount class that test the Account class

public class TestAccount {

public static void main (String[] args) {

// create Account object called myTestAccount with two arguments of type int and double

Account myTestAccount = new Account (1122, 20000);

// set the annual interest rate to 4.5 using instance variable myTestAccount and calling the //setAnnualInterestRate() method

myTestAccount.setAnnualInterestRate (4.5);

// withdraw 2500 using instance variable myTestAccount and calling the withdraw method

myTestAccount.Withdraw(2500);

// deposit 3000 using instance variable myTestAccount and calling the deposit method

myTestAccount.deposit (3000);

// using the print format t0 print the account balance by calling the getBalance method

System.out.printf ("%s$%, 5.2f","The account balance is: ", myTestAccount.getBalance () );

// print a new line to make it more readable

System.out.println ();

// print the monthly interest is by calling the getMonthlyInterest() method

System.out.println ("The monthly interest is: $"+myTestAccount.getMonthlyInterest() );

// print the date the account was created by calling the getDateCreated() method;

System.out.println("This account has been created on "+ myTestAccount.getDateCreated());

}

}