LAB 2 Expressions andFlowControl

# Exercise 1: Using Loops and Branching Statements

## Task 1 – Creating the FooBarBaz Class

public class FooBarBaz {

    public static void main(String[] args) {

        for(int i = 1; i <= 50; i++ ){

            if(i % 3 == 0 )

                System.out.println(i + " Foo");

            else if(i % 5 == 0)

                System.out.println(i + " Bar");

            else if(i % 7 == 0)

                System.out.println(i + " Baz");

            else

                System.out.println(i);

        }

    }

}

## Task 2 – Compiling the FooBarBaz Class

Texto

Descripción generada automáticamente con confianza media

## Task 3 – Running the FooBarBaz Program

Texto

Descripción generada automáticamente

# Exercise 2: Using Conditional Statements in the Account Class

## Task 1 – Modifying the Account Class

package com.mybank.domain;

public class Account {

    private double Balance;

    public Account(double Balance) {

        this.Balance = Balance;

    }

    public Boolean deposit(double amt) {

        Balance = Balance + amt;

        return true;

    }

    public Boolean withdraw(double amt) {

        if (amt <= Balance) {

            Balance = Balance - amt;

            return true;

        }

        return false;

    }

    public double getBalance() {

        return Balance;

    }

}

## Task 2 – Deleting the Current TestBanking Class

Captura de pantalla de un celular

Descripción generada automáticamente

## Task 3 – Copying the TestBanking Class

package com.mybank.test;

import com.mybank.domain.\*;

/\*

 \* This class creates the program to test the banking classes.

 \* It creates a new Customer and Account (with an initial balance),

 \* and performs a series of transactions with the Account object.

 \*/

public class TestBanking {

  public static void main(String[] args) {

    Customer customer;

    Account  account;

    boolean operationResult;

    // Create an account that can has a 500.00 balance and type savings.

    System.out.println("Creating the customer Jane Smith.");

    customer = new Customer("Jane", "Smith");

    System.out.println("Creating her account with a 500.00 balance.");

    customer.setAccount(new Account(500.00));

    account = customer.getAccount();

    operationResult = account.withdraw(150.00);

    System.out.println("Withdraw 150.00: " + operationResult);

    operationResult = account.deposit(22.50);

    System.out.println("Deposit 22.50: " + operationResult);

    operationResult = account.withdraw(47.62);

    System.out.println("Withdraw 47.62: " + operationResult);

    operationResult = account.withdraw(400.00);

    System.out.println("Withdraw 400.00: " + operationResult);

    // Print out the final account balance

    System.out.println("Customer [" + customer.getLastName()

           + ", " + customer.getFirstName()

           + "] has a balance of " + account.getBalance());

  }

}

## Task 4 – Compiling the TestBanking Class

Texto

Descripción generada automáticamente

## Task 5 – Running the TestBanking Program

Texto

Descripción generada automáticamente

# Exercise 3: Using Nested Loops

## Task 1 – Writing the isSubString Method

public class TestIsSubString {

    /\*

     \* Write a isSubString method that searches for a specific string within

     \* another string; the method must return true if the former exists in the

     \* latter string. Otherwise, the method return false.

     \*/

    public static Boolean isSubString(String cadena1, String cadena2){

        return cadena2.indexOf(cadena1) != -1;

    }

    public static void main(String[] args) {

        String text = "The cat in the hat.";

        System.out.println("isSubString(\"cat\", \"The cat in the hat.\") "

                + isSubString("cat", text));

        System.out.println("isSubString(\"bat\", \"The cat in the hat.\") "

                + isSubString("bat", text));

        System.out.println("isSubString(\"The\", \"The cat in the hat.\") "

                + isSubString("The", text));

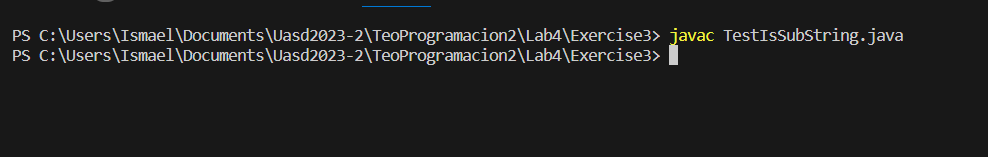
        System.out.println("isSubString(\"hat.\", \"The cat in the hat.\") "

                + isSubString("hat.", text));

    }

}

## Task 2 – Compiling the TestIsSubString Class



## Task 3 – Running the TestIsSubString Program

Texto

Descripción generada automáticamente