

LAB-5

By-Arun Lal

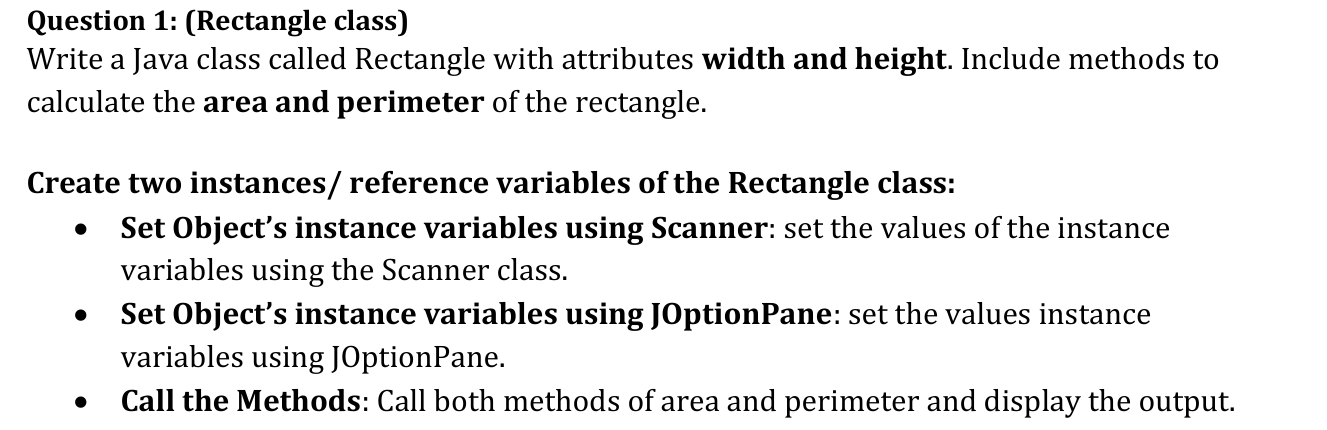


Sec-E

BSCS(II)

CMS-ID=023-24-0120

Exercise of Lab: 5

  
Source code:

import java.util.Scanner;

import javax.swing.JOptionPane;

class Rectangle {

double width;

double height;

public double calculateArea() {

return width \* height;

}

public double calculatePerimeter() {

return 2 \* (width + height);

}

}

public class RectangleDemo {

public static void main(String[] args) {

Rectangle rectangle1 = new Rectangle();

Rectangle rectangle2 = new Rectangle();

Scanner scanner = new Scanner(System.in);

System.out.println("Enter width for rectangle 1:");

rectangle1.width = scanner.nextDouble();

System.out.println("Enter height for rectangle 1:");

rectangle1.height = scanner.nextDouble();

System.out.println("Rectangle 1 Area: " + rectangle1.calculateArea());

System.out.println("Rectangle 1 Perimeter: " + rectangle1.calculatePerimeter());

rectangle2.width = Double.parseDouble(JOptionPane.showInputDialog("Enter width for rectangle 2:"));

rectangle2.height = Double.parseDouble(JOptionPane.showInputDialog("Enter height for rectangle 2:"));

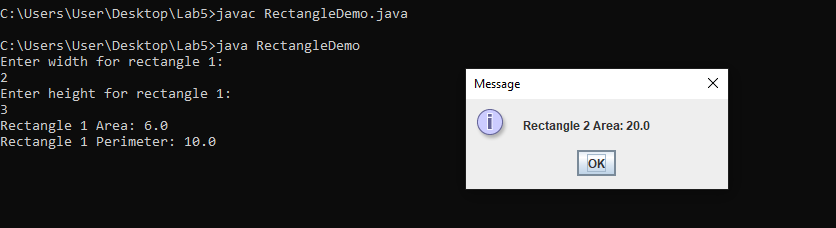
JOptionPane.showMessageDialog(null, "Rectangle 2 Area: " + rectangle2.calculateArea());

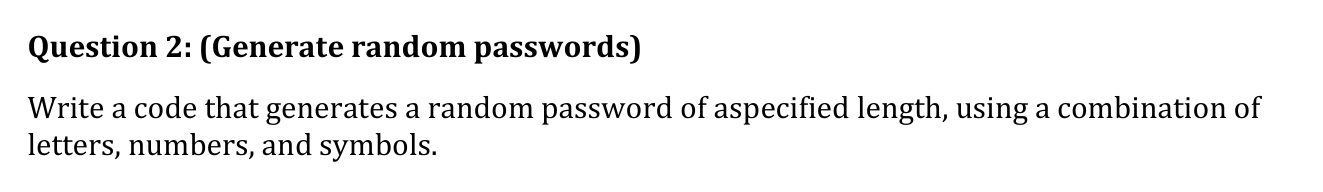
JOptionPane.showMessageDialog(null, "Rectangle 2 Perimeter: " + rectangle2.calculatePerimeter());

}

}

Output:





Source code:

import java.util.Random;

public class PasswordGenerator {

public static void main(String[] args) {

int length = 12;

String password = generateRandomPassword(length);

System.out.println("Generated Password: " + password);

}

public static String generateRandomPassword(int length) {

String upperCase = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

String lowerCase = "abcdefghijklmnopqrstuvwxyz";

String numbers = "0123456789";

String symbols = "!@#$%^&\*()-\_+=<>?";

String allChars = upperCase + lowerCase + numbers + symbols;

Random random = new Random();

StringBuilder password = new StringBuilder();

for (int i = 0; i < length; i++) {

int randomIndex = random.nextInt(allChars.length());

password.append(allChars.charAt(randomIndex));

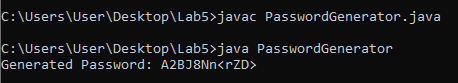
}

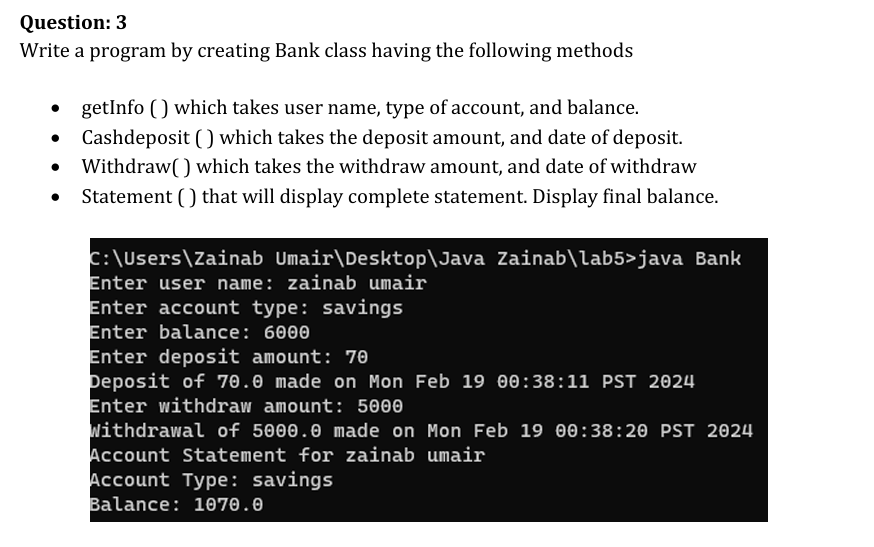
return password.toString();

}

}

Output:





Source code:

class Bank {

String userName;

String accountType;

double balance;

public void getInfo(String userName, String accountType, double balance) {

this.userName = userName;

this.accountType = accountType;

this.balance = balance;

}

public void cashDeposit(double amount, String date) {

balance += amount;

System.out.println("Deposited " + amount + " on " + date);

}

public void withdraw(double amount, String date) {

if (balance >= amount) {

balance -= amount;

System.out.println("Withdrawn " + amount + " on " + date);

} else {

System.out.println("Insufficient balance.");

}

}

public void statement() {

System.out.println("Name: " + userName);

System.out.println("Account Type: " + accountType);

System.out.println("Final Balance: " + balance);

}

}

public class BankDemo {

public static void main(String[] args) {

Bank bank = new Bank();

bank.getInfo("John Doe", "Savings", 1000);

bank.cashDeposit(500, "2023-10-27");

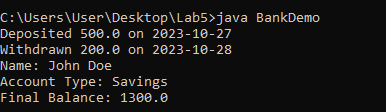
bank.withdraw(200, "2023-10-28");

bank.statement();

}

}

Output:



Source code:

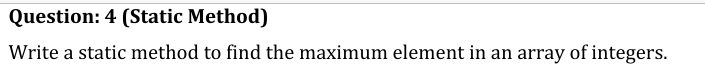
public class MaxFinder {

public static void main(String[] args) {

int[] numbers = { 5, 12, 8, 20, 3 };

System.out.println("Max element: " + findMax(numbers));

}



public static int findMax(int[] arr) {

if (arr == null || arr.length == 0) {

return Integer.MIN\_VALUE;

}

int max = arr[0];

for (int i = 1; i < arr.length; i++) {

if (arr[i] > max) {

max = arr[i];

}

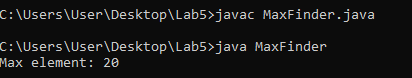
}

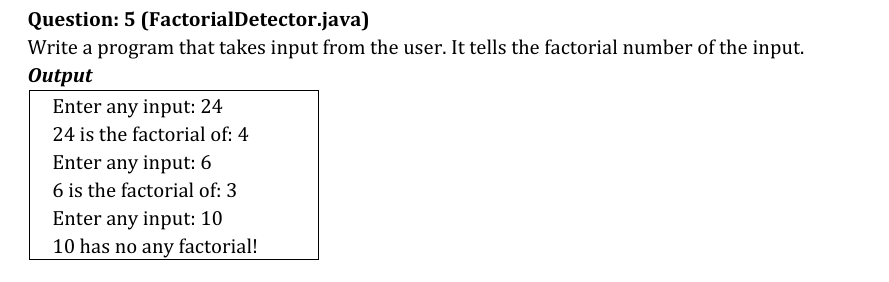
return max;

}

}

Output:





Source code:

import java.util.Scanner;

public class FactorialDetector {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter any input:");

int input = scanner.nextInt();

int factorial = 1;

int i = 1;

while (factorial < input) {

i++;

factorial \*= i;

}

if (factorial == input) {

System.out.println(input + " is the factorial of: " + i);

} else {

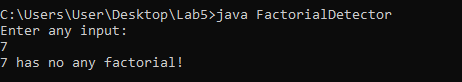
System.out.println(input + " has no any factorial!");

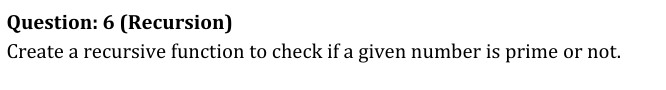
}

}

}

Output:





Source code:

public class PrimeChecker {

public static void main(String[] args) {

int numberToCheck = 17;

if (numberToCheck > 1 && isPrime(numberToCheck, 2)) {

System.out.println(numberToCheck + " is prime.");

} else {

System.out.println(numberToCheck + " is not prime.");

}

}

public static boolean isPrime(int n, int i) {

if (n <= 2) {

return (n == 2);

}

if (n % i == 0) {

return false;

}

if (i \* i > n) {

return true;

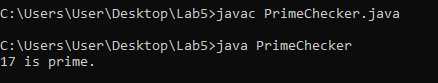
}

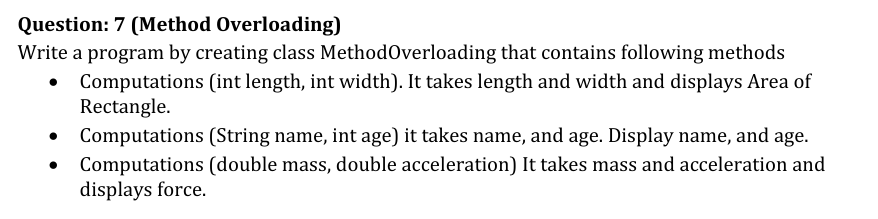
return isPrime(n, i + 1);

}

}

Output:





Source code:

class MethodOverloading {

public void computations(int length, int width) {

System.out.println("Area of Rectangle: " + (length \* width));

}

public void computations(String name, int age) {

System.out.println("Name: " + name + ", Age: " + age);

}

public void computations(double mass, double acceleration) {

System.out.println("Force: " + (mass \* acceleration));

}

}

public class MethodOverloadingDemo {

public static void main(String[] args) {

MethodOverloading mo = new MethodOverloading();

mo.computations(5, 10);

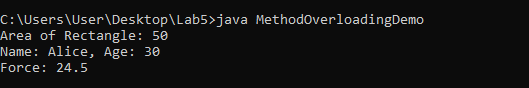
mo.computations("Alice", 30);

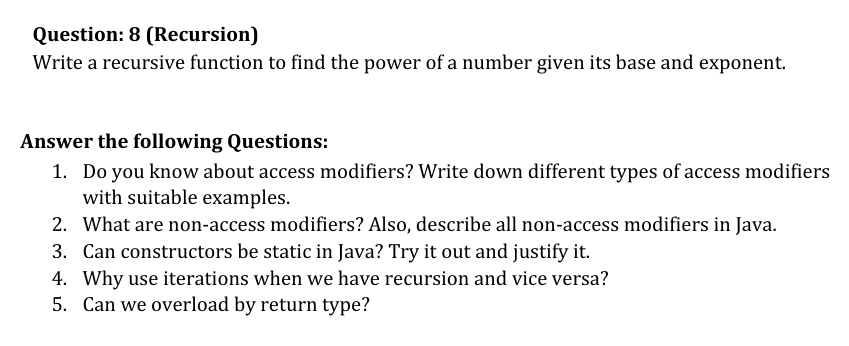
mo.computations(2.5, 9.8);

}

}

Output:





Source code:

public class PowerCalculator {

public static void main(String[] args) {

System.out.println("2^3 = " + power(2, 3));

}

public static int power(int base, int exponent) {

if (exponent == 0) {

return 1;

}

return base \* power(base, exponent - 1);

}

}

Output:

