## JADAVPUR UNIVERSITY

Advanced programming (JAVA and python) LAB ASSIGNMENTS

## Name: SHRUTI PATHAK

## Roll no: 002210503021

## Dept: Computer Science and Engineering

## Class: MCA 1st year 2nd sem

**Session**: 2022-2024

## Java ASSIGNMENT 1:

1. **Write a program to accept two short integers from user and display the sum.**

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_01 {

    public static void main(String args[]) {

        short num1, num2;

        int sum;

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter First number: ");

        num1 = sc.nextShort();

        System.out.println("Enter Second number: ");

        num2 = sc.nextShort();

        sc.close();

        sum = num1 + num2;

        System.out.println("Sum of these numbers: " + sum);

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_01.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_01

Enter First number:

10

Enter Second number:

20

Sum of these numbers: 30

1. **Write a program that accepts number of command line parameters and displays the parameters and count of such parameters.**

**SOURCE CODE:**

public class ass1\_02 {

    public static void main(String args[]) {

        if (args.length == 0)

            System.out.println("No argument passed!!!");

        else {

            System.out.println("The number of command line argument: " + args.length);

            System.out.println("The command line parameters: ");

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

                System.out.println(args[i]);

            }

        }

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_02.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_02

No argument passed!!!

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_02.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_02 A B C

The number of command line argument: 3

The command line parameters:

A

B

C

## Write a program that accepts height in cm as int and displays the height in feet and inches. Assume, 1 inch equals to 2.54 cm and 1 foot equals to 30.5 cm.

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_03 {

    public static void main(String args[]) {

        int cm;

        double inch,foot;

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter height in cm: ");

        cm=sc.nextInt();

        inch=cm/2.54;

        foot=cm/30.5;

        sc.close();

        System.out.println("The height in inch: "+inch);

        System.out.println("The height in foot: "+foot);

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_03.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_03

Enter height in cm:

20

The height in inch: 7.874015748031496

The height in foot: 0.6557377049180327

## Write a program that accepts radius of a circle and displays area of the circle. Declare a constant pi equals to 3.14.

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_04 {

    public static void main(String args[]) {

        final double pi=3.14;

        double r,area;

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter radius of the circle: ");

        r=sc.nextDouble();

        area=pi\*r\*r;

        sc.close();

        System.out.println("Area of the circle is: "+area);

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_04.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_04

Enter radius of the circle:

5

Area of the circle is: 78.5

## Write a program that accepts a String and assigns it to another. Check the outcome of comparison with == and equals() method. Take two Strings and put same input for them. Repeat the equality checking. Observe the outcome.

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_05 {

    public static void main(String args[]) {

        String s1,s2;

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter a string: ");

        s1=sc.nextLine();

        s2=s1;

        System.out.println(s1==s2); //true

        System.out.println(s1.equals(s2)); //true

        System.out.println("Enter first string: ");

        s1=sc.nextLine();

        System.out.println("Enter second string(same as first): ");

        s2=sc.nextLine();

        System.out.println(s1==s2); //false because it compares the address of the strings

        System.out.println(s1.equals(s2)); //true because it compares the content

        sc.close();

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_05.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_05

Enter a string:

shruti

true

true

Enter first string:

abc

Enter second string(same as first):

abc

false

true

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_05

Enter a string:

mno

true

true

Enter first string:

xyz

Enter second string(same as first):

stu

false

false

## Write a program where class contains void show(int) to display the argument passed. Call the function once with short as actual parameter and again double as actual parameter. Add another function as void show(double). Repeat the calls. Observe the outcomes in each case.

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_06 {

    static void show(int n)

    {

        System.out.println(n);

    }

    static void show(double n)

    {

        System.out.println(n);

    }

    public static void main(String args[]) {

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter a short value: ");

        short a=sc.nextShort();

        System.out.println("Enter a double value: ");

        double b=sc.nextDouble();

        show(a);

        show(b);

        sc.close();

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_06.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_06

Enter a short value:

10

Enter a double value:

25.5

10

25.5

## Design a Metric class that supports Kilometre to Mile conversion with distance in Kilometre as argument and Mile to Kilometre conversion with distance in mile as argument. Assume, one Mile equals 1.5 Kilometre.

**SOURCE CODE:**

import java.util.Scanner;

public class ass1\_10 {

    static class Metric{

        double km,mile;

        public double km\_to\_mile(double km) {

            return km/1.5;

        }

        public double mile\_to\_km(double m) {

            return m\*1.5;

        }

    }

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Metric d=new Metric();

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter distance in Kilometer: ");

        d.km=sc.nextDouble();

        System.out.println("Distance in Miles: "+d.km\_to\_mile(d.km));

        System.out.println("Enter distance in mile: ");

        d.mile=sc.nextDouble();

        System.out.println("Distance in kilometer: "+d.mile\_to\_km(d.mile));

        sc.close();

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> javac ass1\_10.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS1> java ass1\_10

Enter distance in Kilometer:

5

Distance in Miles: 3.3333333333333335

Enter distance in mile:

10

Distance in kilometer: 15.0

## Java ASSIGNMENT 2:

## 5. Design a student class with roll, name and score. Support must be there to set the score. Score is non-negative and cannot exceed 100. For invalid score an exception has to be raised. User of set score method will decide about the measures to deal with the exception.

**SOURCE CODE:**

import java.util.\*;

class MyException extends Exception {

    public String toString() {

        return "Score is non-negative and cannot exceed 100";

    }

}

class Student {

    private int roll;

    private String name;

    private double score;

    public void setRoll(int roll) {

        this.roll = roll;

    }

    public void setName(String name) {

        this.name = name;

    }

    public void setScore(double score) throws MyException {

        if (score < 0 || score > 100) {

            throw new MyException();

        }

        this.score = score;

    }

    public int getRoll() {

        return this.roll;

    }

    public String getName() {

        return this.name;

    }

    public double getScore() {

        return this.score;

    }

};

class ass2\_05 {

    public static void main(String args[]) {

        Student s1 = new Student();

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter student roll: ");

        int roll = sc.nextInt();

        s1.setRoll(roll);

        System.out.print("Enter student name: ");

        sc.nextLine();

        String name = sc.nextLine();

        s1.setName(name);

        System.out.print("Enter student's score: ");

        double score = sc.nextDouble();

        try {

            s1.setScore(score);

            System.out.println("Student Roll : " + s1.getRoll() + "\nStudent Name : " + s1.getName()

                    + "\nStudent Score : " + s1.getScore());

        } catch (MyException e) {

            System.out.println("Exception : " + e);

        }

        sc.close();

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> javac ass2\_05.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> java ass2\_05

Enter student roll: 21

Enter student name: Shruti

Enter student's score: 50

Student Roll : 21

Student Name : Shruti

Student Score : 50.0

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> java ass2\_05

Enter student roll: 21

Enter student name: shruti

Enter student's score: 130

Exception : Score is non-negative and cannot exceed 100

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> java ass2\_05

Enter student roll: 21

Enter student name: Shruti

Enter student's score: -50

Exception : Score is non-negative and cannot exceed 100

**6. Consider a wrapper class for a numeric basic type. Check the support for the following: conversion from i) basic type to object ii) object to basic type iii) basic type to String iv) String (holding numeric data) to numeric object v) object to String.**

**SOURCE CODE:**

import java.util.Scanner;

public class ass2\_06 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a integer: ");

        int a = sc.nextInt();

        Integer integer = Integer.valueOf(a);

        System.out.println("Basic type to object => " + integer);

        int b = integer.intValue();

        System.out.println("Object to basic type => " + b);

        String num = String.valueOf(b);

        System.out.println("Basic type to string => " + num);

        Integer newNum = Integer.parseInt(num);

        System.out.println("String (holding numeric data) to  numeric object => " + newNum);

        String newString = Integer.toString(b);

        System.out.println("Object to string => " + newString);

        sc.close();

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> javac ass2\_06.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> java ass2\_06

Enter a integer:

20

Basic type to object => 20

Object to basic type => 20

Basic type to string => 20

String (holding numeric data) to numeric object => 20

Object to string => 20

**7. Take a String input that contains multiple words. Do the following: i) number of times 'a' appears ii) number of times "and" appears iii) whether it starts with "The" or not iv) put the String into an array of characters v) display the tokens in the String (tokens are the substrings separated by space or @ or .)**

**SOURCE CODE:**

import java.util.\*;

public class ass2\_7 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a string: ");

        String inputString = sc.nextLine();

        char[] chars = inputString.toCharArray();

        int count = 0;

        for (char c : chars) {

            if (c == 'a')

                count++;

        }

        System.out.println("Count of \'a\' is " + count);

        int countOfA = 0;

        System.out.println("Enter delimeter::");

        String del = new Scanner(System.in).nextLine();

        String[] arr;

        if (del.equals(" ")) {

            arr = inputString.split(" ");

        } else {

            arr = inputString.split("@");

        }

        String s = "and";

        for (String str : arr) {

            System.out.println(str);

            if (str.compareTo(s) == 0)

                countOfA++;

        }

        System.out.println("Count of \'and\' in the input string " + countOfA);

        if (arr[0].equals("The"))

            System.out.println("Input string starts with \'the\'");

        else

            System.out.println("Input string does not starts with \'the\'");

    }

}

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> javac ass2\_07.java

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\JAVA\ASS2> java ass2\_07

Enter a string:

The sun gives us heat and light for the environment and the growth of organisms.

Count of 'a' is 4

Enter delimeter::

The

sun

gives

us

heat

and

light

for

the

environment

and

the

growth

of

organisms.

Count of 'and' in the input string 2

Input string starts with 'the'

## python ASSIGNMENT 1:

1. **Write a prime generator program using only primes and using python loops.**

**SOURCE CODE:**

min=int(input("Enter the lowest range: "))

max=int(input("Enter the upper range: "))

print("The Prime numbers in the range are: ")

for i in range(min,max+1):

    if i > 1:

        for j in range(2,i):

            if (i%j) == 0:

                break

        else:

            print(i)

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass1.py

Enter the lowest range: 10

Enter the upper range: 30

The Prime numbers in the range are:

11

13

17

19

23

29

1. **Write a discount coupon code using dictionary in Python with different rate coupons for each day of the week.**

**SOURCE CODE:**

import datetime

coupon\_codes={

    "Monday": 0.10,

    "Tuesday": 0.15,

    "Wednesday": 0.20,

    "Thursday": 0.25,

    "Friday": 0.30,

    "Saturday": 0.35,

    "Sunday": 0.40,

}

current\_day= datetime.datetime.now().strftime("%A")

if current\_day in coupon\_codes:

    discount\_rate = coupon\_codes[current\_day]

    coupon\_code= f"DISCOUNT{int(discount\_rate \* 100)}"

    print(f"Today is {current\_day}, and the discount rate is {discount\_rate \*100}%")

    print(f"Use coupon code '{coupon\_code}' at checkout to avail the discount!")

else:

    print("No discount for today")

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass2.py

Today is Saturday, and the discount rate is 35.0%

Use coupon code 'DISCOUNT35' at checkout to avail the discount!

1. **Print first 10 odd and even numbers using iterators and compress. You can use duck typing.**

**SOURCE CODE:**

from itertools import compress,count

numbers=count(1)

odd\_pattern= [True,False]\*10

odd\_numbers= compress(numbers,odd\_pattern)

print("First 10 odd numbers : ")

for \_ in range(10):

    print(next(odd\_numbers),end = " ")

print()

numbers=count(1)

even\_pattern= [False,True]\*10

even\_numbers= compress(numbers,even\_pattern)

print("First 10 even numbers : ")

for \_ in range(10):

    print(next(even\_numbers),end = " ")

print()

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass3.py

First 10 odd numbers :

1 3 5 7 9 11 13 15 17 19

First 10 even numbers :

2 4 6 8 10 12 14 16 18 20

1. **Write a regular expression to validate a phone number.**

**SOURCE CODE:**

import re

n=input('Enter Mobile number : ')

r=re.fullmatch('[6-9][0-9]{9}',n)

if r!=None:

    print('Valid Number')

else:

    print('Not a valid number')

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass4.py

Enter Mobile number : 7679532692

Valid Number

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass4.py

Enter Mobile number : 234567

Not a valid number

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass4.py

Enter Mobile number : oiuyft

Not a valid number

1. **Write first seven Fibinacci numbers using generator next function/ yield in python. Trace  and memorize the function.**

**SOURCE CODE:**

def fibo\_generator():

    a,b=0,1

    count=0

    while count < 7:

        yield a

        a,b=b,a+b

        count +=1

fib\_gen= fibo\_generator()

print("Fibonacci Sequence with 7 terms: ")

for i in range(7):

    fibo\_num=next(fib\_gen)

    print(fibo\_num, end= " ")

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass5.py

Fibonacci Sequence with 7 terms:

0 1 1 2 3 5 8

**6. Write a simple program which loops over a list of user data (tuples containing a username, email and age) and adds each user to a directory if the user is at least 16 years old. You do not need to store the age. Write a simple exception hierarchy which defines a different exception for each of these error conditions:**

* **the username is not unique**
* **the age is not a positive integer**
* **the user is under 16**
* **the email address is not valid (a simple check for a username, the @ symbol and a domain name is sufficient)**

**Raise these exceptions in your program where appropriate. Whenever an exception occurs, your program should move onto the next set of data in the list. Print a different error message for each different kind of exception.**

**SOURCE CODE:**

userdata=[

    ["user1", "user1@gmail.com",12],

    ["user2", "user2@gmail.com",15],

    ["user3", "user3@gmail.com",19],

    ["user3", "user3@gmail.com",19],

    ["user4", "user4@gmail.com",30],

    ["user5", "user5@gmail.com",-21],

    ["user6", "user6@gmail.com",67],

    ["user7", "user7@gmail.com",18],

    ["user8", "user8gmail.com",19]

]

user\_dict={}

for user in userdata:

    try:

        if(user[0] in user\_dict.keys()):

            raise Exception("The username is not unique for: " +user[0])

        elif(user[2] <0):

            raise Exception("The age is not a positive integer for: "+user[0])

        elif(user[2]<16):

            raise Exception("The user is under 16 for: "+user[0])

        elif '@' not in user[1] or '.com' not in user[1]:

            raise Exception("The email address is not valid for: "+user[0])

        else:

            user\_dict[user[0]]= user[1]

    except Exception as e:

        print("Exception occured!!!",str(e))

        continue

print("\nValid users are: ")

print(user\_dict)

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass5.py

Fibonacci Sequence with 7 terms:

0 1 1 2 3 5 8

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass6.py

Exception occured!!! The user is under 16 for: user1

Exception occured!!! The user is under 16 for: user2

Exception occured!!! The username is not unique for: user3

Exception occured!!! The age is not a positive integer for: user5

Exception occured!!! The email address is not valid for: user8

Valid users are:

{'user3': 'user3@gmail.com', 'user4': 'user4@gmail.com', 'user6': 'user6@gmail.com', 'user7': 'user7@gmail.com'}

**8. Create a list of all the numbers up to N=50 which are multiples of five using anonymous function.**

**SOURCE CODE:**

# Create a list of all numbers up to N=50 that are multiples of five using an anonymous function

N = 50

# Use a list comprehension with an anonymous function

multiples\_of\_five = [num for num in range(1, N+1) if (lambda x: x % 5 == 0)(num)]

# Print the list of multiples of five

print(multiples\_of\_five)

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass8.py

[5, 10, 15, 20, 25, 30, 35, 40, 45, 50]

**10. Filter out the odd squares using map, filter, list.**

**SOURCE CODE:**

import math

def is\_odd(num):

    return math.sqrt(num) % 2 != 0

def square(num):

    return num \* num

# List of numbers

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Use map to square each number

squared\_numbers = list(map(square, numbers))

# Use filter to keep only odd numbers

filtered\_numbers = list(filter(is\_odd, squared\_numbers))

print(filtered\_numbers)

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass10.py

[1, 9, 25, 49, 81]

1. **Let's find all Pythagorean triples whose short sides are numbers smaller than 10. use filter and comprehension.**

**SOURCE CODE:**

# Define the function to check if a triple is a Pythagorean triple

def is\_pythagorean\_triple(triple):

    a, b, c = triple

    return a\*\*2 + b\*\*2 == c\*\*2

# Generate all possible combinations of numbers smaller than 10

numbers = range(1, 10)

# Use filter and comprehension to find the Pythagorean triples

pythagorean\_triples = [

    (a, b, c)

    for a in numbers

     for b in numbers

     for c in numbers

    if is\_pythagorean\_triple((a, b, c))

]

# Print the Pythagorean triples

for triple in pythagorean\_triples:

    print(triple)

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> py ass11.py

(3, 4, 5)

(4, 3, 5)

1. **Enumerate the sequence of all lowercase ASCII letters, starting from 1, using**

**enumerate.**

**SOURCE CODE**

list\_chars=["a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q",

            "r","s","t","u","v","w","x","y","z"]

i=1

for ascii,char in enumerate(list\_chars,97):

    print(i,". ",char,"-->",ascii,sep='')

    i=i+1

## output:

PS C:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON> python -u "c:\Users\Shruti Pathak\Documents\MCA 2ND SEM\PYTHON\ass12.py"

1. a-->97

2. b-->98

3. c-->99

4. d-->100

5. e-->101

6. f-->102

7. g-->103

8. h-->104

9. i-->105

10. j-->106

11. k-->107

12. l-->108

13. m-->109

14. n-->110

15. o-->111

16. p-->112

17. q-->113

18. r-->114

19. s-->115

20. t-->116

21. u-->117

22. v-->118

23. w-->119

24. x-->120

25. y-->121

26. z-->122

## 