UNIVERSITY OF RUHUNA
BACHELOR OF INFORMATION AND COMMUNICATION TECHNOLOGY
Practical 10 - Revision
ICT2132 – Object Oriented Programming Practicum

1.   a.   i.   What is an **Object** in object-oriented programing in java. Briefly describe **Three(03) main characteristics** of an object in java.

            ii.   What is the **difference** between a **class** and an **object** in object-oriented programing in java?

            iii.   List down and briefly describe the **four (04)** types of **access modifiers** in Java programming language.

    b.   Briefly explain the following Object-Oriented Programming Concepts by using suitable java code examples.

            i.   **Abstract class**

            ii.   **Tag (Maker) Interface**

    c.   i.   Write **one key role of a constructor** method in a class in Java programming language.

            ii.   List down **two unique features of default constructor** in java programming language.

    d.   Investigate the following java code and answer the questions given below.

```java
public class Student {

        String name = "Perera";

        int age;

        Public void setName(String name){

                name = name;

        }        public void printName(){

                System.out.println(name);

        }

        public static void main(String[] args){

                //your code

        }

}
```

            i.   Write down the java code **segment** which is needed to call **"setName(String name)"** using **reference** variable inside the main method.

            ii.   Write down the java code **statement** which is needed to call **"setName(String name)"** using **anonymous object** inside the main method.

ICT2132 – Object Oriented Programming Practicum

      iii.   What will be the **output** of the program if you insert the following code segment inside the main method? Explain the reason?

```
Student stu = new Student();
stu.setName("Nimal");
stu.printName();
```

      iv.   After including the above code segment what are the **other code changes** to be done in order to get **"Priyantha"** as the output.

      v.   Write down a **parameterized constructor** using java for the above class to assign values to each of its attributes.

2.   a.

```
class Calculation{

        void printData(string name, int age){

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

        }

        void printData(int age, string name){

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

        }

}
```
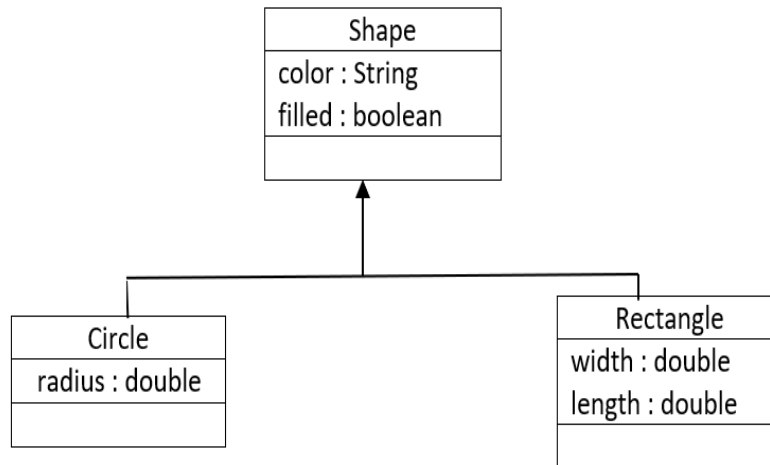
      i.   What is the **OOP concept** used in above given code segment?

      ii.   Can we achieve the same thing identified in (a) (i.) by changing the number of arguments or by changing the data type of arguments?
If yes briefly explain it with examples.

      iii.   Write down **an advantage** using **Method Overriding** in java programming language.

    b.   i.   Define the terms **accessors** and **mutators** in object-oriented programming in java?

      ii.   Write a simple java program to create a class called **Account** according to the following specifications.
- There are **two properties as name** and **balance** that are **inaccessible from the outside** the class
- There should be **accessor** and **mutator methods** for the above properties in the Account class.

Create **another class containing a main method** and create an object from the **Account** class. Invoke the accessor and mutator methods from the object you created.

c.   Consider the following class diagram and answer the questions.

```
           ┌─────────────────┐
           │      Shape      │
           ├─────────────────┤
           │ color : String  │
           │ filled : boolean│
           ├─────────────────┤
           │                 │
           └─────────────────┘
                    △
                    │
        ┌───────────┴────────────┐
┌─────────────────┐      ┌─────────────────┐
│      Circle     │      │    Rectangle    │
├─────────────────┤      ├─────────────────┤
│ radius : double │      │ width : double  │
│                 │      │ length : double │
├─────────────────┤      ├─────────────────┤
│                 │      │                 │
└─────────────────┘      └─────────────────┘
```

      i.   What is the **object-oriented principle** used in the above class diagram?

      ii.   Briefly explain **two advantages** of using the principle mentioned in part (c) (i) in object-oriented programming.

      iii.   Write **Java code segments for each class** in the above class diagram. (Consider Shape and Rectangle are Interfaces and No need to consider about access modifiers)

      iv.   Using **examples from the above class diagram** briefly **describe "Implicit" and "Explicit" casting** in java.

3.   a.   i.   List down **three (03) different situations** where an **exception** can occur in java programming.

      ii.   Consider the below given code segment.

```java
class MyArray {

        public static void main(String[] args) {

                int arr[] ={1,2,3,4,5};

                System.out.println(arr[7]);

        }

}
```

What will happen when you **compile and run** the above java class.
Using your knowledge in exceptions write down the **complete java program** which will ensure the **smooth flow of the program**.

      iii.   Assume that there is a method called **checkEligibility()** to check the student eligibility, which **takes a double type "marks"** as the **input parameter**.
If (marks >= 80.00)
        prints "Eligible"
If (marks < 80.00)
        generate a checked exception of "NotEligible"

The **NotEligible** exception class has only a **single argument parameterized constructor** which takes a **String** value. Considering these requirements, **write a java code segment for checkEligibility () method**.
(Hint : Use custom exceptions knowledge)

b.   Consider the below given code segment.

```
public class PrintNumbersDemo {
        public static void main(String[] args) {
                PrintNumbers pn = new PrintNumbers();
                Pn.start();
        }
}
```

```
public class PrintNumbers {
        public void start() {
                System.out.print("1");
                try{
                        divide();           //Line A
                        print();            //Line B
                }
                catch(NullPointerException e){
                        System.out.print("2");
                }
                finally{
                        System.out.print("3");
                }
                System.out.print("4");
        }
        public void divide() throws ArithmaticException{
                System.out.print("5");
                Int x = 5/0 ;
                System.out.print("6");
        }
        public void print() throws NullPointerException{
                System.out.print("7");
                String a = null;
                a.toString();
                System.out.print("8");
        }
}
```

a.   Write down the **output** when you **compile and run** the above Java program. Write down the **reasons** for your answer.

b.   Write down the **output** when you **interchange Line A** with **Line B** and **compile and run** the above Java program. Write down the **reasons** for your answer.

4.   a.   i.   List down **five (05) key features** of Java.

ii.   Briefly describe what is known as **"Write once, Run everywhere"** concept related to Java?

b.   Write down the answers for following question by considering the **elements and their order in a Java Class**.

Element	Example Code Segment	Required (Yes/No)	Where it's appearing in the Class (Select from the below given **List A**)
Comments			
Package declaration			
Import statements			
Class declaration			
Method declaration			

List A

Anywhere inside the class scope,   Anywhere,

Immediately after the package,   First line in the file,

Immediately after the import

c.   i.   List down **two (02)** key differences between **"Java Primitive Data Types"** and **"Java Reference Data Types"**.

ii.   By using **two (02) key points** briefly describe the **difference between "Java Object"** and **"Java Object Reference"**.

d.   Investigate the following java code and answer the questions given below.

```java
public class ForLoopDemo {
        public static void main(String[] args){
                int age = 10;
                String names[] = { "Nimal", "Kamal" };

                for ( int i = 0, age =10 ; i < names.length ; i++)
                {
                        String name = names[i];
                        System.out.println(name + " , " + age);
                }
        }
}
```

Practical 10 - Revision

ICT2132 – Object Oriented Programming Practicum

      i.   What will be the **output** when you **compile and run** the above Java program. Explain the **reason/s** for your answer.

      ii.   Write down the **Java code changes** which needs to be done in above class in order to get following output.

> **Nimal , 10**
>
> **Kamal , 15**

      iii.   **Replace** the **for loop** with a **for-each** loop to obtain the same output in part (d) (ii) section.

e.   Investigate the following Java code and answer the questions given below.

```java
public class Person {
        String name = "Nimal";
        int age;

        public (String name){
                name = this.name;
        }

        public static void main(String[] args){
                Person kamal = new Person( "Kamal" );
                System.out.println( kamal.name );
                System.out.println( kamal.age );
        }
}
```

      i.   What will be the **output** of the program. Explain the **reason/s** for your answer.

      ii.   Write down the **required changes in the above given code** to obtain following output by using **constructor**

> **Kamal**
> **20**