PAGE No.
DATE / /

DESIGN A CLASS RECTANGLE WITH DATA FIELDS WIDTH, LENGTH, AREA AND COLOR. THE LENGTH, WIDTH AND AREA ARE OF DOUBLE TYPE AND COLOR IS A STRING TYPE. THE METHODS ARE

GET_LENGTH()
GET_WIDTH()
GET_COLOR() &
FIND_AREA()

CREATE 2 OBJECTS OF RECTANGLE AND COMPARE THEIR AREA AND COLOR IF THE AREA AND COLOR BOTH ARE THE SAME FOR THE OBJECTS THEN DISPLAY "MATCHING RECTANGLES" OTHER WISE DISPLAY "NON MATCHING RECTANGLES"

DIAGRAM:

~~RU~~ ACCESS MODIFIER TABLE

THEORY:

CLASS & OBJECTS THEORY

QUESTIONS

i) WHAT IS CLASS & OBJECT IN OOP?
ii) HOW TO DEFINE ANY CLASS IN JAVA
iii) EXPLAIN EVERY CLASS COMPONENT IN DETAIL!
iii) WHAT ARE THE DIFFERENT ACCESS SPECIFIERS IN JAVA AND EXPLAIN IN

iv) DETAILS WITH EX
WHAT IS PACKAGE & INTERFACE
v) WHAT IS THE SIGNATURE FOR SUB CLASS & CLASS HAVE IMPLEMENTATION OF INTERFACE
vi) WHAT IS THE SIGNATURE OF METHOD IN JAVA & EXPLAIN EACH COMPONENT OF METHOD IN DETAIL
vii) HOW TO CREATE AND INITIALIZE THE OBJECT IN JAVA
viii) HOW TO ACCESS THE MEMBERS OF CLASS USING OBJECTS
ix) WHAT IS DIFF IN C++ & JAVA CLASS
x) CAN WE CREATE OBJ FOR MAIN METHOD CLASS

BASE CLASS → SUPER CLASS
CHILD CLASS → SUB CLASS

MULTIPLE INHERITANCE IS NOT IN JAVA WE CANNOT USE EXTEND MORE THAN ONCE IN JAVA
INSTEAD WE USE INTERFACES

1) NORMAL CLASS SIGNATURE

i) ACCESS_SPECIFIER CLASS CLASS_NAME {
    FIELD DECLARATION;
    METHOD DEFINITION;
}

Ex)

```
PUBLIC CLASS STUDENT {
...
...
}
```

2) SIGNATURE OF CLASS DURING INHERITANCE

i) CLASS COLLEGE

```
{
...
...
}
```

} SUPER CLASS

CLASS ETC EXTENDS COLLEGE

```
{
...
...
}
```

} SUB CLASS

3) SIGNATURE OF CLASS DURING IMPLEMENTATION OF INTERFACE

~~int~~ INTERFACE I_2

```
{
VOID DISPLAY2 (); // ONLY FUNC^N DECLARATION
}
```

```
INTERFACE I1
{
  void VOID DISPLAY();
}

CLASS XYZ IMPLEMENTS I1, I2 {
    VOID DISPLAY() {
    ...
    ...
    }
    VOID DISPLAY2() {
    ...
    ...
    }
}
```

ACCESS_SPECIFIER CAN BE USED IN

i) CLASS
ii) METHODS
iii) VARIABLES

CLASS CAN ONLY BE PUBLIC OR DEFAULT

i) PUBLIC: IN THE JAVA PROGRAM ONLY ONE CLASS CAN BE PUBLIC OR NONE.

BECAUSE JAVA COMPILER STARTS ITS EXECUTION FROM THE PUBLIC CLASS

ii) PRIVATE | PROTECTED: NOT USED IN CLASSES AS WE CANNOT USE ITS FUNCTIONS AND DATA MEMBERS

METHOD

ACCESS LOCATION	PRIVATE	PROTECTED	DEFAULT	PUBLIC
i) SAME CLASS	YES	YES	YES	YES
ii) SAME PACKAGE SUB CLASS	NO	YES	YES	YES
iii) SAME PACKAGE NON SUB CLASS	NO	YES	YES	YES
iv) DIFF PACKAGE SUBCLASS	NO	YES	NO	YES
v) DIFF PACKAGE NON SUB CLASS	NO	NO	NO	YES

COMPARE 2 STRINGS

R1.color.equals(R2.color)
R1.color.equalsIgnoreCase( )

```
package p1;
class Demo  //Same class
 {
    Void display()
     {
       System.out.println("HELLO");
     }
     void display2()
     {
       display();
     }
 }

class Demo1 extends Demo //Same package sub class
   {
    void display3()
     {
      display();
     }
   }

class Demo3 //Same package non sub class
   {
    void demo3_display()
     {
       Demo D1 = new Demo();
       D1.display();
     }
}
```

```
Package p2;
import p1.*;

class Demo4 extends Demo //Different package subclass
  {
    void demo4_display()
      {
        display()
      }
  }

class Demo5 //Different package non sub class
  {
    Demo d1 = new Demo()
    d1.display()
  }
```