NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
OOPS ASSIGNMENT-2

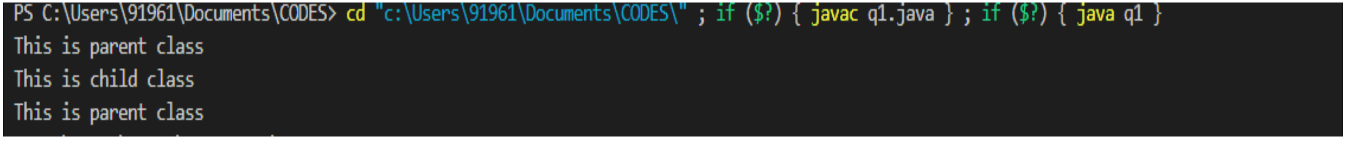
Name: P MANOHAR RAO

Roll No:197158

Section: A

1.Create a class with a method that prints "This is parent class" and its subclass with another
method that prints "This is child class". Now, create an object for each of the class and call
1 - method of parent class by object of parent class
2 - method of child class by object of child class
3 - method of parent class by object of child class

```
class parent
{
void pmethod()
{
System.out.println("This is parent class");
}
} class child extends parent
{
void cmethod()
{
System.out.println("This is child class");
}
} class q1
{
public static void main(String[] args)
{
child A=new child();
parent B=new parent();
B.pmethod();
A.cmethod();
A.pmethod();
}
}
```

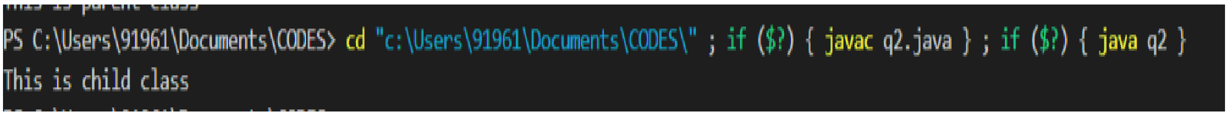


```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac q1.java } ; if ($?) { java q1 }
This is parent class
This is child class
This is parent class
```

2.
In the above example, declare the method of the parent class as private and then repeat the first

two operations (You will get error in the third).

```
class parent
{
private void pmethod()
{
System.out.println("This is parent class");
}
} class child extends parent
{
void cmethod()
{
System.out.println("This is child class");
}
} class q2
{
public static void main(String[] args)
{
child A=new child();
parent B=new parent();
A.cmethod();
}
}
```



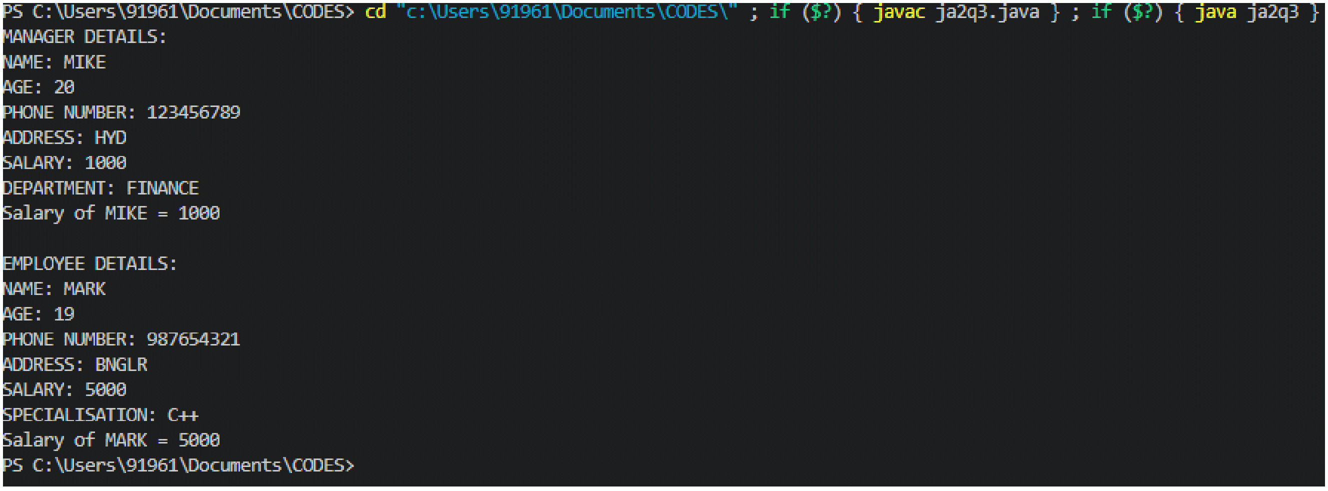
```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac q2.java } ; if ($?) { java q2 }
This is child class
```

3.
Create a class named 'Member' having the following members:
Data members
1 - Name
2 - Age
3 - Phone number
4 - Address
5 - Salary
It also has a method named 'printSalary' which prints the salary of the members. Two classes
'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have
data members 'specialization' and 'department' respectively. Now, assign name, age, phone
number, address and salary to an employee and a manager by making an object of both of these
classes and print the same.

```
class Member
{
String name;
int age;
```

```
String phno;
String address;
int salary;
Member(String name,int age,String phno,String address,int salary)
{
this.name=name;
this.age=age;
this.phno=phno;
this.address=address;
this.salary=salary;
}
void printSalary()
{
System.out.println("Salary of "+name+" = "+salary);
}
void display()
{
System.out.println("NAME: "+name);
System.out.println("AGE: "+age);
System.out.println("PHONE NUMBER: "+phno);
System.out.println("ADDRESS: "+address);
System.out.println("SALARY: "+salary);
}
} class Employee extends Member
{
String specialisation;
Employee(String specialisation,String name,int age,String phno,String address,int salary)
{
super(name,age,phno,address,salary);
this.specialisation=specialisation;
}
void display()
{
System.out.println("EMPLOYEE DETAILS:");
super.display();
System.out.println("SPECIALISATION: "+specialisation);
}
} class Manager extends Member
{
String department;
Manager(String department,String name,int age,String phno,String address,int salary)
{
super(name,age,phno,address,salary);
this.department=department;
}
void display()
{
System.out.println("MANAGER DETAILS:");
super.display();
System.out.println("DEPARTMENT: "+department);
}
} class ja2q3
{
public static void main(String[] args)
{
```

```
Manager m=new Manager("FINANCE","MIKE",20,"123456789","HYD",1000);
Employee e=new Employee("C++","MARK",19,"987654321","BNGLR",5000);
m.display();
m.printSalary();
System.out.println();
e.display();
e.printSalary();
}}
```

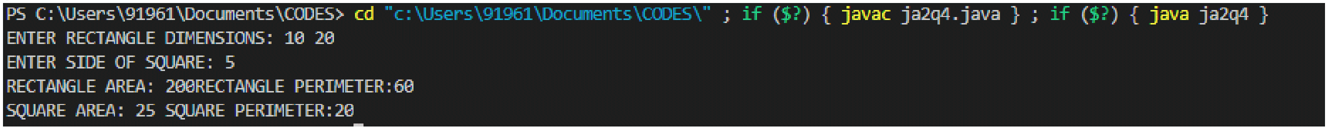


```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q3.java } ; if ($?) { java ja2q3 }
MANAGER DETAILS:
NAME: MIKE
AGE: 20
PHONE NUMBER: 123456789
ADDRESS: HYD
SALARY: 1000
DEPARTMENT: FINANCE
Salary of MIKE = 1000

EMPLOYEE DETAILS:
NAME: MARK
AGE: 19
PHONE NUMBER: 987654321
ADDRESS: BNGLR
SALARY: 5000
SPECIALISATION: C++
Salary of MARK = 5000
PS C:\Users\91961\Documents\CODES>
```

4.

Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super(s,s)'. Print the area and perimeter of a rectangle and a square

```
import java.util.*;
class Rectangle
{
private int l,b;
Rectangle(int l,int b)
{
this.l=l;
this.b=b;
}
int rectArea()
{
return l*b;
}
int rectPerimeter()
{
return 2*(l+b);
}
} class Square extends Rectangle
{
Square(int s)
{
super(s,s);
}
} class ja2q4
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
System.out.print("ENTER RECTANGLE DIMENSIONS: ");
int l=sc.nextInt(),b=sc.nextInt();
Rectangle R=new Rectangle(l, b);
System.out.print("ENTER SIDE OF SQUARE: ");
int s=sc.nextInt();
Square S=new Square(s);
System.out.println("RECTANGLE AREA: "+R.rectArea()+ "RECTANGLE
PERIMETER:"+R.rectPerimeter());
System.out.println("SQUARE AREA: "+S.rectArea()+" SQUARE PERIMETER:"+S.rectPerimeter());
}
}
```



```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q4.java } ; if ($?) { java ja2q4 }
ENTER RECTANGLE DIMENSIONS: 10 20
ENTER SIDE OF SQUARE: 5
RECTANGLE AREA: 200RECTANGLE PERIMETER:60
SQUARE AREA: 25 SQUARE PERIMETER:20
```

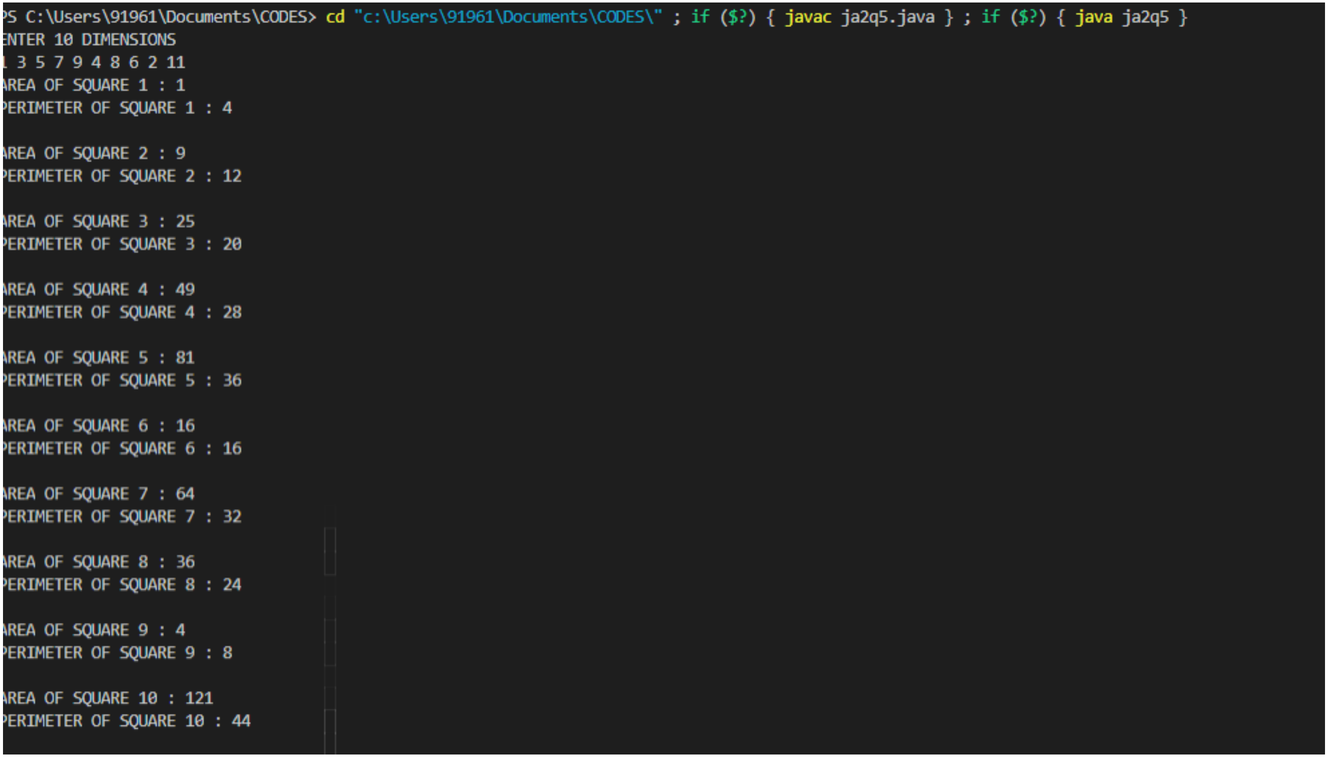
5.

Now repeat the above example to print the area of 10 squares.

Hint-Use array of objects.

```
import java.util.*;
class Rectangle
{
private int l,b;
Rectangle(int l,int b)
{
this.l=l;
this.b=b;
}
int rectArea()
{
return l*b;
}
int rectPerimeter()
{
return 2*(l+b);
}
} class Square extends Rectangle
{
Square(int s)
{
super(s,s);
}
} class ja2q5
{
```

```
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
Square S[]=new Square[10];
System.out.println("ENTER 10 DIMENSIONS");
for(int i=0;i<10;i++)
{
int s=sc.nextInt();
S[i]=new Square(s);
}
for(int i=0;i<10;i++)
{System.out.println("AREA OF SQUARE "+(i+1)+" : "+S[i].rectArea());
System.out.println("PERIMETER OF SQUARE "+(i+1)+" : "+S[i].rectPerimeter()+"\n");
}
}
}
```



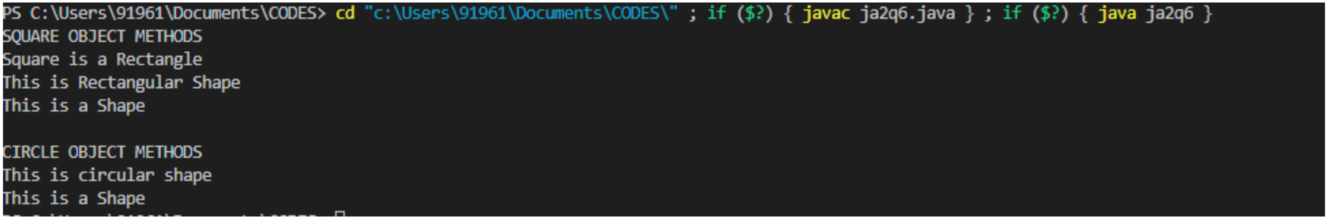
```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q5.java } ; if ($?) { java ja2q5 }
ENTER 10 DIMENSIONS
1 3 5 7 9 4 8 6 2 11
AREA OF SQUARE 1 : 1
PERIMETER OF SQUARE 1 : 4

AREA OF SQUARE 2 : 9
PERIMETER OF SQUARE 2 : 12

AREA OF SQUARE 3 : 25
PERIMETER OF SQUARE 3 : 20

AREA OF SQUARE 4 : 49
PERIMETER OF SQUARE 4 : 28

AREA OF SQUARE 5 : 81
PERIMETER OF SQUARE 5 : 36

AREA OF SQUARE 6 : 16
PERIMETER OF SQUARE 6 : 16

AREA OF SQUARE 7 : 64
PERIMETER OF SQUARE 7 : 32

AREA OF SQUARE 8 : 36
PERIMETER OF SQUARE 8 : 24

AREA OF SQUARE 9 : 4
PERIMETER OF SQUARE 9 : 8

AREA OF SQUARE 10 : 121
PERIMETER OF SQUARE 10 : 44
```

6.

Create a class named 'Shape' with a method to print "This is This is shape". Then create two other
classes named 'Rectangle', 'Circle' inheriting the Shape class, both having a method to print "This
is rectangular shape" and "This is circular shape" respectively. Create a subclass 'Square' of
'Rectangle' having a method to print "Square is a rectangle". Now call the method of 'Shape' and
'Rectangle' class by the object of 'Square' class.

```
class Shape
{
```

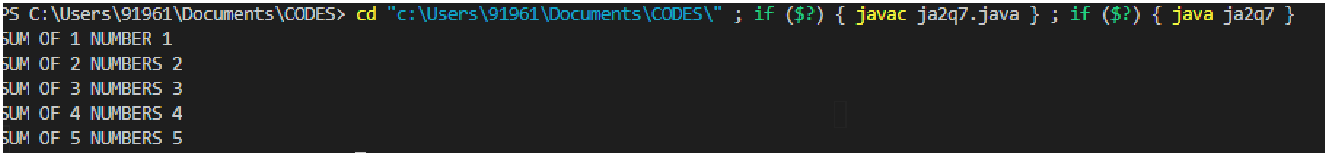
```
void shapeMsg()
{
System.out.println("This is a Shape");
}
} class Circle extends Shape
{
void circleMsg()
{
System.out.println("This is circular shape");
}
} class Rectangle extends Shape
{
void rectMsg()
{
System.out.println("This is Rectangular Shape");
}
} class Square extends Rectangle
{
void squareMsg()
{
System.out.println("Square is a Rectangle");
}
} class ja2q6
{
public static void main(String[] args)
{
Square sq=new Square();
Circle cir=new Circle();
System.out.println("SQUARE OBJECT METHODS");
sq.squareMsg();
sq.rectMsg();
sq.shapeMsg();
System.out.println("\nCIRCLE OBJECT METHODS");
cir.circleMsg();
cir.shapeMsg();
}
}
```



```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q6.java } ; if ($?) { java ja2q6 }
SQUARE OBJECT METHODS
Square is a Rectangle
This is Rectangular Shape
This is a Shape

CIRCLE OBJECT METHODS
This is circular shape
This is a Shape
```

7.
Write a Java Program to find the sum of multiple numbers using Method Overloading

```
class ja2q7
{
static int sum(int a)
{
return a;
}
static int sum(int a,int b)
{
return a+b;
}
static int sum(int a,int b,int c)
{
return a+b+c;
}
static int sum(int a,int b,int c,int d)
{
return a+b+c+d;
}
static int sum(int a,int b,int c,int d,int e)
{
return a+b+c+d+e;
}
public static void main(String[] args)
{
System.out.println("SUM OF 1 NUMBER "+sum(1));
System.out.println("SUM OF 2 NUMBERS "+sum(1,1));
System.out.println("SUM OF 3 NUMBERS "+sum(1,1,1));
System.out.println("SUM OF 4 NUMBERS "+sum(1,1,1,1));
System.out.println("SUM OF 5 NUMBERS "+sum(1,1,1,1,1));
}
}
```



```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q7.java } ; if ($?) { java ja2q7 }
SUM OF 1 NUMBER 1
SUM OF 2 NUMBERS 2
SUM OF 3 NUMBERS 3
SUM OF 4 NUMBERS 4
SUM OF 5 NUMBERS 5
```

8.

Write a program in Java to implement a calculator having four functions such addition, multiplication, division, and subtraction, where the four said functions are defined in four
different packages. Inputs are user defined and use the concept of inheritance for the division
operation.

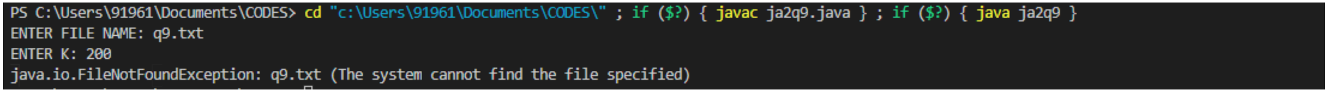
```
package addition;
public class add
{
public int sum(int x,int y)
{
return x+y;
}
}

package subtraction;
public class sub
{
public int subtract(int x,int y)
{
return x-y;
}
}
```

```
package multiplication;
public class mul
{
public int multiply(int x,int y)
{
return x*y;
}
}
package division;
class div
{
int divide(int x,int y)
{
return x/y;
}
}

package division;
import java.lang.*;
import java.util.*;
import addition.*;
import subtraction.*;
import multiplication.*;
class ja2q8 extends div
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
System.out.println("ENTER 2 INTEGERS");
int x=sc.nextInt(),y=sc.nextInt();
div d=new div();
mul m=new mul();
add a=new add();
sub s=new sub();
System.out.println("sum= "+a.sum(x,y));
System.out.println("diff= "+s.subtract(x,y));
System.out.println("product= "+m.multiply(x,y));
System.out.println("division= "+d.divide(x,y));
}
}
```

9.
Write a java program to throw a custom exception, you are trying to read data from the file. If the
file contains more than the specified number (K) characters, k should be taken from the user and
k>=100.

```
import java.util.*;
import java.io.*;
class Myexception extends RuntimeException
{
```

```
int k;
Myexception(int k)
{
this.k=k;
}
public String toString()
{
return "MORE THAN "+k+" CHARACTERS ENTERED";
}
} class ja2q9
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
System.out.print("ENTER FILE NAME: ");
String file=sc.nextLine();
System.out.print("ENTER K: ");
int k=sc.nextInt();
FileInputStream fin=null;
try
{
fin=new FileInputStream(file);
}
catch (FileNotFoundException e)
{
System.out.println(e);
}
if(fin!=null)
{
int i,x=0;
while(true)
{
try
{
i=fin.read();
if(i==-1)
{
System.out.println();
break;
}
x++;
if(x>k)
{
Myexception e=new Myexception(k);
System.out.println();
System.out.println(e);
throw e;
}
System.out.print((char)i);
}
catch(IOException e)
{
System.out.println(e);
}
}
try
{
fin.close();
}
catch (Exception e)
{
System.out.println(e);
}
}
}
}
```



```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q9.java } ; if ($?) { java ja2q9 }
ENTER FILE NAME: q9.txt
ENTER K: 200
java.io.FileNotFoundException: q9.txt (The system cannot find the file specified)
```

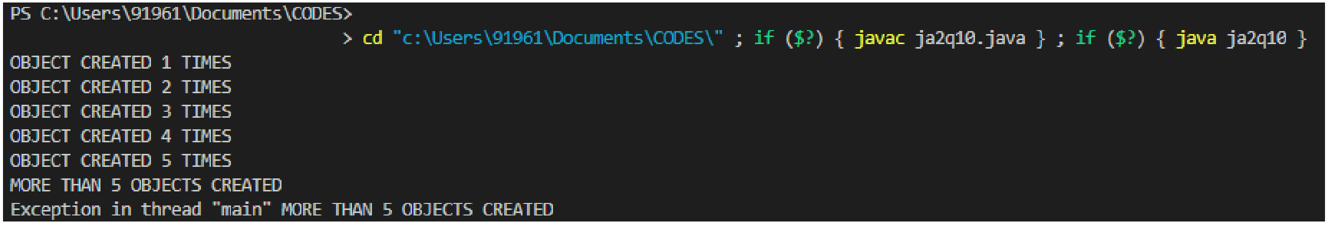
10.

It is known that we can create a number of objects for a class. But from the requirement analysis,

we come to know that only 5 objects have to be created for a particular class.So, if we try to create

more than 5 objects to a class throw a custom exception manually to JVM. Print the corresponding exception message onto the output screen.

```
class exception extends RuntimeException
{
public String toString()
{
return "MORE THAN 5 OBJECTS CREATED";
}
} class Class
{
static int x=0;
Class()
{
if(x==5)
throw new exception();
x++;
System.out.println("OBJECT CREATED "+x+" TIMES");
}
} class ja2q10
{
public static void main(String[] args)
{
for(int i=0;i<10;i++)
{
try
{
Class obj=new Class();
}
catch (exception e)
{
System.out.println(e);
throw e;
}
}
}}
```



```
PS C:\Users\91961\Documents\CODES>
                                > cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q10.java } ; if ($?) { java ja2q10 }
OBJECT CREATED 1 TIMES
OBJECT CREATED 2 TIMES
OBJECT CREATED 3 TIMES
OBJECT CREATED 4 TIMES
OBJECT CREATED 5 TIMES
MORE THAN 5 OBJECTS CREATED
Exception in thread "main" MORE THAN 5 OBJECTS CREATED
```

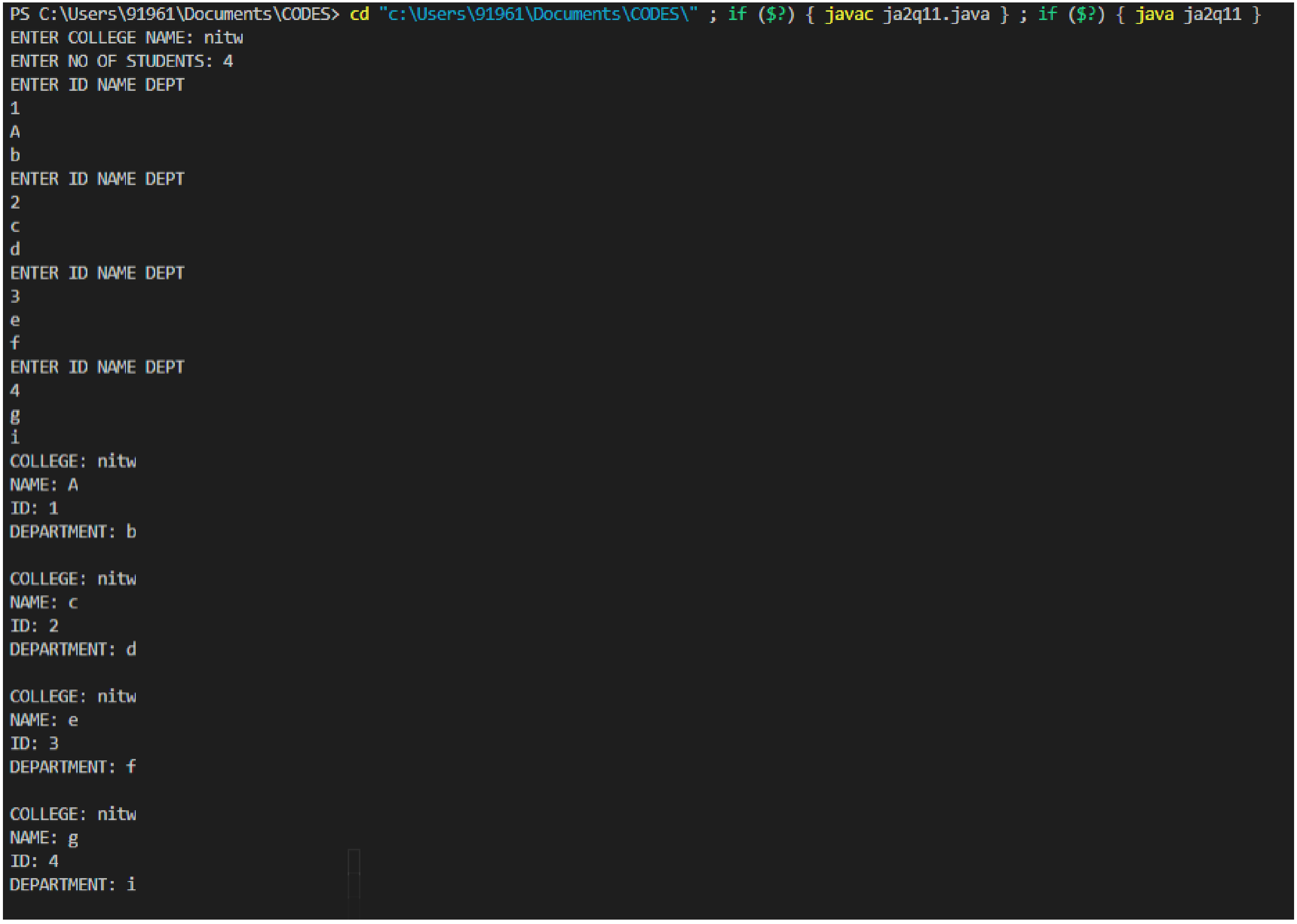
11.

You have been assigned to create a student database of some college. So for a class named Studentinfo create several objects each object is for one student (which means you have to create an array of objects). Studentinfo class is a different class the main class is different.

The number of students should be taken from user input. Student details should be like id, name, department name, college name. College names should be common to all of them. This means the college name does not belong to the object it should belong to the class. So college name is class variable. Now, the student id or name should be unique. If any of these are repeated you should throw a custom exception that which field is repeated (for example, if id is repeated then exception message, should be id repeated.) if everything is fine then print the student data in form (id, name, department name, college). Note: The number of students should be at least 15.

```
import java.util.*;
class idException extends RuntimeException
{
private int id;
idException(int id)
{
this.id=id;
}
public String toString()
{
return "ID "+id+" ALREADY EXISTS RE-ENTER VALUES";
}
}
class nameException extends RuntimeException
{
private String name;
nameException(String name)
{
this.name=name;
}
public String toString()
{
return "NAME "+name+" ALREADY EXISTS RE-ENTER VALUES";
}
}
class Studentinfo
{
static String college;
```

```
int id;
String name,dept;
Studentinfo(int id,String name,String dept)
{
this.id=id;
this.dept=dept;
this.name=name;
}
void print(int n,Studentinfo a[])
{
for(int i=0;i<n;i++)
{
System.out.println("COLLEGE: "+a[i].college);
System.out.println("NAME: "+a[i].name);
System.out.println("ID: "+a[i].id);
System.out.println("DEPARTMENT: "+a[i].dept+"\n");
}
}
} class ja2q11
{
static void check(int id,String name,String dept,Studentinfo a[],int n)
{
for(int i=0;i<n;i++)
{
if(a[i].id==id)
throw new idException(id);
if(a[i].name.equals(name))
throw new nameException(name);
}
}
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
System.out.print("ENTER COLLEGE NAME: ");
Studentinfo.college=sc.nextLine();
System.out.print("ENTER NO OF STUDENTS: ");
int n=sc.nextInt();
Studentinfo a[]=new Studentinfo[n];
for(int i=0;i<n;i++)
{
System.out.println("ENTER ID NAME DEPT");
int flag=1,id=sc.nextInt();
String esc=sc.nextLine();nitw
String name=sc.nextLine(),dept=sc.nextLine();
try
{
check(id, name, dept, a, i);
}
catch(idException e)
{
System.out.println(e);
i--;
flag=0;
}
catch(nameException e)
{
System.out.println(e);
i--;
flag=0;
}
if(flag==1)
a[i]=new Studentinfo(id, name, dept);
}
a[0].print(n, a);
}
}
```

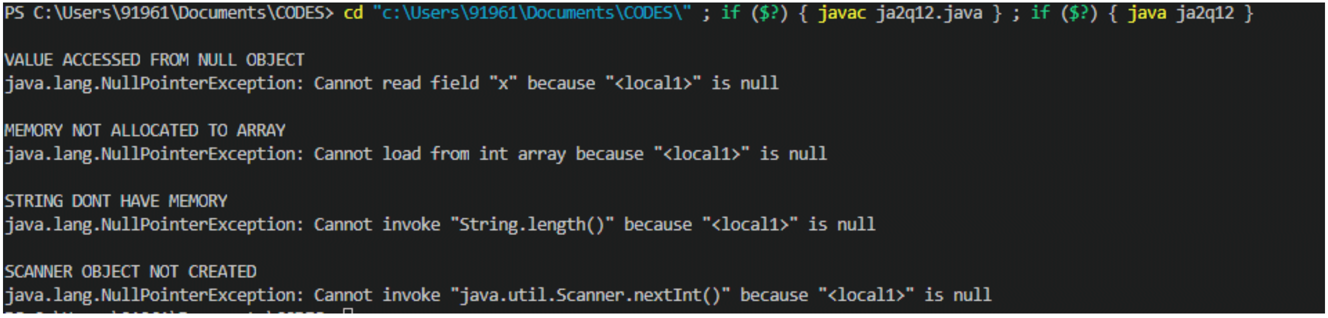


```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q11.java } ; if ($?) { java ja2q11 }
ENTER COLLEGE NAME: nitw
ENTER NO OF STUDENTS: 4
ENTER ID NAME DEPT
1
A
b
ENTER ID NAME DEPT
2
c
d
ENTER ID NAME DEPT
3
e
f
ENTER ID NAME DEPT
4
g
i
COLLEGE: nitw
NAME: A
ID: 1
DEPARTMENT: b

COLLEGE: nitw
NAME: c
ID: 2
DEPARTMENT: d

COLLEGE: nitw
NAME: e
ID: 3
DEPARTMENT: f

COLLEGE: nitw
NAME: g
ID: 4
DEPARTMENT: i
```

12.

Write a java program to illustrate 4 different possible NullPointerException cases

```
import java.util.Scanner;
class abc
{
int x;
abc()
{}
} class ja2q12
{
public static void main(String[] args)
{
try
{
abc obj=null;
int x=obj.x;
}
catch (NullPointerException e)
{
System.out.println("\nVALUE ACCESSED FROM NULL OBJECT");
System.out.println(e);
}
try
{
int a[]=null;
int x=a[0];
}
catch (NullPointerException e)
{
System.out.println("\nMEMORY NOT ALLOCATED TO ARRAY");
System.out.println(e);
}
try
{
String s=null;
int x=s.length();
}
catch (NullPointerException e)
{
System.out.println("\nSTRING DONT HAVE MEMORY");
System.out.println(e);
}
try
{
Scanner sc=null;
int x=sc.nextInt();
}
catch (NullPointerException e)
{
System.out.println("\nSCANNER OBJECT NOT CREATED");
System.out.println(e);
}
}
}
```



```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac ja2q12.java } ; if ($?) { java ja2q12 }

VALUE ACCESSED FROM NULL OBJECT
java.lang.NullPointerException: Cannot read field "x" because "<local1>" is null

MEMORY NOT ALLOCATED TO ARRAY
java.lang.NullPointerException: Cannot load from int array because "<local1>" is null

STRING DONT HAVE MEMORY
java.lang.NullPointerException: Cannot invoke "String.length()" because "<local1>" is null

SCANNER OBJECT NOT CREATED
java.lang.NullPointerException: Cannot invoke "java.util.Scanner.nextInt()" because "<local1>" is null
```