WEEK-10

Father's & Son's age

```
import java.util.*;
class WrongAge extends Exception.
{
    public string toString()
{
    return " Invalid Age Exception";
    }
}
  class Father
{
    int father_age;
    Father (int a) throws Wrong Age
    {
    Scanner sc = new Scanner (System.in).;
  System.out.println ("Enter the age of Father's : ");
  father_age = sc.nextInt ();
  if (father_age <= a || father_age <0 || a< 0)
  {
    throw new WrongAge ();
  }
  }
}.
  class Son extends father
  {
```

```
int son_age;
Son (int a) throws Wrong age.
{
  super (a);
  son_age = a;
  }
  }
public static void main (String Args[])
{
System.out.println ("Enter the age of Son's : ");
try
{
Son s1 = new Son (new Scanner (System.in).nextInt());
}
catch (Wrong age e)
{
System.out.println (e);
}
}
}
```

Generics

```
import java.util.*;
class TwoGen <A, B>
{
```

```
A ob1;
B ob2;

Two Gen (A o1, B o2)
{
    ob1 = o1;
    ob2 = o2;
}
void show_types().
{
    System.out.println("Type 1: " + ob1.getClass().getName());
    System.out.println("Type 2: " + ob2.getClass().getName());
}
A getob1()
{
    return ob1;
}
B getob2()
{
    return ob2;
public class Generics
{
    public static void main(String arg[])
{   Two Gen<Integer, String> obj = new Two Gen<Integer, String>
                                        (20, "Generics");
    obj.showtypes();
    int t=obj.getob1();
    System.out.println("Value = " +t);
    String str=obj.getob2();
    System.out.println("Value = " +str);
}
```