Week 5:

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.

Code:

```
import java.util.Scanner;


class Wrongage extends Exception
{ int detail;
        Wrongage(int d)
        {
        detail=d;
        }
        public String toString()
        {
        return "Entered Wrong age is ["+detail+"]";
        }

}

class Father {
int f;
Scanner in=new Scanner(System.in);
Father()
{

System.out.println("Enter father age ");
f=in.nextInt();
}
void checkage() throws Wrongage
{
if(f<0)
{
throw new Wrongage(f);
}
```

```
System.out.println("Father age positive");
}
}


class Son extends Father{
int s;
Scanner in=new Scanner(System.in);
Son()
{
super();
System.out.println("Enter son age ");
s=in.nextInt();
}

void checkages() throws Wrongage
{
super.checkage();
if(s<0)
{
throw new Wrongage(f);
}
System.out.println("Son age positive");
}


void checkage() throws Wrongage
{
if(s>f)
{
throw new Wrongage(s);
}
System.out.println("Father-Son age correct");
}

}


class Exceptionsssss{
public static void main(String args[])
```

```
{

int f,s;

Father fath=new Father();

Father r;
r=fath;
try{
r.checkage();
}
catch(Wrongage e){
System.out.println("Father age wrong"+e);
}
Son sn=new Son();
r=sn;

try{

sn.checkages();
r.checkage();
}
catch(Wrongage e){
System.out.println("Son age wrong"+e);
}
}
}
```

Output:

```
Enter father age
-10
Father age wrongEntered Wrong age is [-10]
Enter father age
40
Enter son age
10
Father age positive
Son age positive
Father-Son age correct
PS C:\Users\Admin\Desktop\1BM21CS246\246_java>
```