Week 6

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.

```
import java.util.Scanner;
class WrongAgeException extends Exception{
	public String toString(){
		return ("Age cannot be negetive");
	}
}

class AgeException extends Exception{
	public String toString(){
		return("Age of son cannot be greater than Father's age");
	}
}

class Father{
	int father_age;
	Father(int x) throws WrongAgeException{
	father_age=x;
		if(father_age<0){
			throw new WrongAgeException();
	}
   }
}

class Son extends Father{
	int son_age;
	Son(int x,int y) throws AgeException, WrongAgeException{
		super(x);
		son_age=y;
		if(son_age<0){
			throw new WrongAgeException();
		}
		if(son_age>=father_age){
			throw new AgeException();
      }
```

```
    }
}


class Lab_7{
        public static void main(String[] args) {
                try {
                        Scanner s=new Scanner(System.in);
                        System.out.println("Enter father's  age and Son's age : ");
                int x=s.nextInt();
                        int y=s.nextInt();
                        Son so=new Son(x,y);
                        System.out.println("Father is " + so.father_age + " years old and son is " +
so.son_age + " years old");
      }
                catch (WrongAgeException wa) {
                        System.out.println(wa);
                }
                catch (AgeException a){
                        System.out.println(a);
                }
                catch (Exception e){
                        System.out.println("Enter valid values of age");
      }
    }
}
```

Output

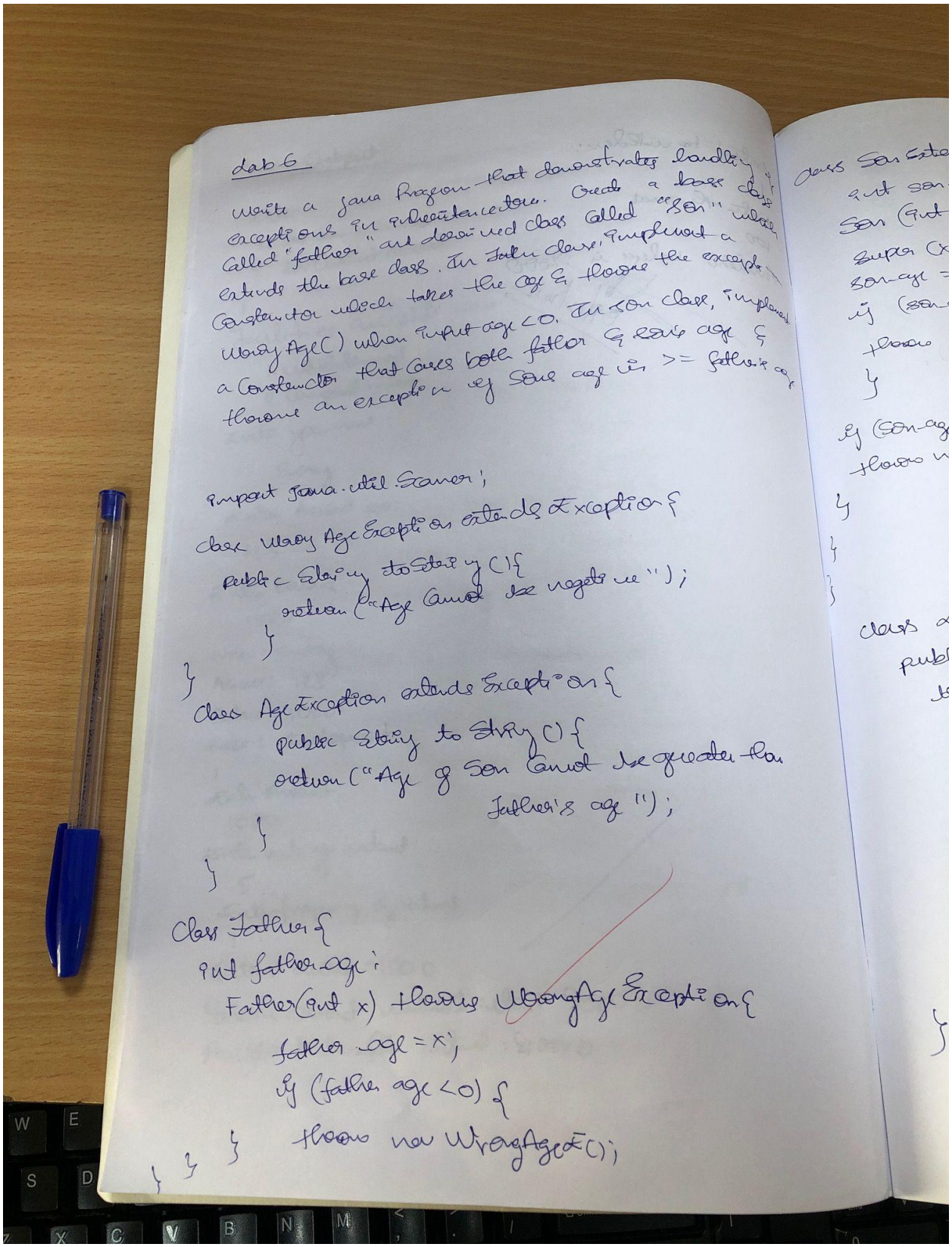
Command Prompt

```
C:\Users\student\Desktop\ooj 1BM21Cs037\week_6>java Lab_7
Enter father's  age and Son's age :
10 100
Age of son cannot be greater than Father's age

C:\Users\student\Desktop\ooj 1BM21Cs037\week_6>java Lab_7
Enter father's  age and Son's age :
-10 -5
Age cannot be negetive
```

```
Enter father's  age and Son's age :
100 50
Father is 100 years old and son is 50 years old
```

Observation



Lab 6

Write a Java 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 & throws the exception WrongAge() when input age < 0. In son class, implement a constructor that calls both father & son's age & throws an exception if son's age is >= father's age

```
import java.util.Scanner;
class WrongAgeException extends Exception{
    public String toString(){
        return("Age cannot be negative");
    }
}
class AgeException extends Exception{
    public String toString(){
    return("Age of son cannot be greater than
                    father's age");
    }
}
Class Father{
  int father_age;
    Father(int x) throws WrongAgeException{
        father_age=x;
        if (father_age<0){
    } } } throw new WrongAgeE();
```

handling
base class
"Son" which
used a
the exceptions
base, implement
age {
= father's age

than

```
class Son extends Father {
    int son_age;
    Son (int x, int y) throws AgeException, WrongAgeException {
    super (x);
    son_age = y;
    if (son_age < 0){
    throw new WrongAgeException();
    }

if (son_age >= father_age){
throw new AgeException();

}

}

}

class Lab6{
  public static void main (String [] args) {
    try {
        Scanner S= new Scanner (System.in);
        System.out.println("Enter father & Son's age :");
        int x = S.nextInt();
        int y = S.nextInt();
        Son So = new Son (x,y);
        System.out.println("Father's age is "+ So.father_age+
        "years old and Son's age is "+ So.son_age +" years");

    }
```

```
catch (WrongAgeException wa){
    System.out.println(wa);
}
catch (AgeException a){
System.out.println(a);
}
Catch (Exception e){
System.out.println("Enter valid value of age");
}
}
}
```

Output

Enter father's age and son's age:
10 100
Age of son cannot be greater than father's age
Enter father's age and son's age:
-10 -5
Age cannot be negative
Enter father's age and son's age
100 50
Father is 100 years old and son is 50 years old.

Done
30-12-22