

LAB - PRGM - 7

FATHER AND SON - AGE

EXCEPTION

Program :- Throw exceptions if father age < 0 & son age \geq father age with son's class inheriting Father class :-

PRGM: import java.util.Scanner;

```
class WrongAge extends Exception
{
    public WrongAge (String str)
    {
        super (str);
    }
}
```

```
class Father
{
    int fatherAge;
    Father (int fatherAge) throws WrongAge
    {
        this.fatherAge = fatherAge;
        if (fatherAge  $< 0$ )
        {
            throw new WrongAge ("Father's Age cannot be less than zero");
        }
    }
}
```

class Son extends Father

{

int sonAge;

Son(int fatherAge, int sonAge) throws WrongAge

{

super(fatherAge);

this.sonAge = sonAge;

if (sonAge < 0)

{

throw new WrongAge("son's age cannot be less than zero");

}

if (sonAge >= fatherAge)

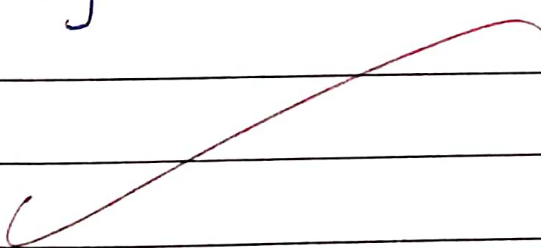
{

throw new WrongAge("son's age cannot be greater than or equal to father's age");

}

}

}



```
class AgeException Main  
{
```

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

```
        Scanner input = new Scanner (Scanner.in);
```

```
        System.out.print ("Enter Father's age : ");
```

```
        int fatherAge = input.nextInt();
```

```
        System.out.print ("Enter son's Age : ");
```

```
        int sonAge = input.nextInt();
```

```
        try {
```

```
            Son son = new Son (fatherAge, sonAge);
```

```
            System.out.println ("Father's and son's age  
                                are valid");
```

```
        } catch (WrongAge e)  
        {
```

```
            System.out.println ("Exception : " + e);
```

```
        }
```

```
    }
```

```
}
```


OUTPUT :

① Enter father's age : 50

Enter son's age : 20

Father's and son's age are valid

② Enter father's age : 50

Enter son's age : 60

Exception : Wrong Age : son's age cannot be greater than or equal to Father's age

③ Enter father's age : -1

Enter son's age : 20

Exception : Wrong Age : Father's age cannot be less than zero

```
import java.util.Scanner;
```

```
class WrongAge extends Exception
```

```
{
```

```
public WrongAge(String str)
```

```
{
```

```
super(str);
```

```
}
```

```
}
```

```
class Father
```

```
{
```

```
int fatherAge;
```

```
Father(int fatherAge) throws WrongAge
```

```
{
```

```
this.fatherAge = fatherAge;
```

```
if(fatherAge<0)
```

```
throw new WrongAge("Father's Age cannot be less than zero");
```

```
}
```

```
}
```

```
class Son extends Father
```

```
{
```

```
int sonAge;
```

```
Son(int fatherAge,int sonAge) throws WrongAge
```

```
{
```

```
super(fatherAge);
```

```
this.sonAge = sonAge;
```

```
if(sonAge<0)
throw new WrongAge("Son's Age cannot be less than zero");

if(sonAge>=fatherAge)
throw new WrongAge("Son's age cannot be greater than or equal to Father's age");

}

}

class AgeExceptionMain
{
public static void main(String[] args)
{
Scanner input = new Scanner(System.in);

System.out.print("Enter father's age: ");
int fatherAge = input.nextInt();

System.out.print("Enter son's age: ");
int sonAge = input.nextInt();

try{
Son son = new Son(fatherAge,sonAge);
System.out.println("Father's and Son's age are valid");

}catch(WrongAge e){
System.out.println("Exception: " + e);
}

}

}
```

Output:

Enter father's age: 50

Enter son's age: 20

Father's and Son's age are valid

Enter father's age: 50

Enter son's age: 60

Exception: WrongAge: Son's age cannot be greater than or equal to Father's age

Enter father's age: -1

Enter son's age: 20

Exception: WrongAge: Father's Age cannot be less than zero

Enter father's age: 0

Enter son's age: 20

Exception: WrongAge: Son's age cannot be greater than or equal to Father's age

Enter father's age: 0

Enter son's age: -1

Exception: WrongAge: Son's Age cannot be less than zero