21/11/24. O White a Program that demonstrates handling of exceptions in inheritance fue Create abose class called Jather and a deviced class called "Son' which extends the base class. In Hather class, implement a constructor which takes the age and Knows the exception Wrong Age () when the infut age 20. In the Son Class, implement a Constructor that uses both father and son's age and throws an exception if son's age is > = father is ago. Class Wrong Age Exception & Xtends Exception & fullic String to String () Return "le long Age Exception"; Class Jather 5 Father (ine age) throws WongAge Exception & throw new WrongAge Exception this age = age;

cine getAge()} System our problem " father's age fraget 3500; Seturnage; Son S. System out puntle ("Enter faker's og:"); in father Age = Scanner next toll; Father | - new Father (father Age); System our pinoler "Takka sage!" + Class Son extends Father Higel Age(1). & int San Son Age; System out print ("Enter Son's age"); Son (int father Age, but son Age) Knows Super (Jahren Age);

If (son Age CO 11 son Age > = father Age) { Son S1 = new Son (Jahrstye, Son Aze).
System. our plusely ("Son's Aze!" +
S1-get Son Aze(1). throw new Wrong Age Exception (); Hun sonAge = son Age; Catch (Whom Age Ex Ception e) of System but purely ("Ex ception (augle "+ 0). Ant get Son Age (15 4 leturn lon Age; Output Enter faller 18 age: 12 Enter Son Rage: 34 public class ExceptionHand ling Inheritance of Exception Chight: Wang Age Cx Capto public static word main (string () args? Father's age: 12
Father's age: 12
Enter son't age: -23 fly of got gather for sales Exception Caugho: Wrong Age Exception Father (1940);

Enter father'x age: -33
Ex Cleption Caught: Whongstyr Ex Ception 4: = very Falty (last Son SI = new Som (lationer Sons

```
import java.util.Scanner;
|class WrongAgeException extends Exception {
   public String toString() {
      return "WrongAgeException";
}
class Father {
         int age;
         Father(int age) throws WrongAgeException {
   if (age < 0) {
      throw new WrongAgeException();
   }
                 }
this.age = age;
        int getAge() {
    return age;
class Son extends Father {
  int sonAge;
         Son(int fatherAge, int sonAge) throws WrongAgeException {
   super(fatherAge);
   if (sonAge < 0 || sonAge >= fatherAge) {
        throw new WrongAgeException();
   }
                 }
this.sonAge = sonAge;
         int getSonAge() {
    return sonAge;
public class lab7 {
   public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
                 try {
                          System.out.print("Enter Father's age: ");
int fatherAge = scanner.nextInt();
                          Father f1 = new Father(fatherAge);
System.out.println("Father's age: " + f1.getAge());
System.out.print("Enter Son's age: ");
int sonAge = scanner.nextInt();
                          Son s1 = new Son(fatherAge, sonAge);
System.out.println("Son's age: " + s1.getSonAge());
                 } catch (WrongAgeException e) {
    System.out.println("Exception caught: " + e);
} finally {
    scanner.close();
1 1 1
```

C:\1bm23cs299>java lab7 Enter Father's age: 12

Father's age: 12 Enter Son's age: 34

Exception caught: WrongAgeException

C:\1bm23cs299>java lab7 Enter Father's age: 12

Father's age: 12 Enter Son's age: -23 Exception caught: WrongAgeException

C:\1bm23cs299>java lab7 Enter Father's age: -33

Exception caught: WrongAgeException