SRI VENKATESA PERUMAL
COLLEGE OF ENGINEERING AND TECHNOLOGY
|AUTONOMOUS|ACCREDITED BY NAAC|
RVS Nagar, K.N. Road, Puttur, Chittoor dist, AP.|www.svpcet.org

Department of ECE

Subject : oop’s using java **Subject code**:200E0501

Class: III YEAR I SEM

Exception Handling:

* Java Examines exceptions handling.

*A Exception handling defines denormal. Conditions of code series of Bequences at any time Process.

* A Exception handling is also defines a run time error.

*A computer languages does not support exception handling but Java supports exception handling. due to examine check and status of an errors. AnException handling, describes an Exception Condition which is occurs, a Piere of in Code.

* Exception handing can be categorized 5 blocks they are:

1. try

2.catch

3. throw

4.throws

5. Finally

* A try block defines in exception handling it monitors the program statement and it is also find the errors.

*A catch block defines in a exception.. handing it catches the exception based on rational manner.

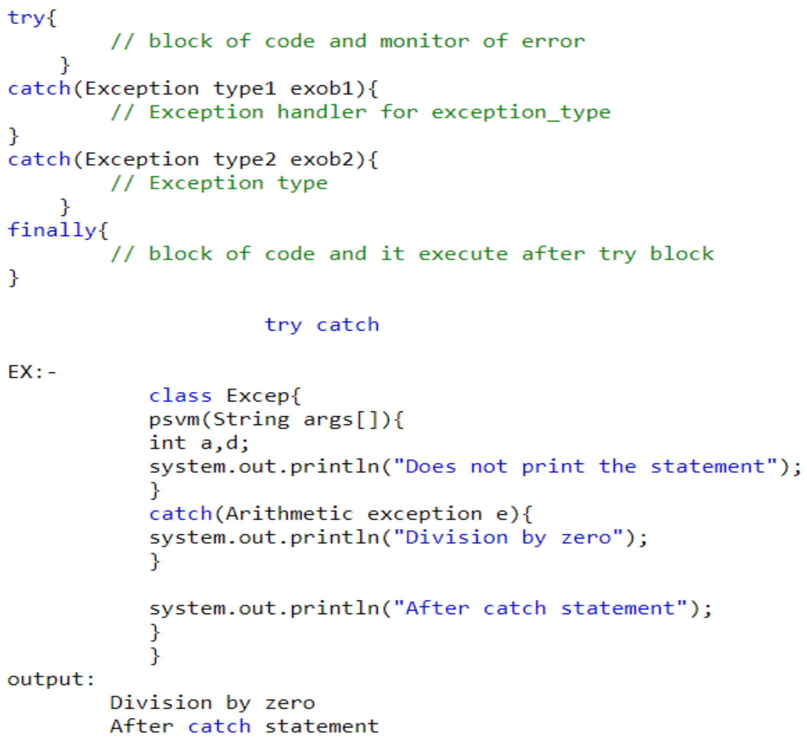
be forwarded

* A throw block defines in exception handling. When an exception can it is also referred as throw exception.

*A throws block define in exception handling it throws can be outside exception outside of method.

*A Finally block defines in exception handling which will e execute the code after try. block. that code must Part in finally block

*The general form of Exception Handling as Follows,



```
try{
        // block of code and monitor of error
    }
catch(Exception type1 exob1){
        // Exception handler for exception_type
}
catch(Exception type2 exob2){
        // Exception type
    }
finally{
        // block of code and it execute after try block
}

                    try catch

EX:-
           class Excep{
           psvm(String args[]){
           int a,d;
           system.out.println("Does not print the statement");
           }
           catch(Arithmetic exception e){
           system.out.println("Division by zero");
           }

           system.out.println("After catch statement");
           }
           }
output:
        Division by zero
        After catch statement
```

*A try and catch represents the garchan of removal and handle of evors... Simply enclose the code which is the inside of try, black.

*Immediately following a try block which" include catch classes, and it specifies the types of catch blocks.

*The following example with uses arithmetic expansion by divide by zero with try &... catch blocks. *In the above program, a println method which inside of try block & it does not displayed, any Program catch blocks suppose We place differently it means that does not call catch blocks. then the expansion not to move from try block to catch blocks. A Program must controls definetly it moves next line of programs either try or catch.

Java Built in Exception:

* All inside Packages must support standard Package name as Java.lang all languages In Java. it means functions must store a расказа A Java built in exceptions is a standard run time exception. The Vaviou Kinds of Java built in exceptions follows:

i) Arithmetic Exception as

ii) Array index out of Bound Exception.

iii) Array store exception.

iv) class cast exception

v) ehumeration constant Not: Present exception

vi) illegal argument exception Present Exception

Vii) illegal monitor exception

Viii) illegal state Exception

ix) illegal thread state exception

x) index out of bound exception.

Xi) Negative array size exception.

Xil) Null Pointer exception

Xiii) Number format Exception

XIV) Security exception

XV) string index out of Bound exception

xvi) Type No Present exception

xvii) un supported operation exception

Chained Exceptions:

1. To bring JDK 1.4 It is a special features stands for Java development kit hence it incoparate exception Subsystems.

2.This encoporate. Exception subsystem is also called as Chained Exception;

3.A chained Exception associate another Exception into exception.

4. The Second Exception is to deserbe the Cause of 1st Exception.

5.It we allow chained Exception and it represe two constructors two method. The General form of chained exception as follows:

Throwable (Throwable cause exce)

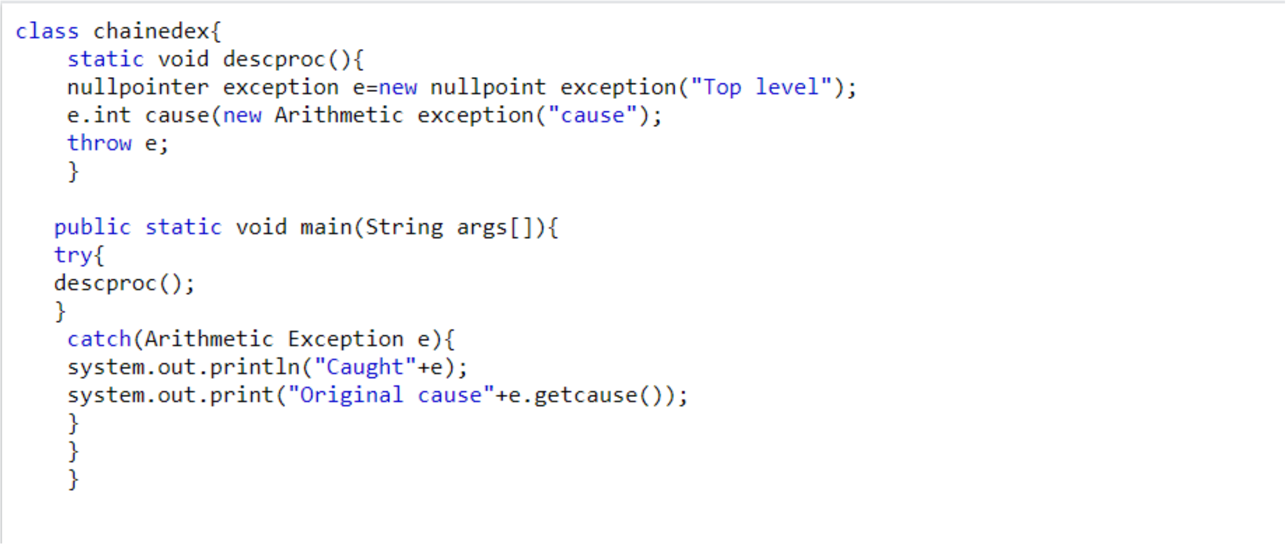
Throwable (string msg, cause exite)

6. In above general form the first form represents to describe the cause of current Exception we Can underline all exceptions must occur.

7. In the above General form second form represents at the descriptions in same time to Causes an Exception.

8. To demonstrate and illustrate to handle chained exception as follows:

EX:



```
class chainedex{
    static void descproc(){
    nullpointer exception e=new nullpoint exception("Top level");
    e.int cause(new Arithmetic exception("cause");
    throw e;
    }

   public static void main(String args[]){
   try{
   descproc();
   }
    catch(Arithmetic Exception e){
    system.out.println("Caught"+e);
    system.out.print("Original cause"+e.getcause());
    }
    }
    }
```

NOTE:-

In the above program top level is wed in Null pointer exception. and original cause is wed in arithemetic Exception.

*To Display a top level exception is caught: Java – Long.null pointer Exception Top level

*To Display a original cause is cause : Java, long- atithe metic Exception= original cause,

*TO call a came by using method is get cause().

Input/Output System:

* Infact print; printen and input/output methods are use Significantly

* all real applications in Java are not text based but we Console

* All Input/output methods Supports graphical Oriented program java Supports graphical Oriented Programs by using tool Kit called AWT. (abstract window tool kit). Hence it is a swing of interaction to the user.

* All input output method are most strongly and flexible with files and networks.

* All input output methods. Consistant. A cohesive methods and method. are cchesive and defines binding consistant definex of keeping of method.

Streams:

* A Stream. defines a flow of system.

* A Stream is. Subsystem also flow of input and output system.

* A stream defines Produce information. abstract to Consume and produce information.

* A Stream is also defines which links between Physical devices and input output subsystem.

*A Stream must access input data from disk file, keyboards and networke Socket.

* A Stream must access output data and it implement abstract which read from disk and Network connection..

* A Streame can can implement clause heirarcel Of java.io package.

Reading Console input:

* A console input accomplish to reading.. characters based on sywem.in. Where in The represe input and It occupies characters through Streams and it follows object name Called Buffer reader.

* Buffer reader supports Buffer Stream reader.

* The general form of reading console input as follows.

Buffer Reader (Reader Input Reader)

* In the above general form Buffer reader is an object, reader is an abstract clase and input reader reprecent. inStancel of Buffer reader which is created then it is also called Concrete abstract class.input reader is also defines which converts bytes to characters.

Example : - use Buffer Reader to read from console?

```
import java.io.*;
class RCJ{
public static void main(String args[])
{
BufferReader br=new BufferReader(new inputstreamReader(system.in));
system.out.println("Enter the character:"+q is quit);

du{
  c=(char)br.read();
  system.out.println(c);
  }while(c!='q')
  }
    }
   }
  }
```