Exception Handling

Types of Errors in program:

===================

1. Compile Time Error

2. Logical Error

3. Run Time Error

Exception:

---------

Exception is a run time error that can occur at run time, whenever an exception is occurs your program is abnormally terminated.

Exception Handling:

---------------------------

-> Exception Handling is a mechanism though which we continue the normal flow of the execution even an exception raised may not be raised with the following keywords

1. try

2. catch

3. finally

4. throw

5. throws

-> Generally Exceptions are in two types.

1. Checked Exception

2. Un-Checked Exception

1. Checked Exception :

=======================

-> Checked Exceptions are the exceptions that are cheked by comipler during compilation time.

-> And These Exceptions are must be caught or declared to be thrown by programmer explicitly

-> Checked Exceptions are the exceptions that are mandatory to be handled at the time writing the program,

otherwise compiler will not allow to compile the code.

Example:

1. FileNotFoundException

2. ClassNotFoundException

3. IOException

4. SQLException etc...

1.what is the output of the code below

import java.io.\*;

class printwriterexample

{

public static void main(String[] args)

{

PrintWriter writer = new PrintWriter(new File("demo.txt"));

writer.write("Welcome");

writer.flush();

writer.close();

}

}

2.What is the output of the code below

import java.io.\*;

class printwriterexample

{

public static void main(String[] args)

{

try

{

PrintWriter writer = new PrintWriter(new File("D:\\demo.txt"));

writer.write("Welcome");

writer.flush();

writer.close();

}

catch (Exception e)

{

System.out.println(e.getMessage());

}

}

}

2. Un-Checked Exception :

===================

-> Un-Checked Exceptions are the exceptions that are un- cheeked by compiler during compilation time.

-> Un-Checked Exceptions are the exceptions that are not mandatory to be handled at the time writing the program,

Example:

1. ArithmeticException

2. ArrayIndexOutOfBoundsException

3. NumberFormatException

4. OutOfMemoryError.

1.What is the output of the below code

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

2.Handling the above exception using try, catch

Import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(Exception e)

{

System.out.println(e.getMessage());

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

3.Exception-Recoverable

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(Exception e)

{

System.out.println("Denominator should not be zero");

System.out.println(10/2);

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

4.What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

5.What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(ArithmeticException a)

{

System.out.println(a.getMessage());

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

6.What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

catch(ArithmeticException a)

{

System.out.println(a.getMessage());

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

}

7.What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

try

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

Catch(Exception e)

{

System.out.println(e.getMessage());

}

}

public static void main(String args[])

{

exceptiondemo1 e=new exceptiondemo1();

e.display();

}

**}**

8.What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

public static void main(String args[])

{

exceptiondemo1 ed=new exceptiondemo1();

try{

ed.display();

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

catch(Exception e)

{

System.out.println(e.getMessage());

}

}

}

9. What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

public static void main(String args[])

{

exceptiondemo1 ed=new exceptiondemo1();

try

{

ed.display();

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

}

}

10. What is the output of the below code?

import java.io.\*;

import java.util.\*;

class exceptiondemo1

{

void display()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter a value:");

int a=sc.nextInt();

System.out.println("Enter b value:");

int b=sc.nextInt();

System.out.println(a+b);

System.out.println(a-b);

System.out.println(a\*b);

System.out.println(a/b);

}

public static void main(String args[])

{

exceptiondemo1 ed=new exceptiondemo1();

try

{

ed.display();

}

catch(ArrayIndexOutOfBoundsException ae)

{

System.out.println(ae.getMessage());

}

catch(NumberFormatException ne)

{

System.out.println(ne.getMessage());

}

catch(NullPointerException np)

{

System.out.println(np.getMessage());

}

catch(ArithmeticException a)

{

System.out.println(a.getMessage());

}

}

}

Error:

-----

-> An Error is also an exception that are occure due to lack of system resources not from the program writen by user.

-> Errors also comes under Un-Checked Exception

Example :

Error

-----

-> 1.OutOfMemoryError

-> 2.StackOverFlowError

-> 3.VirtualMachineError

etc...

Difference Between Exception and Error:

=======================================

-> Exceptions are the errors occures at run time due to program writen by the user and we can recover these exception by changing exception code with proper logic

where as Errors are also occures at run time due to lack of system resoursces and we can't recoverable because completly system faluts.

Example :

ArithmeticException (Exception)

OutOfMemoryError (Error)