Exception :-

- During the execution of Java Program, JVM faces abnormal situations based on code declaration

- If JVM faces abnormal situation then, JVM triggers an event, this event is known as 'exception'

- If exception event not generated in Java program then it results in termination of Java program

- If termination of Java program takes place then code is considered to be as non-physible code for execution.

- If any event is generated by the JVM then programmer need to handle the event so that all the lines present in the program gets executed.

eg.

```
public class Demo1
{
    psvm(String[] args)
    {
        int a = 10;
        int b = 0;
        int c = a/b;
        Sop("Result is: "+c);
    }
```

o/p

exception in thread main java.lang.ArithmeticException: / by zero at Demo1.main(Demo1.java:8)

name of exception

name of program

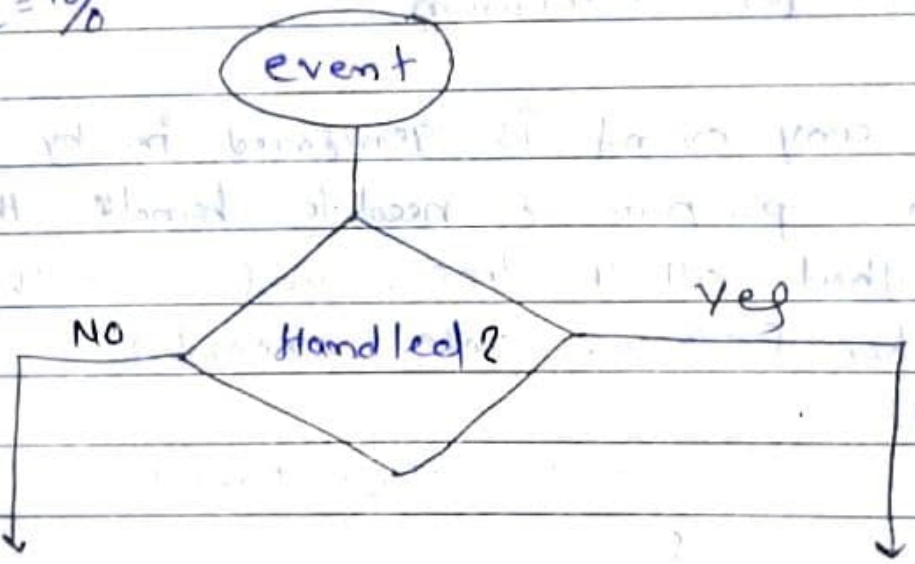
line no.

Exception handling:

Handling the event generated by Jvm during program execution is known as exception handling

- It is recommended to handle events during execution flow only

```
int a = 10/0
```



* Jvm dies, before that Jvm prints the reason for program termination

* Result in normal flow of the program
* All remaining statements are going to get executed

• To handle the event or exception, below blocks are used.

★ Syntaxe :-

```
try
{
    // Risky code
}
Catch (event/ExceptionName refvariable) Vx
{
    // Event handled message
    ;
}
```

compare

Try block :-

- It is use to declare risky code only
- Controller visits inside try block only once throughout life time of program
- try block should be followed by either "catch" block or "finally block"
- multiple try blocks are not allowed

Catch block :-

- It is use to handle the event generated from try block
- Catch block will get executed only if "event" gent generated in try block
- Catch block should be declared after try block.
- Any no. of catch block can be declared for single try block.

Date.

* finally :-

- finally is a block use to close closely resources of current program

- finally block will get executed in all the circumstances

- finally block should be followed by catch block, you can declare finally block after try block, but it is not recomended

- we can declare statements in between catch & finally block, but not recomended.

* printStackTrace();

It is an method use to get fully qualified detail of an exception

* Difference bet^n "throw" & throws'

Throw :- throw is an keyword use to throw new custom exception in current BLC.

Throws :- throws is an keyword use to show/declare type of exception generate inside method or class