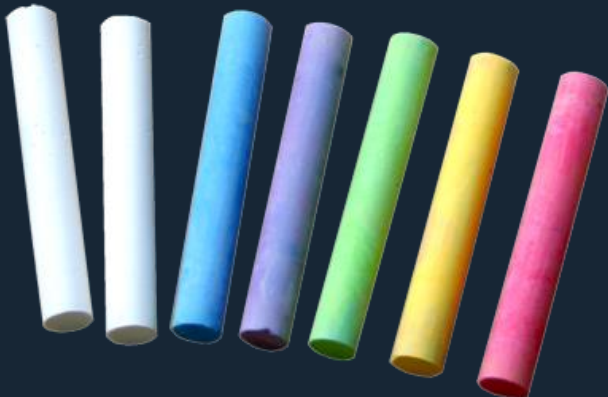


Exception Handling



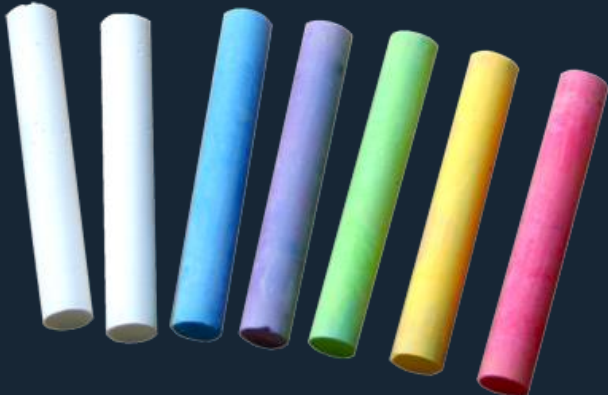
Error

- There are three types of error in program.
 1. Syntax Error
It arise when there is a mistake in syntax.
 2. Semantic Error
It leads programs producing unexpected outputs.
 3. Runtime Error
It leads to abnormal termination of programs or even cause the system to crash.



What is Exception?

- An *exception* is an *abnormal* condition that arises in a code sequence at run time.
- An occurrence of an exception disturbs the normal flow of instructions in a program.
- An exception is raised only at runtime.



What is Exception?

- During program execution, an unexpected situation such as inability to find a file or problem in network connectivity or division by zero, array index negative, etc may arise.
- Such situations are known as exceptional events, or exceptional conditions or error conditions or simply exception.



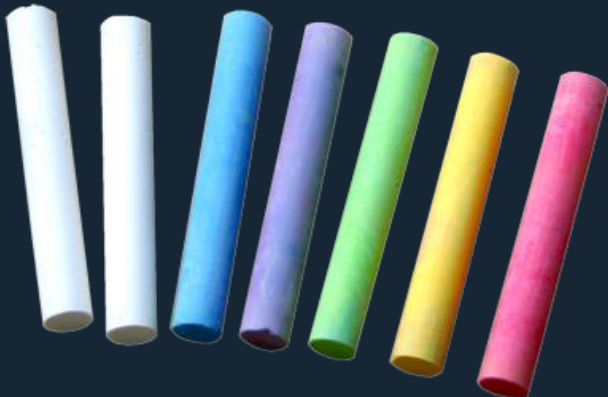
What is Exception Handler?

- Method that handle the exception is called an exception handler.

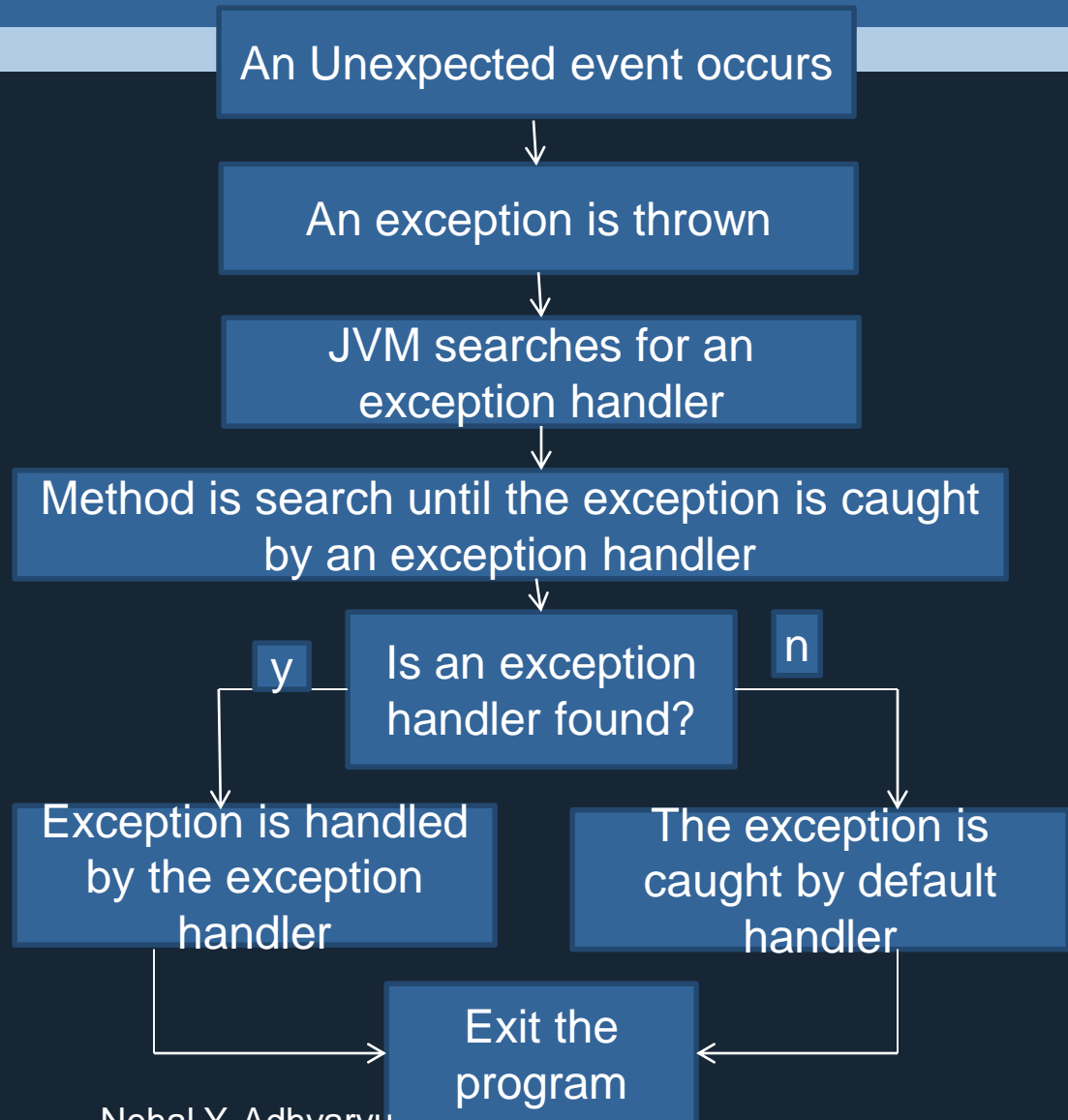
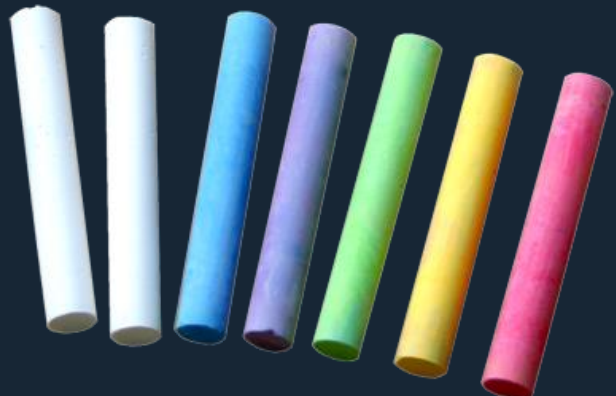


Advantage of Exception Handling

- It allow you to write software that is better organized and easier to understand.
- You no longer need to distribute code to check and handle errors throughout your program.
- It allow graceful termination of a program.

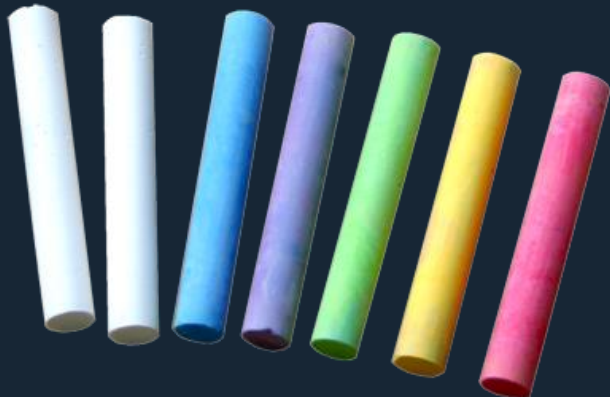


Event Handling Mechanism



Exception

- Java exception handling is managed via five keywords:
try, catch, throw, throws, and finally.
- **Try:**
Program statements that you want to monitor for exceptions are contained within a **try block**. If an **exception occurs within the try block**, it is thrown.



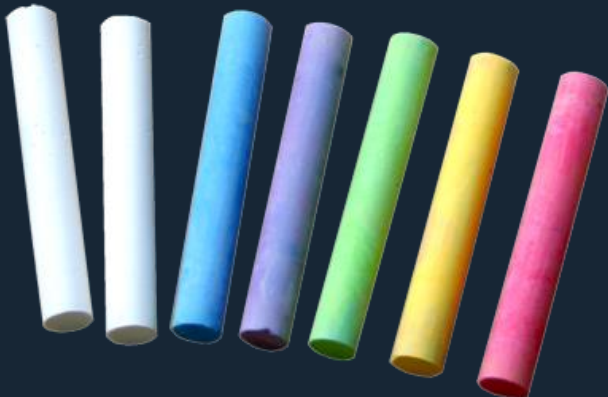
Exception

- The three possible forms of try statement are given below:
 - try-catch
 - A try block followed by one or more catch block
 - try-finally
 - A try block followed by a finally block.
 - try-catch-finally
 - A try block followed by one or more catch blocks followed by finally block



Exception

- **Catch:**
Your code can catch this exception using **catch** and handle it in some rational manner.
- **Finally:**
Any code that must be executed before termination of a program ,is put in finally block.

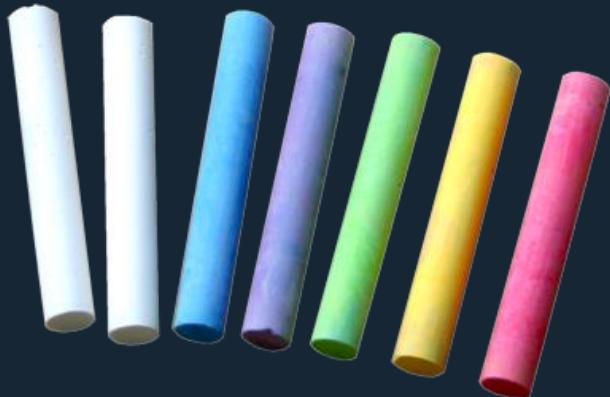


Exception

A finally block can not contain a break, continue to transfer control only within the finally block.

If the control is transferred out of the finally block by a break, continue, an error occurs.

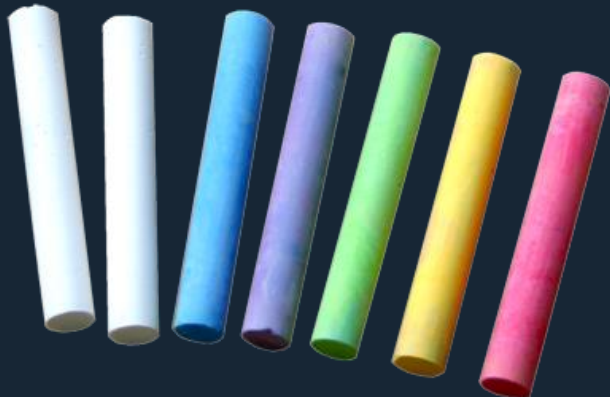
A return statement is not allowed in finally block.



Exception

- the general form of an exception-handling block:

```
try {  
    // block of code to monitor for  
    errors }  
catch (ExceptionType1 exOb) {  
    // exception handler for  
    ExceptionType1 }
```



Exception

```
catch (ExceptionType2 exOb) {  
    // exception handler for  
ExceptionType2 }  
    // ...  
finally {  
    // block of code to be executed  
    before try block ends }  
}
```

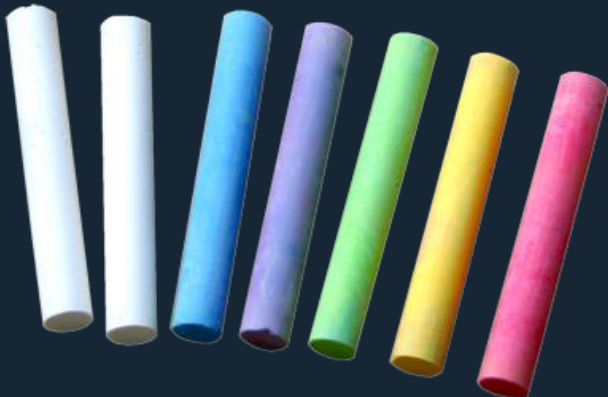
Note:

ExceptionType is the type of
exception that has occurred.



The throw statement

- With a throw statement it is possible for user explicitly throw an exception in a java program.
- Syntax:
 throw object;
- Note:
 Object must be of class type System.Exception or a class that is derived from System.Exception, otherwise otherwise it will gives you compilation error.



The throw statement

- There are two ways to throw exception:
 1. Inside a catch block, you may throw the same exception object that was provided as an argument. This can be done with the following syntax:

```
catch(ExceptionType obj) {  
    throw obj;  
}
```

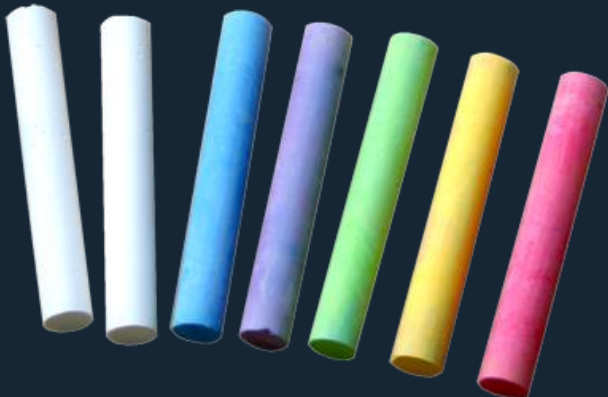


The throw statement

2. Alternatively, you may create and throw a new exception object as follows:

```
throw new ExceptionType(args);
```

Here, `ExceptionType` is the type of the exception object and `args` is the optional argument list for its constructor.



throws

- If a method is capable of causing an exception that it does not handle, it must specify this behavior so that callers of the method can guard themselves against that exception.
- User can do this by including a throws clause in the method's declaration.

