Course 3

HANDLING EXCEPTIONS IN JAVA

We handle errors using try / catch block

1) Try block
   Contains normal code to execute
2) catch block
   Contains error handling code
   receives exception information

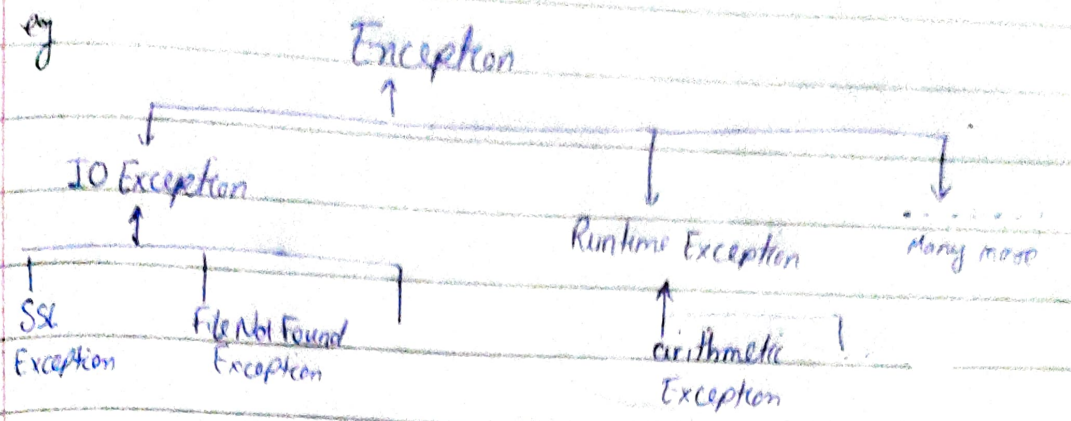
eg

```
try {
    int result = i / (j - 2);
} catch (Exception ex) {                    // when j = 2
    System.out.println("Error " + ex.getMessage());
    ex.printStackTrace();
}
```

↳ in development mode to know line & fn where error occurred

- Exceptions are represented as Class
- all exceptions have base class Exception which is further classified in various classes

eg



- So we can handle specific exceptions also
- Since it runs top to bottom, add most specific exception above general exception class

eg

```
try {
    int result = i / (j-2);
    System.out.println(result);
} catch (ArithmeticException ex){
    System.out.println("Invalid math operation");
} catch (Exception ex) {
    System.out.println("Error:" + ex.getMessage());
}
```

- Checked exception → compiler raise an error if not handled, all exceptions except runtime exceptions are checked
- Unchecked exception → compiler does not enforce handling, all runtime exceptions are ex.

Both type can crash program, if they happened.