

Using if-else avoids the need of exception handling?

No, by using if-else we cannot propagate the problem to the caller. But by using exceptional handling (try-catch) we can propagate the problem to the caller.

Eg: To withdraw money from the atm, we use atm card. To perform any operation with our account, the pin should be given. If the pin is not correct, it says please re-enter the pin.

The machine gives 3 chances to enter correct pin, even after three chances the wrong pin is entered, the account will be blocked.

here user entered wrong pin, so it can be treated as an exception, and it is propagated to the bank server and his account has been blocked. even if he uses other atm he can’t access his account.

If-else can’t propagate that message (wrong input by user for 3 times in atm) to the server, to block his account.

So if-else can’t replace exception handling. (There are some other reasons why it can’t be replaced go through the internet for extra information).

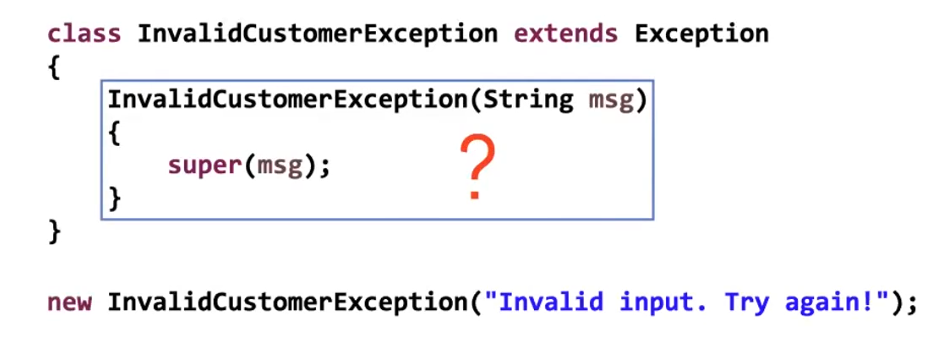
Custom Exception:

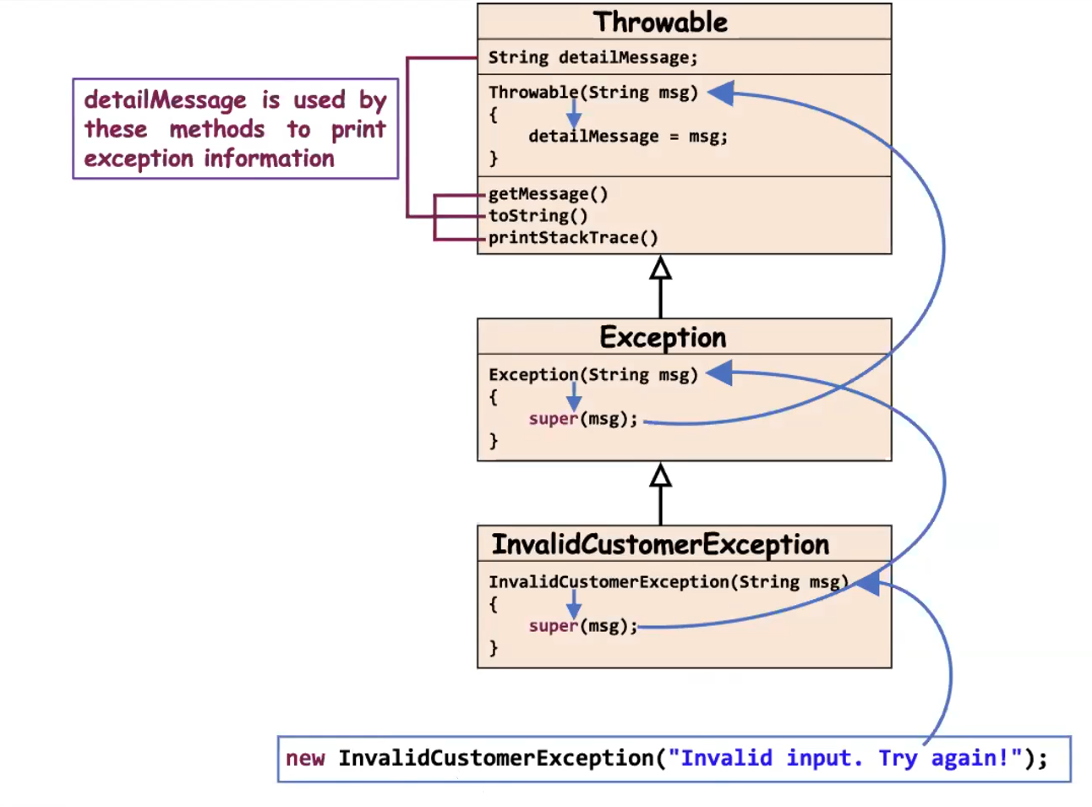
Eg: Custom\_Exceptions

To write any user defined exceptions we need to extend java Exception class to our user defined exception class.



Control flow :





Eg: Exceptional\_Handling\_Eg9

If some exception occurs during initialization then it is called ExceptionInIntializerError

Static block is used to initialize the static variables. But when arrays are declared default values are assigned , values will be assigned to it only when the object is created. Which leads to exception ExceptionInIntializerError

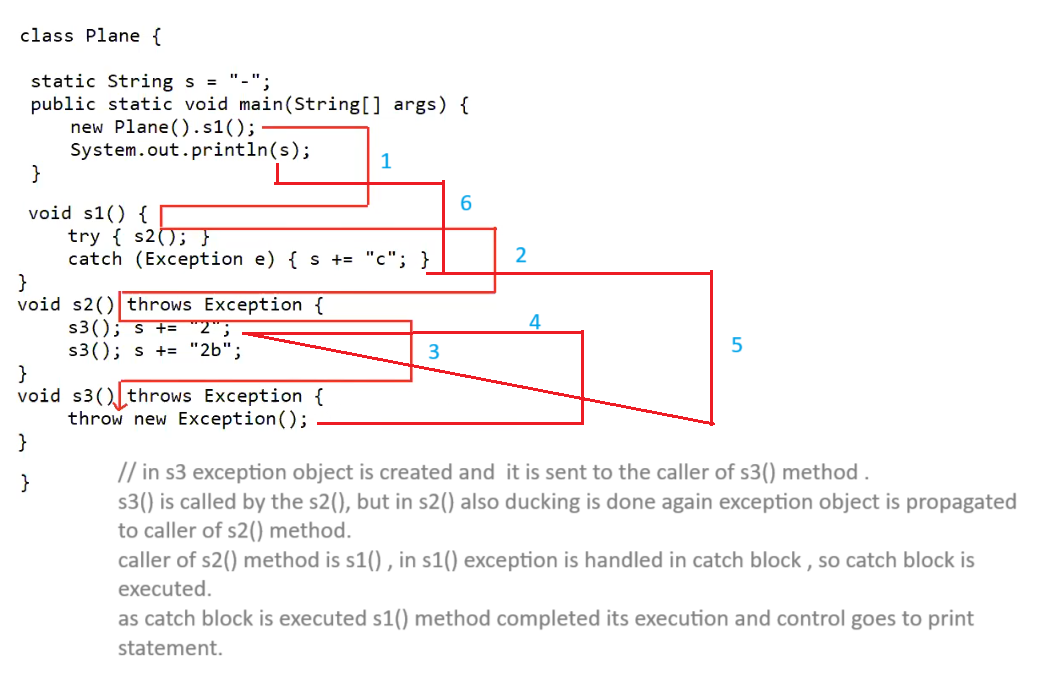
Eg: Exceptional\_Handling\_Eg10

// go through the code

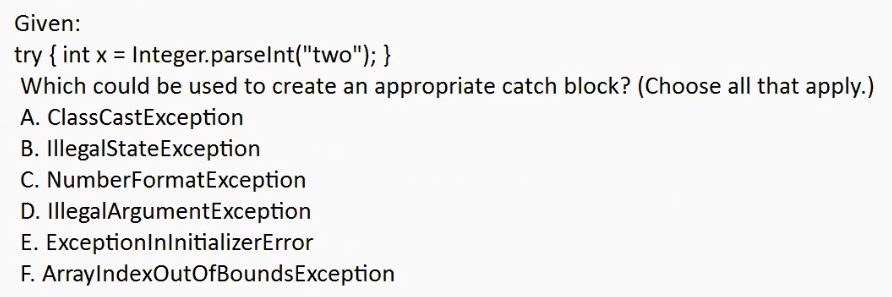
Eg: Exceptional\_Handling\_Eg11

// go through the code

Eg: Exception\_Handling\_Eg12



Note : Always use printStackTrace() method for exception.



Auto unboxing is done here Integer is stored in int.

Ans : C,D

ClassCastException

When we are performing type casting at that time if the runtime object and reference are not mapping it will lead to ClassCastException.

illegalStateException

multiple times when we try to call start() method in threads.

NumberFormatException

When you try to give String and at that time if it cannot be converted to int type we get this

IllegalArgumentException

To a parsexxx method you should provide String type argument otherwise it will lead to IllegalArgumentException

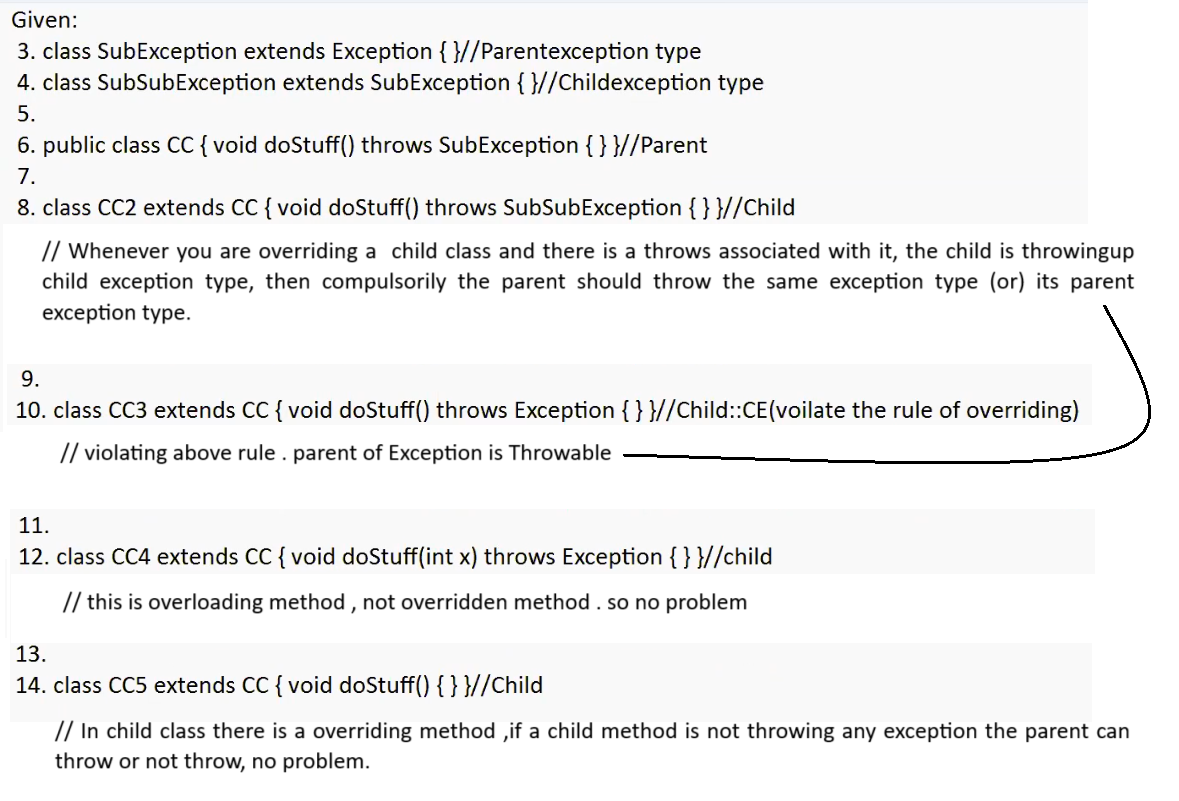
ExceptionalInIntializerError

This will come whenever we try to work with static block and there if some problem occurs.

Eg: Exceptional\_Handling\_Eg13

// go through the code

Eg: Exceptional\_Handling\_Eg14



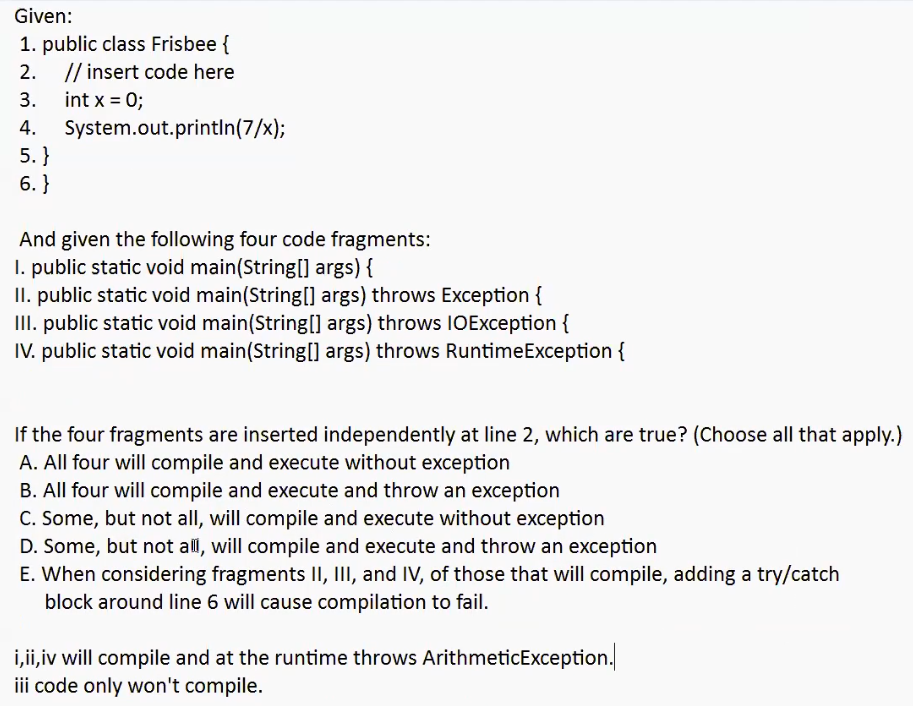
Eg: Exceptional\_Handling\_Eg15

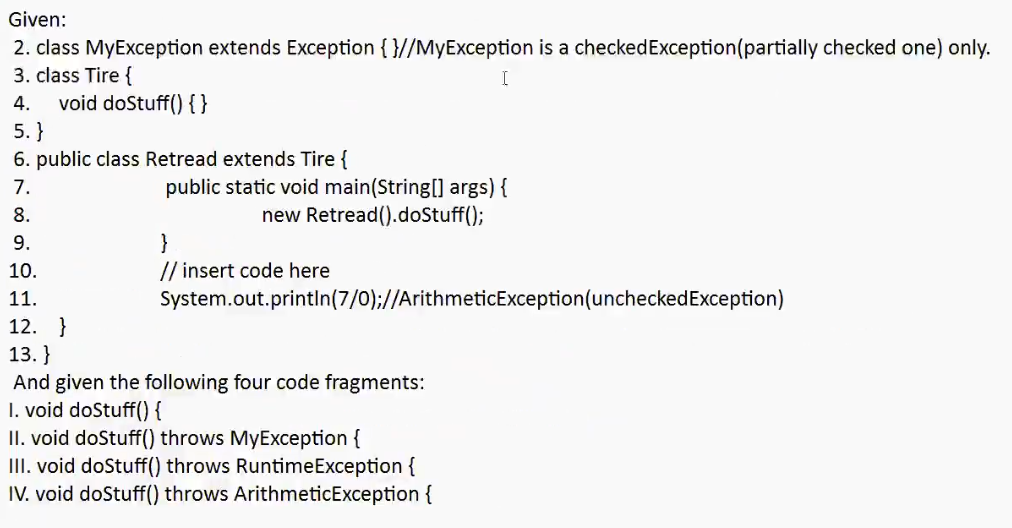
// go through the code

Eg: Exceptional\_Handling\_Eg16

//go through the code

Eg: Exceptional\_Handling\_Eg17





1. valid
2. CE // MyException is a checked one , it should be written if compiler finds the risky code . else it would result in the compile time error .
3. valid
4. valid

// Exception is a partially checked . means a checked exception where its child classes are Unchecked.

Note:

We can create the custom **unchecked exception**by extending the **RuntimeException**in Java.

**Unchecked exceptions**inherit from the **Error**class or the **RuntimeException**class

If you want to write a checked exception that is automatically enforced by the Handle or Declare Rule, you need to extend the Exception class.