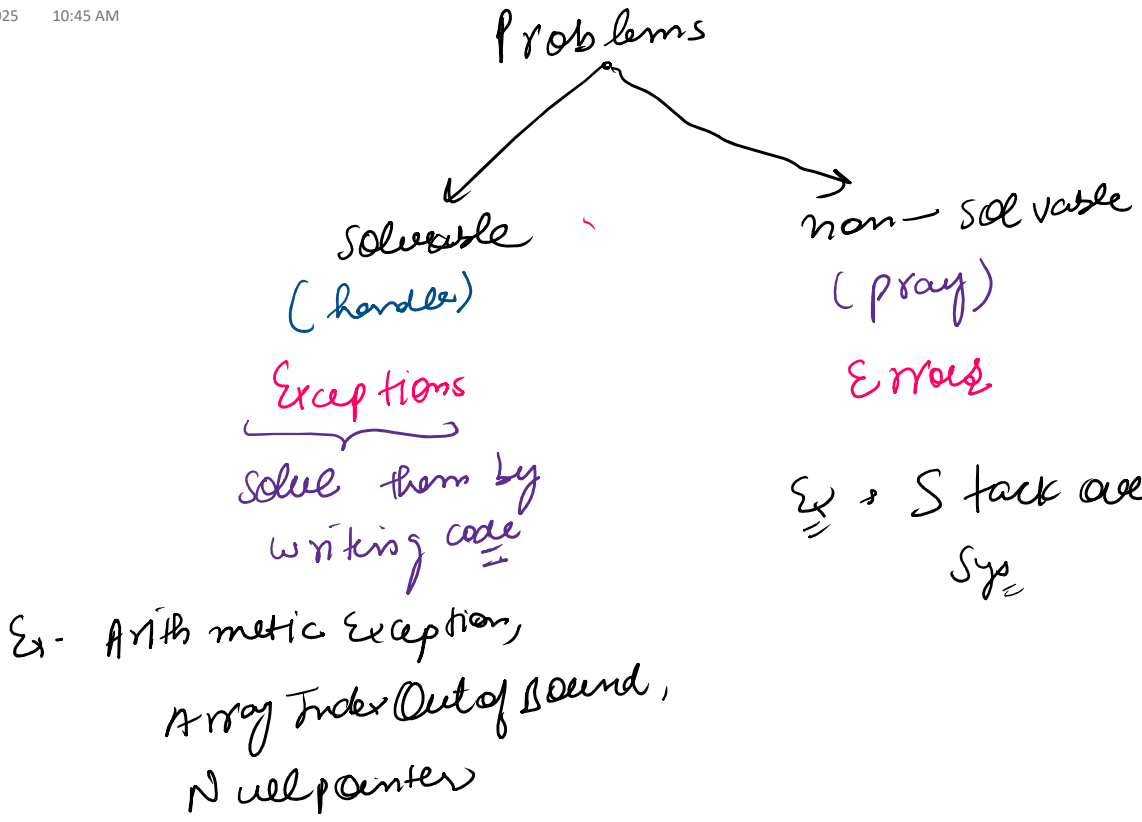
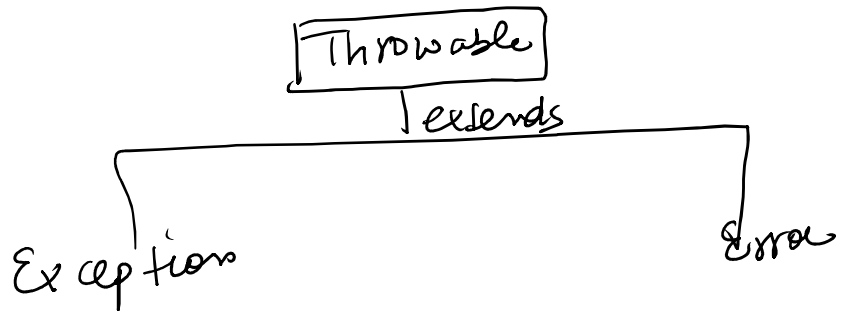
# 5.5 Exception Handling

Saturday, November 1, 2025 10:45 AM



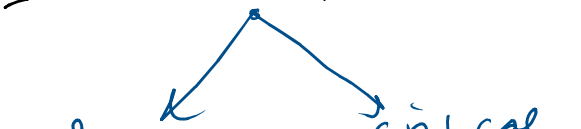




## # Exception Handling

to be handled while writing the code

Exceptions
- compile time (checked exceptions)
  - FileNotFound
  - ClassNotFound
- Runtime (unchecked exceptions)
  - issue in the logic

## # Types of statements in programming




which never fail

which might fail.

int a = 5;

int c = b/0 ] divide by zero

int n = 5+10;

[FileReader f = new C=>]

→ Critical statements upon failing throw an exception.

# why to handle exceptions

→ Because we cannot afford our code to break.

# try - catch block

→ critical statements are put inside a try - catch block

(*) A single try can have multiple catch blocks

(*) We cannot catch a parent exception before a child exception.

... have catch or finally or both.

@ a try should have catch or finally or both.

# finally block →

→ one finally is associated with one try.

→ No matter whether, try or catch executed (finally will definitely execute.

→ finally is generally used when we want to free up the resources, close connections etc.

→ one try block can have only one finally block

→ finally is only executed when the corresponding try is executed.

# throw keyword.

# throw keyword

↓

throw is used to manually throw an exception.

# Custom Exceptions

# throws

→ throws is used when we want to inform the caller / compiler that a particular method may throw this exception

- We can write multiple exception with throws key board.

- It's mandatory to write throws with compile time / checked exceptions.

- throw is only used to throw exception objects.