**Exceptional handling Assignments by gaurav diwan**

**Java with DSA and System design**

Q1] explain different types of errors in java?

Ans.

1. Compile-Time Errors: Syntax or type-related errors detected by the compiler.

2. Run-Time Errors: Errors that occur during program execution, such as division by zero.

3. Logical Errors: Errors in the program's logic that result in incorrect behavior but do not cause immediate crashes or error messages.

Q2] what is an exception in java?

Ans. An exception in Java is an abnormal event or runtime error that disrupts the normal flow of a program.

Q3] how can you handle exception in java ? explain with an example.

Ans. **Handling exceptions in Java:** Use **try**, **catch**, and **finally** blocks.

Example : try {

// code that may cause an exception

int result = 10 / 0;

} catch (ArithmeticException e) {

// handle the exception

System.out.println("Error: " + e.getMessage());

} finally {

// optional block, always executed

System.out.println("This will always be executed.");

}

Q4] why do we need exceptional handling in java?

Ans. Exception handling in Java is crucial for:

1. Robustness: Preventing program crashes due to unexpected errors.

2. Maintainability: Easier identification and resolution of issues.

3. User Experience: Providing meaningful error messages to users.

4. Program Flow: Allowing controlled handling of unexpected situations.

Q5] what is the difference between exception and error in java ?

Ans. Exception:

- Caused by undesirable conditions within a program.

- Can be caught and handled by the program.

Error:

- Caused by external factors or serious issues.

- Typically cannot be caught or handled by the program.

Q6] name different types of exceptions in java

Ans. 1. Checked Exceptions: Checked at compile-time (e.g., IOException).

2. Unchecked Exceptions (Runtime Exceptions): Not checked at compile-time (e.g., NullPointerException).

3. Errors: Irrecoverable issues (e.g., OutOfMemoryError).

Q7] can we just use try instead of catch and finally?

Ans. No, a `try` block alone is not sufficient; you must include either a `catch` or `finally` block (or both) after a `try` block in Java.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***