**Core Java - Assignment**

**Module - 1**

1. **Exception Handling :**

**Theory : -**

1. Types of Exceptions: Checked and Unchecked:

-> Exception are things that break the flow of the code. When it occurs it show some kind of error and stop the execution of further code.

Checked Exception :

-> Checked exception which occurs at compile when you run it generates byte code or you can say class file . In that file if exception is checked then it must be written try catch block which handles the run time exception if error caught because it was handled during compile time.

-> Common checked exceptions include: IOException, SQLException, ClassNotFoundException

Unchecked Exception :

-> Unchecked exceptions are exceptions that are not checked at compile-time. These exceptions are subclasses of RuntimeException. Unchecked exceptions do not need to be declared in a method's throws clause and can be caught optionally.

->Common unchecked exceptions include: NullPointerException, ArraylndexOutOfBoundsException, ArithmeticException

1. try, catch, finally, throw, throws

-> try and catch

-> The try block is used to enclose code that might throw an exception. The catch block is used to handle the exception that occurs in the try block.

-> finally

-> The finally block is used to execute important code such as closing resources, regardless of whether an exception is thrown or not. It always executes after the try and catch blocks.

-> throw

-> The throw keyword is used to explicitly throw an exception. It can be used to throw both checked and unchecked exceptions.

-> throws

-> The throws keyword is used in a method signature to declare that the method might throw one or more exceptions. It is used to propagate checked exceptions to the caller of the method.

1. Custom Exception Classes

-> In Java, you can create custom exception classes to handle specific error conditions in your application. Custom exceptions can extend either Exception (for checked exceptions) or RuntimeException (for unchecked exceptions).

-> Creating a Custom Checked Exception

-> To create a custom checked exception, extend the Exception class.

-> Creating a Custom Unchecked Exception

-> To create a custom unchecked exception, extend the RuntimeException class.