**Names: IRIZA Yvonne**

**ID: 25875**

**Assignment of oop on exceptions**

-----------

Checked Exceptions

**1.IOException**

import java.io.\*;

public class IOExceptionExample {

public static void main(String[] args) {

try {

// Attempting to write to a file in a directory that doesn't exist

FileWriter writer = new FileWriter("/invalid\_directory/output.txt");

writer.write("Testing IOException");

writer.close();

} catch (IOException e) {

// Handling the IOException when file writing fails

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

}

}

}

**2.FileNotFoundException**

import java.io.\*;

public class FileNotFoundExceptionExample {

public static void main(String[] args) {

try {

// Trying to open a non-existent file

BufferedReader reader = new BufferedReader(new FileReader("nonexistent.txt"));

System.out.println(reader.readLine());

} catch (FileNotFoundException e) {

// Handling the FileNotFoundException when the file is missing

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

} catch (IOException e) {

// Handling other IOExceptions

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

}

}

}

**3.EOFException**

import java.io.\*;

public class EOFExceptionExample {

public static void main(String[] args) {

try (DataInputStream dis = new DataInputStream(new FileInputStream("data.bin"))) {

while (true) {

// Attempting to read integers from the file until the end

System.out.println(dis.readInt());

}

} catch (EOFException e) {

// Handling EOFException when the end of the file is reached

System.out.println("EOFException occurred: End of file reached.");

} catch (IOException e) {

// Handling other IOExceptions

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

}

}

}

**4. SQLException**

import java.sql.\*;

public class SQLExceptionExample {

public static void main(String[] args) {

try {

// Connecting to a database and executing an invalid SQL query

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/testdb", "user", "pass");

Statement stmt = conn.createStatement();

stmt.executeQuery("INVALID SQL");

} catch (SQLException e) {

// Handling SQLException for database errors

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

}

}

}

**5. ClassNotFoundException**

public class ClassNotFoundExceptionExample {

public static void main(String[] args) {

try {

// Attempting to load a class that doesn't exist

Class.forName("com.example.NonExistentClass");

} catch (ClassNotFoundException e) {

// Handling ClassNotFoundException for missing classes

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

}

}

}

-

**Unchecked Exceptions**

**6.ArithmeticException**

public class ArithmeticExceptionExample {

public static void main(String[] args) {

try {

// Division by zero, which triggers ArithmeticException

int result = 100 / 0;

} catch (ArithmeticException e) {

// Handling ArithmeticException for invalid arithmetic operations

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

}

}

}

**7.NullPointerException**

public class NullPointerExceptionExample {

public static void main(String[] args) {

try {

// Null reference access triggers NullPointerException

String str = null;

str.length();

} catch (NullPointerException e) {

// Handling NullPointerException for null object access

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

}

}

}

**8.ArrayIndexOutOfBoundsException**

public class ArrayIndexOutOfBoundsExceptionExample {

public static void main(String[] args) {

try {

// Accessing an array index out of bounds

int[] numbers = {1, 2, 3};

System.out.println(numbers[5]);

} catch (ArrayIndexOutOfBoundsException e) {

// Handling ArrayIndexOutOfBoundsException for invalid index access

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

}

}

}

**9. ClassCastException**

public class ClassCastExceptionExample {

public static void main(String[] args) {

try {

// Invalid type casting triggers ClassCastException

Object obj = "Hello";

Integer num = (Integer) obj;

} catch (ClassCastException e) {

// Handling ClassCastException for incompatible type casts

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

}

}

}

**10. IllegalArgumentException**

public class IllegalArgumentExceptionExample {

public static void main(String[] args) {

try {

// Passing invalid arguments triggers IllegalArgumentException

setAge(-5);

} catch (IllegalArgumentException e) {

// Handling IllegalArgumentException for invalid arguments

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

}

}

public static void setAge(int age) {

if (age < 0) {

throw new IllegalArgumentException("Age cannot be negative.");

}

}

}

**11. NumberFormatException**

public class NumberFormatExceptionExample {

public static void main(String[] args) {

try {

// Converting invalid string format to number

int num = Integer.parseInt("invalid123");

} catch (NumberFormatException e) {

// Handling NumberFormatException for invalid numeric conversions

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

}

}

}

}