Java Assignment 3

```java
/*
    Ques : Write a program in java to handle below exceptions
            a. Divide by Zero
            b. Array Index Out Of Bound
            c. Number Format
            d. Null Pointer
*/

import java.util.Scanner;

public class Ques_1 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] array = {1, 2, 3, 4, 5};

        try {

            int result = 10 / 0;
        } catch (ArithmeticException e) {
            System.out.println("Error: Division by zero occurred");
        }

        try {
            int index = 10;
            int value = array[index];
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Error: Array index out of bounds occurred");
        }

        try {

            String str = "abc";
            int number = Integer.parseInt(str);
        } catch (NumberFormatException e) {
            System.out.println("Error: Number format exception occurred");
        }
        try {
            String str = null;
            int length = str.length();
        } catch (NullPointerException e) {
            System.out.println("Error: Null pointer exception occurred");
        }
    }
}
```

```java
/*
    Ques : Write a program in java to handle custom exception with single try block and
multiple catch block.
*/

class CustomException extends Exception {
    public CustomException(String message) {
        super(message);
    }
}

public class Ques_2 {
    public static void main(String[] args) {
        try {
            int[] array = {1, 2, 3, 4, 5};

            int result = 10 / 0; // ArithmeticException
            int index = 10;
            int value = array[index]; // ArrayIndexOutOfBoundsException
            String str = "abc";
            int number = Integer.parseInt(str); // NumberFormatException
            String nullStr = null;
            int length = nullStr.length(); // NullPointerException


            if (value < 0) {
                throw new CustomException("Negative value not allowed");
            }
        } catch (ArithmeticException e) {
            System.out.println("Error: Division by zero occurred");
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Error: Array index out of bounds occurred");
        } catch (NumberFormatException e) {
            System.out.println("Error: Number format exception occurred");
        } catch (NullPointerException e) {
            System.out.println("Error: Null pointer exception occurred");
        } catch (CustomException e) {
            System.out.println("Custom Error: " + e.getMessage());
        }
    }
}
```

```java
/*
    Ques : Write a program in java to show the use of finally keyword.
*/

public class Ques_3 {
    public static void main(String[] args) {
        try {
            System.out.println("Inside try block");
            int result = 10 / 2;
            System.out.println("Result: " + result);
        } finally {
            System.out.println("Inside finally block");
        }
        System.out.println("Outside try-finally block");
    }
}
```

```java
/*
    Ques : Write a program in java for handling exceptions with nested try block.
*/

public class Ques_4 {
    public static void main(String[] args) {
        try {

            int[] numbers = {1, 2, 3};
            int divisor = 0;

            try {

                for (int i = 0; i <= numbers.length; i++) {
                    System.out.println(numbers[i] / divisor);
                }
            } catch (ArithmeticException e) {
                System.out.println("Inner try block: Division by zero occurred");
            } finally {

                System.out.println("Inner finally block executed");
            }

        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Outer try block: Array index out of bounds occurred");
        } finally {

            System.out.println("Outer finally block executed");
        }

        System.out.println("Outside try-catch-finally block");
    }
}
```

```java
/*
    Ques : Write a program in java for custom exception to check speed of car on
            highway, if speed exceeds 120Km/hr then throw a ‘Speed Limit Exceeded’
            exception. (use throw)
*/
class SpeedLimitExceededException extends Exception {
    public SpeedLimitExceededException(String message) {
        super(message);
    }
}

class Car {
    private String carName;
    private double speed;

    public Car(String carName) {
        this.carName = carName;
    }

    public void setSpeed(double speed) throws SpeedLimitExceededException {
        if (speed > 120) {
            throw new SpeedLimitExceededException("Speed Limit Exceeded: " + speed + "
Km/hr");
        } else {
            this.speed = speed;
            System.out.println(carName + " is running at " + speed + " Km/hr");
        }
    }
}


public class Ques_5 {
    public static void main(String[] args) {
        Car car = new Car("Toyota");
        try {
            car.setSpeed(100);
            car.setSpeed(130);
        } catch (SpeedLimitExceededException e) {
            System.out.println("Caught SpeedLimitExceededException: " + e.getMessage());
        }
    }
}
```

```java
/*
    Ques : Write a program in java for handling checked exceptions using throws keyword.
*/

import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class Ques_8 {
    public static void main(String[] args) {
        try {
            readFile("nonexistent_file.txt");
        } catch (FileNotFoundException e) {
            System.out.println("File not found: " + e.getMessage());
        }
    }

    public static void readFile(String fileName) throws FileNotFoundException {
        File file = new File(fileName);

        Scanner scanner = new Scanner(file);
        while (scanner.hasNextLine()) {
            System.out.println(scanner.nextLine());
        }
        scanner.close();
    }
}
```