

## JAVA FUNDAMENTALS 6.2 PRACTICE:

**1.Describe the difference between a syntax error, a logic error, and an exception.**

**Syntax error:**

```
public class Example {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!" // Missing closing parenthesis and  
        semicolon  
    }  
}
```

**Logic error:**

```
public class Example {  
    public static void main(String[] args) {  
        int a = 5;  
        int b = 10;  
        int sum = a * b; // Logic error: should be addition instead of multiplication  
        System.out.println("Sum is: " + sum);  
    }  
}
```

**Exception:**

```
public class Example {  
    public static void main(String[] args) {  
        int[] numbers = new int[5];  
        try {  
            numbers[10] = 25; // Runtime exception:  
            ArrayIndexOutOfBoundsException
```

```

    } catch (ArrayIndexOutOfBoundsException e) {
        System.out.println("Error: " + e.getMessage());
    }
}
}

```

**2. What is wrong with the following code? It should print "Hello World!" four times to the screen.**

```

String str = "Hello World";
for(int i = 0; i < 4; i++) {
    System.out.println(str);
    str+= "!";
}

```

**Solution:**

**The correction of the above code is:**

```

public class HelloWorld {
    public static void main(String[] args) {
        String str = "Hello World";
        for (int i = 0; i < 4; i++) {
            System.out.println(str);
            str += "!";
        }
    }
}

```

**3. Describe an exception that you have experienced in your program before. Explain how it could be handled with a try/catch block of code.**

**Solution:**

```

public class ArrayExample {
    public static void main(String[] args) {
        int[] numbers = new int[5];
        try {
            numbers[10] = 25;
        } catch (ArrayIndexOutOfBoundsException e) {

```

```

        // Handling the exception
        System.out.println("Error: Array index is out of bounds.");
        System.out.println("Exception details: " + e.getMessage());
    }
}

```

**Explanation:**

- \* The array numbers has 5 elements, with valid indices ranging from 0 to 4.
- \* Attempting to access numbers[10] triggers an `ArrayIndexOutOfBoundsException` because index 10 is outside the valid range.

**4. Write a segment of code that has:**

- a. A syntax error
- b. A logic error
- c. An exception

**code segment:**

```

public class ErrorExample {
    public static void main(String[] args) {
        int num = 5;
        int result = num + ;
        int a = 10;
        int b = 20;
        int sum = a * b;
        System.out.println("Sum is: " + sum);
        int[] numbers = new int[3];
        try {
            numbers[5] = 10;
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Caught an exception: " + e.getMessage());
        }
    }
}

```

**a. syntax error:**

```
int result = num + ;
```

**b. A logic error:**

```
int sum = a * b;
```

**c. An exception:**

```
int[] numbers = new int[3];  
try {  
    numbers[5] = 10;  
} catch (ArrayIndexOutOfBoundsException e) {  
    System.out.println("Caught an exception: " + e.getMessage());  
}
```

**5. What is the difference between a checked exception and an unchecked exception?**

**Checked Exceptions:**

**Definition:** Checked exceptions are exceptions that are checked at compile-time by the Java compiler. The compiler ensures that these exceptions are either caught or declared to be thrown by the method.

**Unchecked Exceptions:**

**Definition:** Unchecked exceptions are exceptions that are not checked at compile-time. They are derived from `RuntimeException` and its subclasses. The compiler does not require explicit handling of these exceptions.