

Wednesday January 9

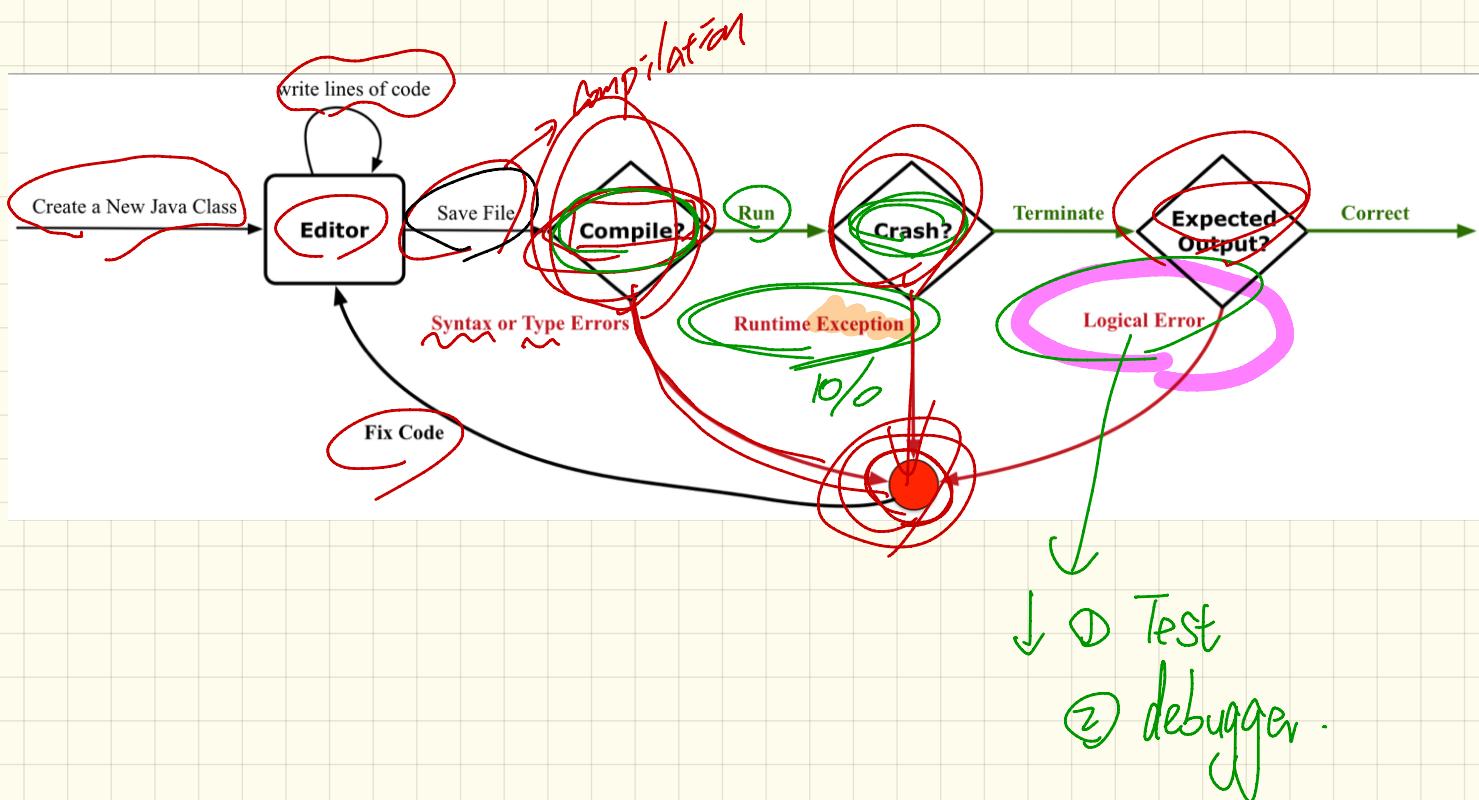
Lecture 2

- Office Hours 3pm ~ 5pm
W / F

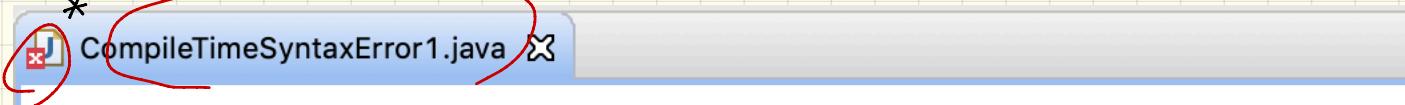
- Lab 0 Part I
W 6pm
F 5:30pm

- More exercises : Codingbat . com

Development Process



Error at the Compile Time Syntax Error (I)



```
public class CompileTimeSyntaxError1 {  
    public static void main(String[] args) {  
        // Syntax Error: missing semicolon  
        System.out.println("Hello")  
    }  
}
```

The code contains a syntax error: a missing semicolon after the string "Hello" in the println statement. The word "Syntax Error: missing semicolon" is highlighted in green. The entire code block is highlighted in pink.

Error at the Compile Time : Syntax Error (2)

The screenshot shows a Java code editor window with the following code:

```
public class CompileTimeSyntaxError2 {  
    public static void main(String[] args) {  
        // Syntax Error: missing ending double quote  
        System.out.println("Hello);  
    }  
}
```

Annotations highlight several issues:

- A red circle highlights the file icon in the title bar.
- A red circle highlights the close button in the title bar.
- A red circle highlights the file icon in the toolbar.
- A red circle highlights the close button in the toolbar.
- A red circle highlights the opening double quote " in the println statement.
- A red circle highlights the closing double quote " in the println statement.
- A red circle highlights the closing parenthesis) in the println statement.

The error message "Syntax Error: missing ending double quote" is displayed above the problematic line of code.

Error at the Compile Time : Syntax Error (3)

The screenshot shows a Java code editor window with the following code:

```
public class CompileTimeSyntaxError3 {  
    public static void main(String[] args) {  
        System.out.println("Hello");  
  
    /* Error 3: missing ending curly bracket */  
}
```

Annotations highlight several errors:

- A red circle with a black 'X' surrounds the file icon in the title bar.
- A red circle with a black 'X' surrounds the opening brace '{' after 'args'.
- A red circle with a black 'X' surrounds the closing brace '}' at the bottom of the code.
- A pink circle surrounds the closing brace '}' at the end of the class definition.
- An orange circle surrounds the word 'CompileTimeSyntaxError3' in the class declaration.
- An orange circle surrounds the opening brace '{' at the end of the class definition.

Error at the Compile Time : Syntax Error (4)

The screenshot shows a Java code editor window with the following code:

```
public class CompileTimeSyntaxError4 {
    public static void main(String[] args) {
        System.out.println("Hello");
    }
}
```

The code editor highlights several errors with colored circles:

- A red circle highlights the opening brace of the class definition: `{`.
- A blue circle highlights the opening brace of the `main` method: `{`.
- A pink circle highlights the closing brace of the `main` method: `}`.
- A green circle highlights the closing brace of the class definition: `}`.
- A red 'X' icon is visible in the top-left corner of the editor window.

A note at the bottom of the code area states: */* Error 3: extra ending curly bracket */*

Error at the Compile Time : Type Error (I)

The screenshot shows a Java code editor window with a file named "CompileTimeTypeError1.java". The code contains a class definition with a main method. A red circle highlights the file icon in the toolbar. A red arrow points from the handwritten note "type Error" to the multiplication operator (*) in the println statement.

```
public class CompileTimeTypeError1 {
    public static void main(String[] args) {
        /* Type error: Apply operator to the wrong values */
        System.out.println("York" * 23);
    }
}
```

type Error

Error at the Compile Time : Type Error (2)

The screenshot shows a Java code editor window titled "CompileTimeTypeError2.java". The code defines a class named "CompileTimeTypeError2" with a main method. The main method contains a comment indicating a type error due to an undeclared variable. The variable "i" is declared as an int and assigned the value 23. The println statement attempts to divide "i" by 3. A red arrow points from the error message "undeclared variable" to the variable "i". Red circles highlight the variable "i", the assignment operator "=", the value "23", the class name "System", the object name ".out", and the method name "println".

```
public class CompileTimeTypeError2 {
    public static void main(String[] args) {
        /* Type error: Refer to undeclared variable */
        int i = 23;
        System.out.println(i / 3);
    }
}
```

undeclared
variable

Error at the Run Time : Exception

RunTimeException.java

```
public class RunTimeException {  
    public static void main(String[] args) {  
        /* Runtime exception: code compiles but crashes at runtime */  
        System.out.println(10 / 0);  
    }  
}
```

Error at the Run Time : Logical Error

RunTimeLogicalError.java

```
import java.util.Scanner;

public class RunTimeLogicalError {
    public static void main(String[] args) {
        /* Runtime logical error: code compiles, does not crash at runtime,
         * but does not behave as expected.
         */
        → Scanner input = new Scanner(System.in);

        → System.out.println("Enter the integer radius of a circle:");
        → int radius = input.nextInt();

        → System.out.println("Area of circle is: " + (2 * 3.14 * radius));
        input.close();
    }
}
```

Document Your Code

- **Single-Lined Comments:**

```
// This is Comment 1.  
... // Some code  
// This is Comment 2.
```

- **Multiple-Lined Comments:**

```
/* This is Line 1 of Comment 1.  
 */  
... // Some code  
/* This is Line 1 of Comment 2.  
 * This is Line 2 of Comment 2.  
 * This is Line 3 of Comment 2.  
 */
```

character

~~" "~~

' - '

' a '

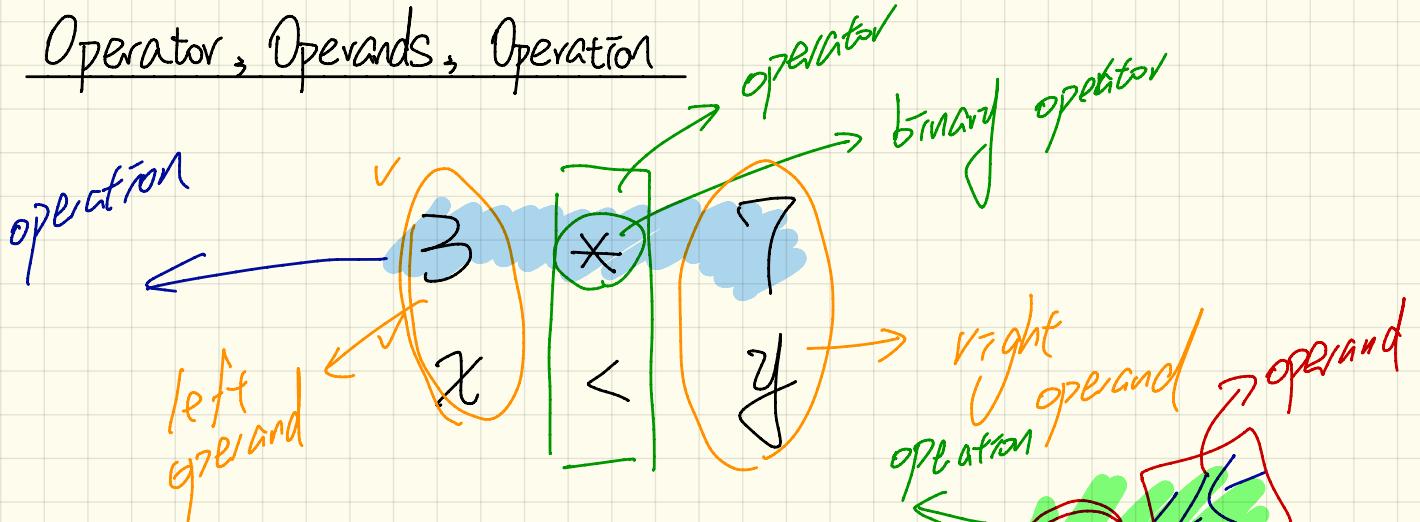
String : Seg. of char.

" " ✓ empty String

' a '

" abcd "

Operator, Operands, Operation



- ✓ - An operation consists of an operator and one or more operands.
- ✓ - An operator has one or more applicable operands.

