

Key Point: *Programming errors can be categorized into three types: syntax errors, runtime errors, and logic errors.*

1.10.1 Syntax Errors

```
1 public class ShowSyntaxErrors {  
2     public static main(String[] args) {  
3         System.out.println("Welcome to Java!");  
4     }  
5 }
```

Listing 1.4 ShowSyntaxErrors.java

```
$ javac ShowSyntaxErrors.java  
ShowSyntaxErrors.java:2: error: invalid method declaration; return type required  
    public static main(String[] args) {  
            ^  
ShowSyntaxErrors.java:3: error: unclosed string literal  
    System.out.println("Welcome to Java!");  
                        ^  
2 errors  
tt@CS:~/cs1513/java$
```

Figure 1.10 The compiler reports syntax errors.

The program has two errors as reported in Figure 1.10.

- The reserve word **void** is missing before main.
- The string **Welcome to Java!** should be closed with a closing quotation mark on line 3.

1.10.2 Runtime Errors

```
1 public class ShowRuntimeErrors {  
2     public static void main(String[] args) {  
3         System.out.println(1 / 0);  
4     }  
5 }
```

Listing 1.5 ShowRuntimeErrors.java

```
tt@CS:~/cs1513/java$ javac ShowRuntimeErrors.java  
tt@CS:~/cs1513/java$ java ShowRuntimeErrors  
Exception in thread "main" java.lang.ArithmeticException: / by zero  
    at ShowRuntimeErrors.main(ShowRuntimeErrors.java:3)
```

Figure 1.11 The runtime error causes the program to terminate abnormally.

- Observe that compiling the program ShowRuntimeErrors.java causes no error
- Note that executing ShowRuntimeErrors.java causes an ArithmeticExcetion / by zero on line 3.

1.10.3 Logic Errors

```
1 public class ShowLogicErrors {  
2     public static void main(String[] args) {  
3         System.out.print ("Celsius 34 is Fahrenheit degree ");  
4         System.out.println((9 / 5) * 35 + 32);  
4     }  
5 }
```

Listing 1.5 ShowLogicErrors.java

Celsius 35 is Fahrenheit degree 67

- But $35^{\circ}C$ is not equal to $67^{\circ}F$.
- $35^{\circ}C$ equals $95^{\circ}F$
- In Java, the division for integers is the quotient – the fractional part is truncated – so in Java $9 / 5$ is **1**. To get the correct result, you need to use $9.0 / 5$, which results in **1.8**.

1.10.4 Common Errors

- **Common Error 1: Missing Braces**

```
public class Welcome {
```

} ← Type this closing brace right away to match the opening brace

- **Common Error 2: Missing Semicolon**

```
public static void main (String[] args) {  
    System.out.println("Programming is fun!");  
    System.out.println("Fundamentals First");  
    System.out.println("Problem Driven") ← Missing a semicolon  
}
```

- **Common Error 3: Missing Quotation Marks**

```
System.out.println("Problem Driven");
```

↑ Missing a quotation mark

- **Common Error 4: Misspelling Names**

```
1 public class Test {  
2     public static void Main(string[] args) {  
3         System.out.println((10.5 + 2 * 3) / (45 - 3.5));  
4     }  
5 }
```

- Function main is not capitalized.
- Type String is capitalized.