Chapter 2: Java Fundamentals

Input and Output statements

Standard Output Window

- Using `System.out`, we can output multiple lines of text to the standard output window.

- The exact style of standard output window depends on the Java tool you use.
The println Method

- We use `println` instead of `print` to skip a line.

```java
int x = 123, y = x + x;
System.out.println("Hello, Dr. Caffeine.");
System.out.print(" x = ");
System.out.println(x);
System.out.print(" x + x = ");
System.out.println(y);
System.out.println(" THE END");
```

Standard Input

- To input primitive data values, we use the Scanner class.
- 4 steps are needed to be able to use input primitive:
  - Step 1: import the Scanner class:
    ```java
    import java.util.Scanner;
    ```
  - Step 2: declaring a reference variable of a Scanner
    ```java
    Scanner read;  //we named the object read
    ```
  - Step 3: creating an instance of the Scanner
    ```java
    read = new Scanner(System.in);
    ```
  - Step 4: use specific methods to enter data
    ```java
    int x = read.nextInt();
    ```
Example

```java
import java.util.Scanner;

Scanner input; // declaring the reference variable of a Scanner
int area, length, width; // declaring variables to store entries
input = new Scanner(System.in); // creating an instance
length = input.nextInt(); // reading the length from the keyboard
width = input.nextInt(); // reading the width from the keyboard
area = length * width; // computing the area

// displaying the result
System.out.println("the length is " + length);
System.out.println("the width is " + width);
System.out.println("the area is " + area);
```

Common Scanner Methods

• Method | Example
--- | ---
Scanner input = new Scanner(System.in); | 
nextByte( ) | byte b = input.nextByte( );
nextDouble( ) | double d = input.nextDouble( );
nextFloat( ) | float f = input.nextFloat( );
nextInt( ) | int i = input.nextInt( );
nextLong( ) | long l = input.nextLong( );
nextShort( ) | short s = input.nextShort( );
next() | String str = input.next();