

Class Declaration

```
public class CashRegister
{
    private int itemCount;
    private double totalPrice;
    // Instance variables

    public void addItem(double price)
    {
        itemCount++;
        totalPrice = totalPrice + price;
    }
    // Method
    . . .
}
```

Selected Operators and Their Precedence

(See Appendix B for the complete list.)

[]	Array element access
++ -- !	Increment, decrement, Boolean <i>not</i>
* / %	Multiplication, division, remainder
+ -	Addition, subtraction
< <= > >=	Comparisons
== !=	Equal, not equal
&&	Boolean <i>and</i>
	Boolean <i>or</i>
=	Assignment

Conditional Statement

```
    Condition
    /
if (floor >= 13)
{
    actualFloor = floor - 1;
}
// Executed when condition is true
else if (floor >= 0)
{
    actualFloor = floor;
}
// Second condition (optional)
else
{
    System.out.println("Floor negative");
}
// Executed when all conditions are false (optional)
```

Loop Statements

```
    Condition
    /
while (balance < TARGET)
{
    year++;
    balance = balance * (1 + rate / 100);
}
// Executed while condition is true
```

```
    Initialization    Condition    Update
    /      /      /
for (int i = 0; i < 10; i++)
{
    System.out.println(i);
}
```

Variable and Constant Declarations

Type	Name	Initial value
int	cansPerPack	= 6;
final double	CAN_VOLUME	= 0.335;

Method Declaration

Modifiers	Return type	Parameter type and name
public static	double	cubeVolume(double sideLength)
{		
double volume = sideLength * sideLength * sideLength;		
return volume;		
}		

Exits method and returns result.

Mathematical Operations

Math.pow(x, y)	Raising to a power x^y
Math.sqrt(x)	Square root \sqrt{x}
Math.log10(x)	Decimal log $\log_{10}(x)$
Math.abs(x)	Absolute value $ x $
Math.sin(x)	Sine, cosine, tangent of x (x in radians)
Math.cos(x)	
Math.tan(x)	

String Operations

```
String s = "Hello";
int n = s.length(); // 5
char ch = s.charAt(1); // 'e'
String t = s.substring(1, 4); // "ell"
String u = s.toUpperCase(); // "HELLO"
if (u.equals("HELLO")) ... // Use equals, not ==
for (int i = 0; i < s.length(); i++)
{
    char ch = s.charAt(i);
    Process ch
}
```

```
do
{
    System.out.print("Enter a positive integer: ");
    input = in.nextInt();
}
while (input <= 0);
```

Loop body executed at least once

Set to a new element in each iteration

```
for (double value : values)
{
    sum = sum + value;
}
```

An array or collection

Executed for each element