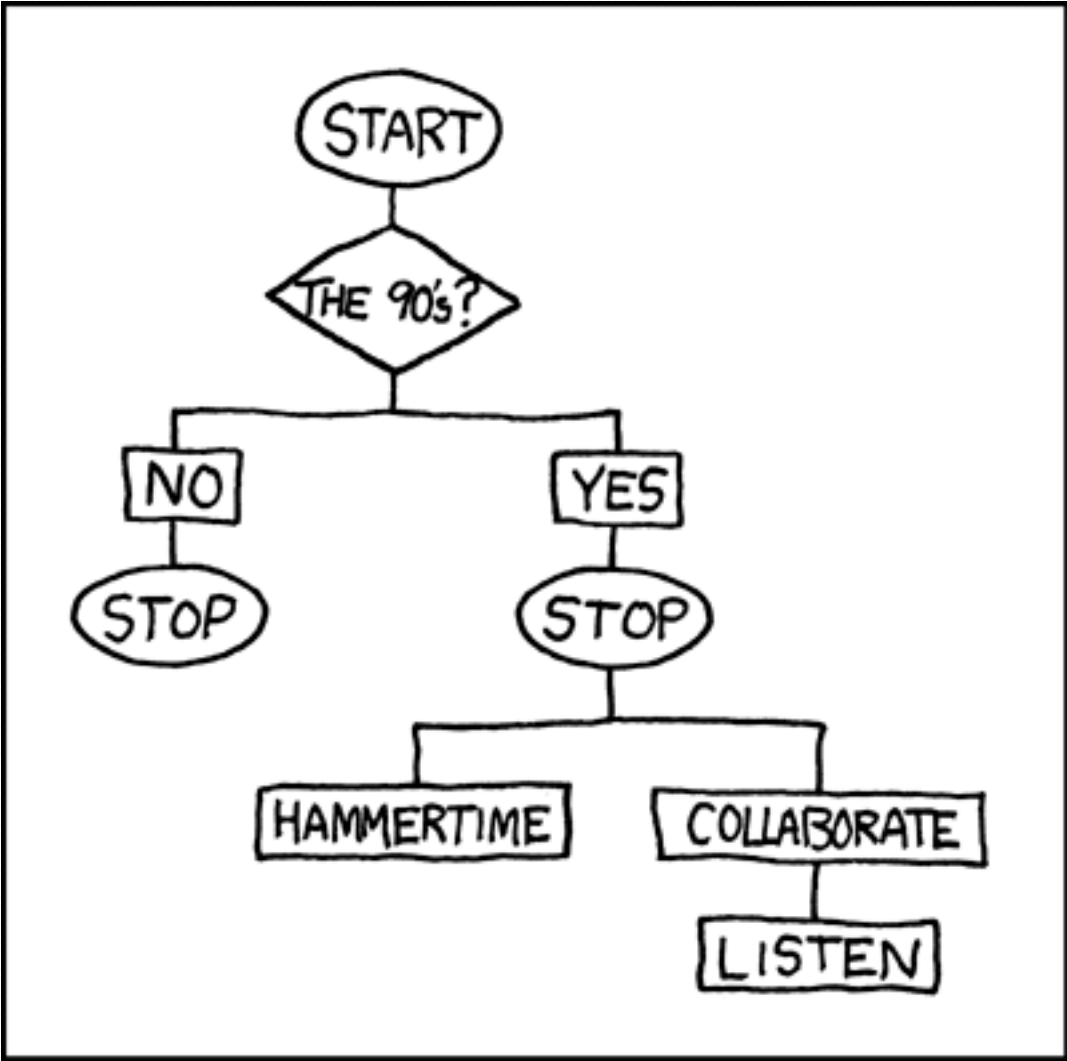


# Building Java Programs

## Chapter 4

### Lecture 4-1: `Scanner`; `if/else`

**reading: 3.3 – 3.4, 4.1, 4.5**



# Interactive Programs with Scanner

**reading: 3.3 - 3.4**

# Interactive programs

**interactive program:** Reads input from the console.

- While the program runs, it asks the user to type input.
- The input typed by the user is stored in variables in the code.
  
- Can be tricky; users are unpredictable and misbehave.
- But interactive programs have more interesting behavior.

# Scanner

- **Scanner:** An object that can read input from many sources.
  - Communicates with `System.in`
  - Can also read from files (Ch. 6), web sites, databases, ...
- The `Scanner` class is found in the `java.util` package.

```
import java.util.*;    // so you can use Scanner
```

- Constructing a `Scanner` object to read console input:

```
Scanner name = new Scanner(System.in);
```

- Example:

```
Scanner console = new Scanner(System.in);
```



# Scanner methods

Method	Description
<code>nextInt()</code>	reads an <code>int</code> from the user and returns it
<code>nextDouble()</code>	reads a <code>double</code> from the user
<code>next()</code>	reads a one-word <code>String</code> from the user
<code>nextLine()</code>	reads a <i>one-line</i> <code>String</code> from the user

- Each method waits until the user presses Enter.
- The value typed by the user is returned.

```
System.out.print("How old are you? "); // prompt
int age = console.nextInt();
System.out.println("You typed " + age);
```

- **prompt:** A message telling the user what input to type.

# Scanner example

```
import java.util.*; // so that I can use Scanner
```

```
public class UserInputExample {
```

```
    public static void main(String[] args) {
```

```
        Scanner console = new Scanner(System.in);
```

```
        → System.out.print("How old are you? ");
```

```
        → int age = console.nextInt();
```



age

```
        → int years = 65 - age;
```

years

```
        System.out.println(years + " years until retirement!");
```

```
    }
```

```
}
```

- Console (user input underlined):

```
How old are you? 29  
36 years until retirement!
```



# Scanner example 2

```
import java.util.*;    // so that I can use Scanner

public class ScannerMultiply {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);

        System.out.print("Please type two numbers: ");
        int num1 = console.nextInt();
        int num2 = console.nextInt();

        int product = num1 * num2;
        System.out.println("The product is " + product);
    }
}
```

- Output (user input underlined):

```
Please type two numbers: 8 6
The product is 48
```

- The Scanner can read multiple values from one line.



# Input tokens

- **token:** A unit of user input, as read by the `Scanner`.
  - Tokens are separated by *whitespace* (spaces, tabs, new lines).
  - How many tokens appear on the following line of input?

```
23 John Smith 42.0 "Hello world" $2.50 " 19"
```

- When a token is not the type you ask for, it crashes.

```
System.out.print("What is your age? ");  
int age = console.nextInt();
```

Output:

```
What is your age? Timmy  
java.util.InputMismatchException  
    at java.util.Scanner.next(Unknown Source)  
    at java.util.Scanner.nextInt(Unknown Source)  
    ...
```

# Scanners as parameters

- If many methods need to read input, declare a `Scanner` in `main` and pass it to the other methods as a parameter.

```
public static void main(String[] args) {  
    Scanner console = new Scanner(System.in);  
    int sum = readSum3(console);  
    System.out.println("The sum is " + sum);  
}
```

**// Prompts for 3 numbers and returns their sum.**

```
public static int readSum3(Scanner console) {  
    System.out.print("Type 3 numbers: ");  
    int num1 = console.nextInt();  
    int num2 = console.nextInt();  
    int num3 = console.nextInt();  
    return num1 + num2 + num3;  
}
```

# The `if/else` statement

**reading: 4.1, 4.6**

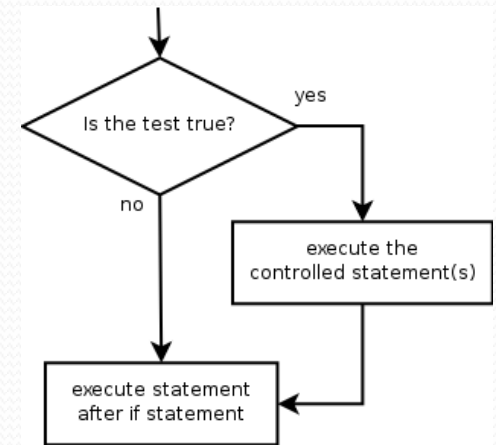
# The `if` statement

*Executes a block of statements only if a test is true*

```
if (test) {  
    statement;  
    ...  
    statement;  
}
```

- **Example:**

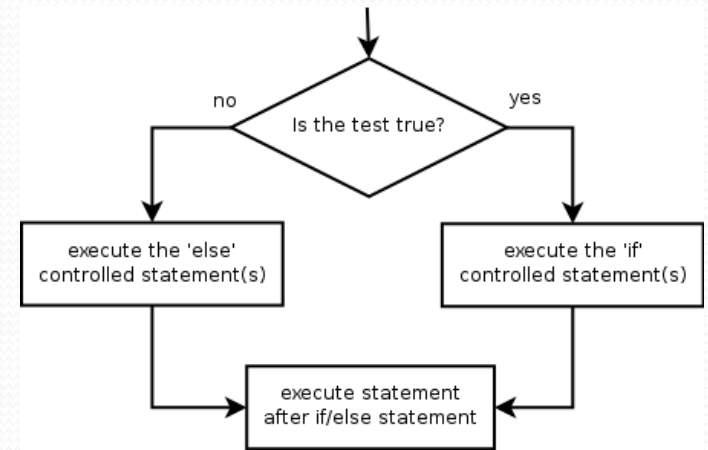
```
double gpa = console.nextDouble();  
if (gpa >= 2.0) {  
    System.out.println("Application accepted.");  
}
```



# The `if/else` statement

*Executes one block if a test is true, another if false*

```
if (test) {  
    statement(s);  
} else {  
    statement(s);  
}
```



- **Example:**

```
double gpa = console.nextDouble();  
if (gpa >= 2.0) {  
    System.out.println("Welcome to Mars University!");  
} else {  
    System.out.println("Application denied.");  
}
```

# Relational expressions

- `if` statements and `for` loops both use logical tests.

```
for (int i = 1; i <= 10; i++) { ...  
if (i <= 10) { ...
```

- These are `boolean` expressions, seen in Ch. 5.
- Tests use *relational operators*:

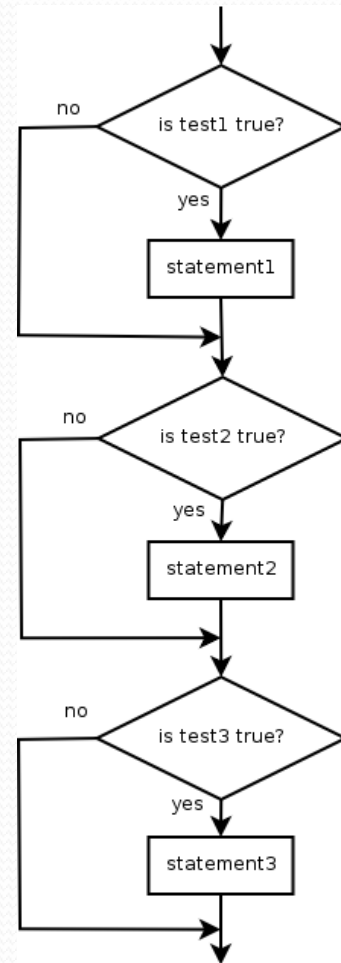
Operator	Meaning	Example	Value
<code>==</code>	equals	<code>1 + 1 == 2</code>	true
<code>!=</code>	does not equal	<code>3.2 != 2.5</code>	true
<code>&lt;</code>	less than	<code>10 &lt; 5</code>	false
<code>&gt;</code>	greater than	<code>10 &gt; 5</code>	true
<code>&lt;=</code>	less than or equal to	<code>126 &lt;= 100</code>	false
<code>&gt;=</code>	greater than or equal to	<code>5.0 &gt;= 5.0</code>	true



# Misuse of `if`

- What's wrong with the following code?

```
Scanner console = new Scanner(System.in);
System.out.print("What percentage did you
earn? ");
int percent = console.nextInt();
if (percent >= 90) {
    System.out.println("You got an A!");
}
if (percent >= 80) {
    System.out.println("You got a B!");
}
if (percent >= 70) {
    System.out.println("You got a C!");
}
if (percent >= 60) {
    System.out.println("You got a D!");
}
if (percent < 60) {
    System.out.println("You got an F!");
}
```



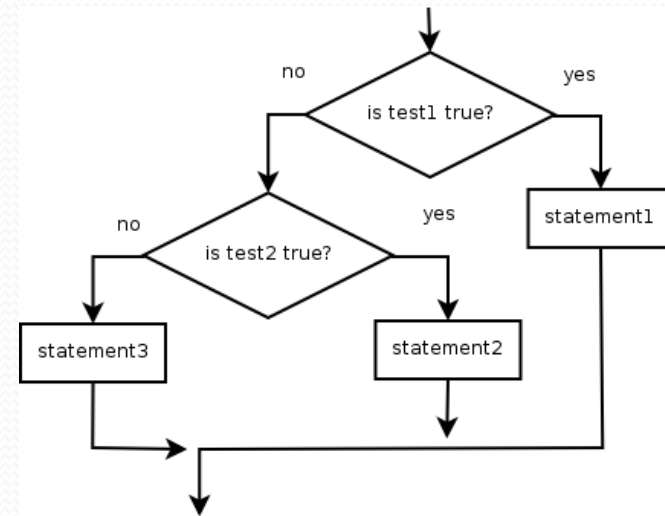
# Nested if/else

*Chooses between outcomes using many tests*

```
if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
} else {  
    statement(s);  
}
```

- Example:

```
if (x > 0) {  
    System.out.println("Positive");  
} else if (x < 0) {  
    System.out.println("Negative");  
} else {  
    System.out.println("Zero");  
}
```



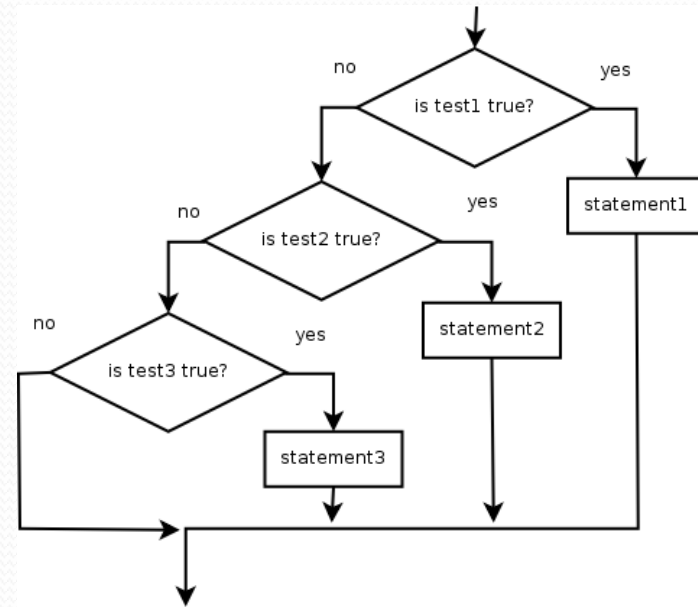
# Nested if/else/if

- If it ends with `else`, exactly one path must be taken.
- If it ends with `if`, the code might not execute any path.

```
if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
}
```

- Example:

```
if (place == 1) {  
    System.out.println("Gold medal!");  
} else if (place == 2) {  
    System.out.println("Silver medal!");  
} else if (place == 3) {  
    System.out.println("Bronze medal.");  
}
```



# Nested `if` structures

- exactly 1 path (*mutually exclusive*)

```
if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
} else {  
    statement(s);  
}
```

- 0 or 1 path (*mutually exclusive*)

```
if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
} else if (test) {  
    statement(s);  
}
```

- 0, 1, or many paths (*independent tests; not exclusive*)

```
if (test) {  
    statement(s);  
}  
if (test) {  
    statement(s);  
}  
if (test) {  
    statement(s);  
}
```

# Which nested `if/else`?

- **(1) `if/if/if` (2) nested `if/else` (3) nested `if/else/if`**
  - Whether a user is lower, middle, or upper-class based on income.
    - **(2)** nested `if / else if / else`
  - Whether you made the dean's list ( $\text{GPA} \geq 3.8$ ) or honor roll (3.5-3.8).
    - **(3)** nested `if / else if`
  - Whether a number is divisible by 2, 3, and/or 5.
    - **(1)** sequential `if / if / if`
  - Computing a grade of A, B, C, D, or F based on a percentage.
    - **(2)** nested `if / else if / else if / else if / else`

# Nested if/else question

Formula for body mass index (BMI):

$$BMI = \frac{weight}{height^2} \times 703$$

BMI	Weight class
below 18.5	underweight
18.5 - 24.9	normal
25.0 - 29.9	overweight
30.0 and up	obese

- Write a program that produces output like the following:

This program reads data for two people and computes their body mass index (BMI).

```
Enter next person's information:  
height (in inches)? 70.0  
weight (in pounds)? 194.25
```

```
Enter next person's information:  
height (in inches)? 62.5  
weight (in pounds)? 130.5
```

```
Person 1 BMI = 27.868928571428572  
overweight  
Person 2 BMI = 23.485824  
normal  
Difference = 4.3831045714285715
```



# Nested if/else answer

```
// This program computes two people's body mass index (BMI) and  
// compares them. The code uses Scanner for input, and parameters/returns.
```

```
import java.util.*; // so that I can use Scanner
```

```
public class BMI {  
    public static void main(String[] args) {  
        introduction();  
        Scanner console = new Scanner(System.in);  
  
        double bmi1 = person(console);  
        double bmi2 = person(console);  
  
        // report overall results  
        report(1, bmi1);  
        report(2, bmi2);  
        System.out.println("Difference = " + Math.abs(bmi1 - bmi2));  
    }  
  
    // prints a welcome message explaining the program  
    public static void introduction() {  
        System.out.println("This program reads data for two people and");  
        System.out.println("computes their body mass index (BMI).");  
        System.out.println();  
    }  
    ...  
}
```

# Nested if/else, cont'd.

```
// reads information for one person, computes their BMI, and returns it
public static double person(Scanner console) {
    System.out.println("Enter next person's information:");
    System.out.print("height (in inches)? ");
    double height = console.nextDouble();

    System.out.print("weight (in pounds)? ");
    double weight = console.nextDouble();
    System.out.println();

    double bodyMass = bmi(height, weight);
    return bodyMass;
}

// Computes/returns a person's BMI based on their height and weight.
public static double bmi(double height, double weight) {
    return (weight * 703 / height / height);
}

// Outputs information about a person's BMI and weight status.
public static void report(int number, double bmi) {
    System.out.println("Person " + number + " BMI = " + bmi);
    if (bmi < 18.5) {
        System.out.println("underweight");
    } else if (bmi < 25) {
        System.out.println("normal");
    } else if (bmi < 30) {
        System.out.println("overweight");
    } else {
        System.out.println("obese");
    }
}
}
```