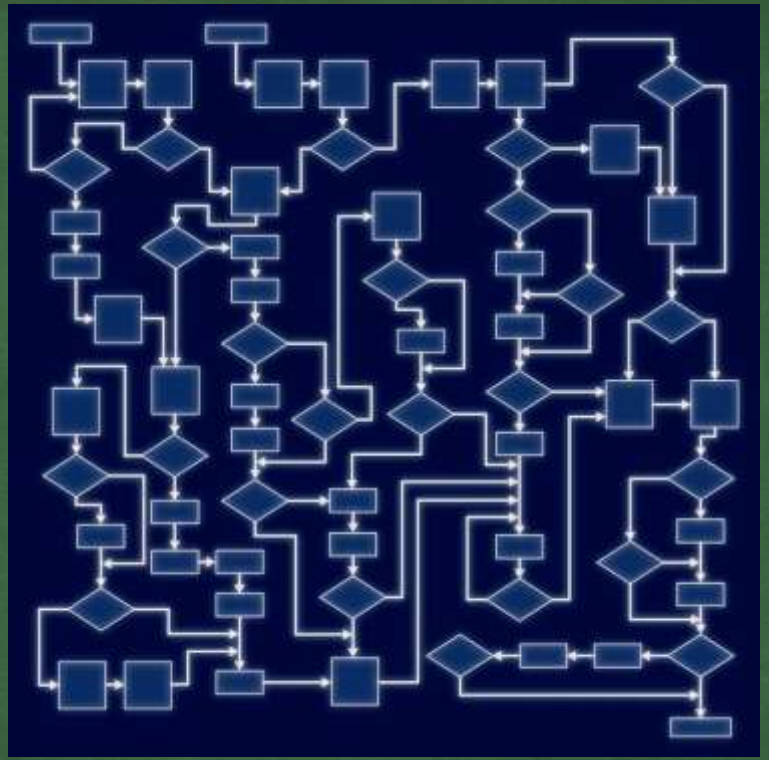
Conditionals

- Logic Operations
- **if/else**
- **switch/case**
- Scope



See Also

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html

http://www.homeandlearn.co.uk/java/java_if_else_statements.html

https://www.youtube.com/watch?v=PAaqgTr7Cx4

The Compare Operators

- Numeric comparisons
- Always result in a Boolean value (true or false)
- Usually found in flow-control expressions

```
boolean c = (a == b); // true if a and b are the same

a != b; // true if a and b are different

a < b;  // true if a is less than b
a <= b; // true if a is less than or equal to b

a > b;  // true if a is greater than b
a >= b; // true if a is greater than or equal to b
```

Logic Operations

- Booleans are NOT numbers
- They have their own operators (non numeric)
- These operations are commonly found in flow control

```
boolean a = true;
boolean b = false;

boolean c = a || b; // true if a is true OR b is true

boolean d = a && b; // true if a and b are true

boolean e = !a; // true if a is false

boolean f = !( (a || b ) && (!c) ); // can use parentheses
```

Flow Control

```
start:

if thermostat<70 GOTO here:
   TURN AC ON

here:
PRINT THERMOSTAT VALUE

GOTO start
```

Forwards: if/else

- Keyword “if” followed by expression
- One statement for the “condition pass”
- One optional statement for the “condition fail”

```
System.out.println("Begin");

if(a==2)
System.out.println("TWO");
else
System.out.println("NOT TWO");

System.out.println("Done");
```

```
if(a==2)
    System.out.println("TWO");


if(a==2) System.out.println("TWO");
```

Spacing Doesn't Matter

- Spacing helps us read the code, but the compiler ignores it
- The "print HERE" is not part of the else

```
System.out.println("Begin");

if(a==2)
    System.out.println("TWO");
else
    System.out.println("NOT");
    System.out.println("HERE");

System.out.println("Done");
```

Use Braces

- Braces group a block of statements as one.
- Spacing is for us. The compiler uses the braces.
- A good habit is to always use them.

```
if(a==2) {
    System.out.println("TWO");
} else {
    System.out.println("NOT");
    System.out.println("HERE");
}
```

```
if(a==2)
{
    System.out.println("TWO");
}
else
{
    System.out.println("NOT");
    System.out.println("HERE");
}
```

Nested if/else

- One time to NOT use the braces is with a nested "else" construct

```
if(a==2) {
    System.out.println("TWO");
} else {
    if(a==3) {
        System.out.println("THREE");
    } else {
        if(a==4) {
            System.out.println("FOUR");
        } else {
            System.out.println("NONE");
        }
    }
}
```

```
if(a==2) {
    System.out.println("TWO");
} else
    if(a==3) {
        System.out.println("THREE");
    } else
        if(a==4) {
            System.out.println("FOUR");
        } else {
            System.out.println("NONE");
        }
```

Nested if/else

- One time to NOT use the braces is with a nested "else" construct

```
if(a==2) {
    System.out.println("TWO");
} else if(a==3) {
    System.out.println("THREE");
} else if(a==4) {
    System.out.println("FOUR");
} else {
    System.out.println("NONE");
}
```

```
if(a==2)
    System.out.println("TWO");
else if(a==3)
    System.out.println("THREE");
else if(a==4)
    System.out.println("FOUR");
else
    System.out.println("NONE");
```

Ternary Operator

- Shortcut for picking between two values

```
int a;
if(i==5) {
    a = 100;
} else {
    a = 200;
}

System.out.println(a);

a = i==5 ? 100 : 200;


int min = x < y ? (x < z ? x : z) : (y < z ? y : z);
```

switch

- Use the “switch” to select one of a list of values
- Must be numeric with constant cases
- Faster than if/else with long lists

```
int a = 2;

switch(a)
{
case 2:
    System.out.println(“TWO");
case 3:
    System.out.println(“THREE");
case 4:
    System.out.println(“FOUR");
default:
    System.out.println("NONE");
}
```

Code label

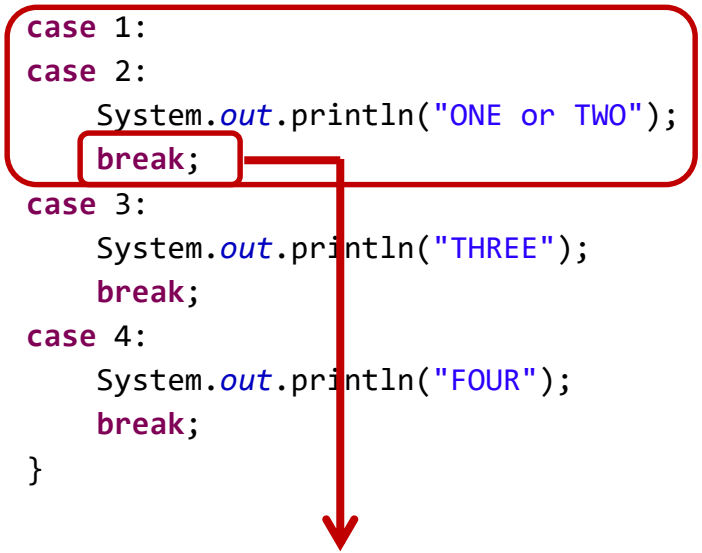
switch

- Use a “break” to end a case
- Sometimes you might want to “fall through”

```
int a = 2;

switch(a) {

case 1:
case 2:
    System.out.println("ONE or TWO");
    break;
case 3:
    System.out.println("THREE");
    break;
case 4:
    System.out.println("FOUR");
    break;
}
```



switch

- Cases must be constants
- Cases must be numeric

```
int a = 2;

switch(a) {

case 1:
case 2:
    System.out.println("ONE or TWO");
    break;
case 3:
    System.out.println("THREE");
    break;
case 4:
    System.out.println("FOUR");
    break;
}
```

String Switches

```
String a = "Apple";

switch(a) {

case "Bread":
    System.out.println("I like Bread");
    break;

case "Apple":
    System.out.println("I like Apples");
    break;

}
```

Scope

- Variables are created and destroyed with braces
- Inner scopes have access to the outer variables
- At the closing brace the variables are gone

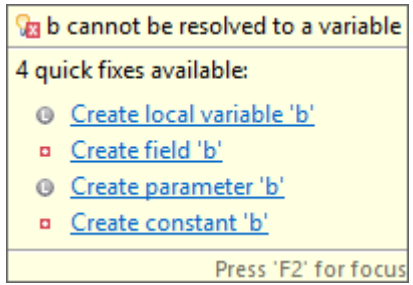
```
public static void main(String[] args) {

    int a = 10;

    if(a==10) {
        int b = a + 20;
    }

    System.out.println(a);
    System.out.println(b);

}
```



Complex Expressions

```
public static boolean checkValue(int v) {
    System.out.println(v);
    return true;
}

public static void main(String[] args) {

    int a = 10;
    int b = 20;

    if(checkValue(a) || (a==10 && b==50) || checkValue(b)) {
        System.out.println("PASS");
    }

}
```

Tinkering

- Write a function that prints “yes” if a parameter character is an uppercase letter or “no” if it is anything else.
- Write a function that prints “yes” if a parameter int is even or “no” if it odd. (Hint … divide by two and check the remainder)