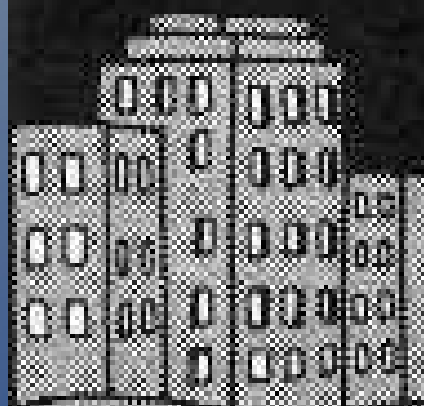


Decisions in Java

Introduction to Selection

NOTICE

You must be as
Tall as this sign
to attack the city





how tall are you?

if you are as tall as this sign,
you may attack the city

if you are not,
you cannot attack the city



Going Outside

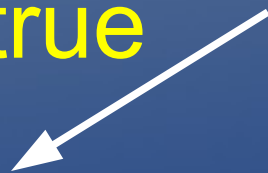


Check the
Weather



Raining?

true



Bring an
Umbrella

false



Bring
Sunscreen

What is Selection?

Selection is choosing between possible choices.
In other words, it is making a decision.

The ability to make a decision allows for programs that start to display intelligence. They are not really intelligent, of course, but sophisticated programs can appear intelligent.

Sample Program – Selection

```
class OldEnoughToVote
{
    public static void main (String [] args)
    {
        System.out.println("What is your age?");
        int age;
        age = In.getInt();

        if (age >= 18)
        {
            System.out.println("You can vote!");
        }
        else
        {
            System.out.println("You cannot vote!");
        }
    }
}
```

Comparing Values

| Relational Operator | Meaning | Example | Result |
|---------------------|-----------------------------|---------|--------|
| == | is equal to | 5 == 5 | true |
| != | is not equal to | 5 != 6 | true |
| < | is less than | 3 < 7 | true |
| <= | is less than or equal to | 4 <= 4 | true |
| > | is greater than | 3 > 7 | false |
| >= | is greater than or equal to | 7 >= 3 | true |

Comparing Characters

- it makes sense to compare the values of numbers (integers or decimals)
- it is also possible to compare characters
 - lowercase letters from 'a' to 'z'
 - uppercase letters from 'A' to 'Z'
 - characters that represents numbers '0' to '9'
 - special characters such as '+', '\$', and '@'
- every character is represented in the computer as a numeric value

Comparing Characters ASCII & Unicode

- there are two standards for representing characters as numbers
 - ASCII allows for about 8,000 options
 - Unicode allows for about 65,000 options
 - ASCII is (generally) contained within Unicode
- when two characters are compared to each other, we actually compare their numeric code

Comparing Characters ASCII & Unicode

- some examples from the Unicode table:

'A' = 65

'Z' = 90

'a' = 97

'z' = 122

'?' = 63

'0' = 48

'9' = 57