int myNumber = 10;
switch ( myNumber ) {
    case 1:
    case 2:
    case 3:
        System.out.println ( "You got 1, 2, or 3" );
        break;
    case 5:
        System.out.println ( "You got 5" );
        break;
    default:
        System.out.println ( "Some other number" );
}
int myNumber = 10;
if (myNumber == 1 || myNumber == 2 || myNumber == 3) {
    System.out.println( "You got 1, 2, or 3" );
} else if (myNumber == 5) {
    System.out.println( "You got 5" );
} else {
    System.out.println( "Some other number" );
}
The switch statement

- Tests value of expression and jumps to location in the switch statement.
- Expression is limited to certain types:
  - int, short, byte, char
  - String
  - *Cannot* be float or double
- Positions in switch are marked with *case labels* of the form case *constant:*, where *constant* is literal of same type as expression.
- Optional default: as last case.
Break statements in switch

- The `break` makes computer skip the rest of the switch statement.
- If you leave it out, computer will just fall through to next case.
- Occasionally want this (multiple cases handled same way)
Another example

```java
String computerMove;

switch ((int)(3*Math.random())) {
    case 0:
        computerMove = "Rock";
        break;
    case 1:
        computerMove = "Paper";
        break;
    case 2:
    default:
        computerMove = "Scissors";
        break;
}

System.out.println("Computer picks " + computerMove);
```