

Decision: switch

do something based on
variable or expression

```

int number //asked for one digit
           // #
switch (number) {
  case 1:
    System.out.println("One");
    break; //stop (get out of switch)
  case 2:
    S.o.println("two");
    break;
    :
  case 9:
    S.o.println("nine");
    break;
  default: //no case applies
    S.o.println("Not 1..9");
    break;
} //end switch

```

```
int number;
switch (number){
  case 3:
  case 6:
  case 9:
    s.o.pln("divisible by 3");
    break;
  case 1:
  case 2:
  case 4:
  case 5:
  case 7:
  case 8:
    s.o.pln("Not div. by 3");
    break;
  default:
    s.o.pln("Not 1..9");
}
```

do-while

```
do {  
    //body
```

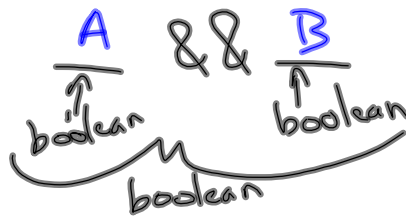
```
} while (condition);
```

```
int count = 1;  
do {  
    s.o.pln(count);  
    count++;  
} while (count <= 10);
```

boolean
values: true, false

```
boolean b = true;
s.o.pln(b); // prints True
s.o.pln(!b); // not true = false
```

AND &&



A	B	A && B
T	T	T
T	F	F
F	T	F
F	F	F

```
if (x > 0 && x < 10)
    s.o.pln("x is 1..9");
```

~~0 < x < 10~~

short circuit
if part A is false
the statement A && B is false
don't test B.

A & B - no short circuit

```
if (x != 0 && y/x < 10) {
} else {
}
}
```

Not !A

A	!A
T	F
F	T

if(! (x < 0))

same as if(x >= 0)

OR $A \parallel B$
 $\text{if}(x < 0 \parallel x > 10)$

$A \parallel B$

A	B	$A \parallel B$
T	T	T
T	F	T
F	T	T
F	F	F

XOR - exclusive OR

$A \wedge B$

A	B	$A \wedge B$
T	T	F
T	F	T
F	T	T
F	F	F