

Strutture di controllo

Modulo 226A

Implementare in base alle classi (senza ereditarietà)

If..else

```
if (condition) {  
    // block of code to be executed if the condition is true  
}
```

```
if (condition) {  
    // block of code to be executed if the condition is true  
} else {  
    // block of code to be executed if the condition is false  
}
```

```
if (condition1) {  
    // block of code to be executed if condition1 is true  
} else if (condition2) {  
    // block of code to be executed if the condition1 is false and condition2 is true  
} else {  
    // block of code to be executed if the condition1 is false and condition2 is false  
}
```

Switch

```
switch(expression) {  
  case x:  
    // code block  
    break;  
  case y:  
    // code block  
    break;  
  default:  
    // code block  
}
```

Java Loop

```
while (condition) {  
    // code block to be executed  
}
```

```
do {  
    // code block to be executed  
}  
while (condition);
```

```
for (statement 1; statement 2; statement 3) {  
    // code block to be executed  
}
```

```
for (type variableName : arrayName) {  
    // code block to be executed (Foreach)  
}
```

Break e Continue

- **Break** interrompe il ciclo
- **Continue** interrompe l'iterazione e passa alla successiva

```
for (int i = 0; i < 10; i++) {  
    if (i == 4) {  
        break;  
    }  
    System.out.println(i);  
}
```

```
for (int i = 0; i < 10; i++) {  
    if (i == 4) {  
        continue;  
    }  
    System.out.println(i);  
}
```

Blocchi di codice e *scope* delle variabili

Il termine *scope* si riferisce alla regione del programma in cui è possibile fare riferimento ad una variabile dichiarata. In altre parole, lo *scope di una variabile* è la porzione di codice in cui essa è *visibile*.

Lo *scope* di una *variabile locale* è il *blocco di codice* in cui la essa è dichiarata, a partire dalla dichiarazione e fino alla fine del blocco.

Ad esempio:

	a	b	c	d	e	f
1 public static void main(String[] args){						
2 int a = 10;	✓					
3 if(a == 10){	✓					
4 int b = 20;	✓	✓				
5 if(b == 20){	✓	✓				
6 int c = 30;	✓	✓	✓			
7 c = 40;	✓	✓	✓			
8 }else{	✓	✓				
9 int d = 50;	✓	✓		✓		
10 d = 60;	✓	✓		✓		
11 }	✓	✓				
12 }else{	✓					
13 int e = 70;	✓				✓	
14 int f = 80;	✓				✓	✓
15 e = 90;	✓				✓	✓
16 }	✓					
17 }	✓					
18 }						

Nel codice precedente lo *scope* delle variabili è rispettivamente:

- a: dalla riga 2 alla riga 17
- b: dalla riga 4 alla riga 11
- c: nelle righe 6 e 7
- d: nelle righe 9 e 10
- e: nelle righe 13, 14 e 15
- f: nelle righe 14 e 15