

Les conditions et boucles en Java

Condition if/else-if/else

Si une des condition dans les if ou else-if entre les parenthese est vrai ont s'arrete la et on rentre dans les {...} pour apres continue de execute le code, si aucune est vrai on passe au else

```
int x = 3;
```

```
if (x == 2) {  
    System.out.println("Equal to 2 (condition is true)");  
} else {  
    System.out.println("Not equal to 2 (condition is false)");  
}
```

```
int age = 30;  
if (age >= 1 && age < 18) {  
    System.out.println("You're young");  
} else if (age >= 18 && age < 65) {  
    System.out.println("You're an adult");  
} else if (age >= 65 && age <= 110) {  
    System.out.println("You're old");  
} else {  
    System.out.println("Wrong number or no valid");  
}  
  
System.out.println("Hello");
```

While

Ont exécuté le code entre les {...} pendant que la condition “()” du while est vrai, ont l'utilise souvent quand on ne sait pas combien de fois ont veut répète une execution.

```
int number = 10;  
  
while (number >= 0) {  
    System.out.println(number);  
    number--;  
}
```

```

boolean run = true;
Scanner input = new Scanner(System.in);
System.out.println("Select a number");
int choice = input.nextInt();

while (run) {
    if (choice > 0 && choice < 5) {
        System.out.println("Good choice");
        run = false;
    } else {
        System.out.println("Wrong choice select between 1 and 4");
        choice = input.nextInt();
    }
}

```

For

On exécute le code entre les (...) du for pendant que la condition du for soit true, le for se fait en quatre statement (declaration) : declaration de la variable int i = 0, condition a vérifier y <= 10, puis incrémentation de la variable i pour compléter la condition. La boucle for s'utilise quand on sait combien de fois on veut répéte un code

```

for (int x = 0; x <= 10; x++) {
    System.out.println(x);
}

for (int i = 0; i <= 10; i++) {

    for (int j = 0; j <= 10; j++) {
        System.out.print("*");
    }

    System.out.println();
}

```