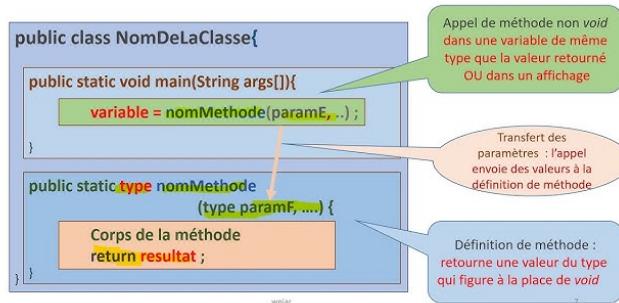


Les Appels de Méthodes en Java

Méthode *non void* en java → Fonction



En Java, les **fonctions** et les **méthodes** sont des blocs de code réutilisables qui effectuent des actions spécifiques. Une fonction en Java est en fait une méthode, car Java est un langage orienté objet, et toutes les fonctions doivent être définies dans une classe. Voici un aperçu de comment déclarer et appeler des méthodes en Java.

Déclaration d'une méthode

```
modificateur de visibilité void ou type nom ( liste de paramètres )
{
    corps de la méthode
}
```

Pour déclarer une méthode en Java, vous devez spécifier son type de retour, son nom, et éventuellement des paramètres. Voici la syntaxe générale :

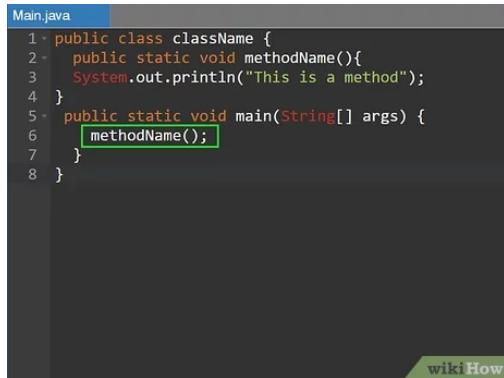
```
public class MaClasse {

    // Méthode sans retour et sans paramètres
    public void maMethode() {
        // Corps de la méthode
        System.out.println("Hello, World!");
    }

    // Méthode avec retour et paramètres
    public int additionner(int a, int b) {
        return a + b;
    }
}
```

}

Appel d'une méthode



The screenshot shows a code editor window titled "Main.java". The code is as follows:

```
1 public class className {
2     public static void methodName(){
3         System.out.println("This is a method");
4     }
5     public static void main(String[] args) {
6         methodName();
7     }
8 }
```

The line "methodName();" is highlighted with a green box. In the bottom right corner of the editor window, there is a watermark that says "wikHow".

Pour appeler une méthode, vous devez créer une instance de la classe si la méthode n'est pas **static**, puis utiliser le nom de l'instance suivi d'un point et du nom de la méthode.

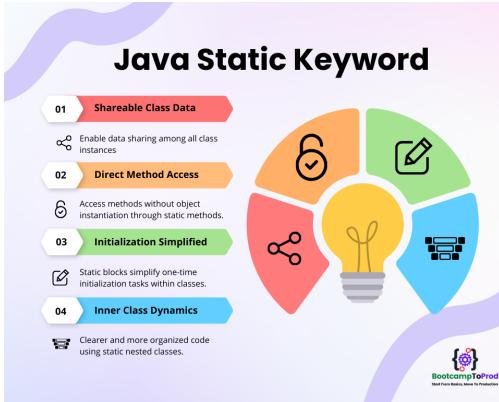
Exemple avec une méthode void :

```
public class App {
    public static void main(String[] args) {
        MaClasse objet = new MaClasse();
        objet.maMethode(); // Appel de la méthode
    }
}
```

Exemple avec une méthode qui retourne une valeur :

```
public class App {
    public static void main(String[] args) {
        MaClasse objet = new MaClasse();
        int resultat = objet.additionner(5, 3); // Appel de la méthode avec retour
        System.out.println("Le résultat est : " + resultat);
    }
}
```

Méthodes static



Les méthodes statiques peuvent être appelées sans créer une instance de la classe. Elles sont liées à la classe elle-même plutôt qu'à une instance particulière.

Exemple de méthode static :

```
public class Utilitaire {  
    public static int multiplier(int a, int b) {  
        return a * b;  
    }  
}
```

Appel d'une méthode static :

```
public class App {  
    public static void main(String[] args) {  
        int resultat = Utilitaire.multiplier(4, 2); // Appel sans instance  
        System.out.println("Le résultat est : " + resultat);  
    }  
}
```

Conclusion

- **Méthode void** : Ne retourne aucune valeur. Appelée avec `objet.nomDeLaMethode()`.
- **Méthode avec retour** : Retourne une valeur. Appelée avec `objet.nomDeLaMethode()` et peut être assignée à une variable.
- **Méthode static** : Appelée sans instance de classe via `NomDeLaClasse.nomDeLaMethode()`.

Ainsi, en Java, les concepts de fonction et méthode sont essentiellement équivalents, car toutes les fonctions doivent être encapsulées dans des classes sous forme de méthodes.