

## メソッドの宣言

```
修飾子 戻り値の型 メソッド名 ( 引数列 ) {  
    処理内容  
    return 文 ;  
}
```

修飾子	public や static などメソッドの属性（修飾子）を記述します。
戻り値の型	メソッドの処理が終了したときに返す値のデータ型（基本型、参照型）を指定します。 値を返さない（戻り値が無い）場合は、「void」と記述します。
メソッド名	メソッド名は自由ですが、1文字目は半角英字小文字である必要があります。 引数列の内容が異なれば同名のメソッド名で複数宣言する事もできます。（多重定義）
引数列	メソッドが要求する値（引数）を指定します。引数の書式は「型 引数名」で指定します。 引数は「,」で区切って複数記述する事もできます。（型 引数 1, 型 引数 2, ... , 型 引数 n）
return 文	メソッドから呼び出し元に処理を返します。その際、戻り値を指定する事ができます。 戻り値を指定する書式は「return 式 (値)」となります。 戻り値が不要な場合（void 宣言時）は省略します。

```
public class Execute {
```

```
① public static void main(String[] args) {  
    System.out.println("main");  
    methodA();  
    System.out.println("main");  
    int num = methodB(10);  
    System.out.println("main");  
    System.out.println("num=" + num);  
}
```

```
② public static void methodA() {  
    System.out.println("methodA");  
}
```

```
③ public static int methodB(int i) {  
    System.out.println("methodB");  
    i = i/2;  
    return i;  
}
```

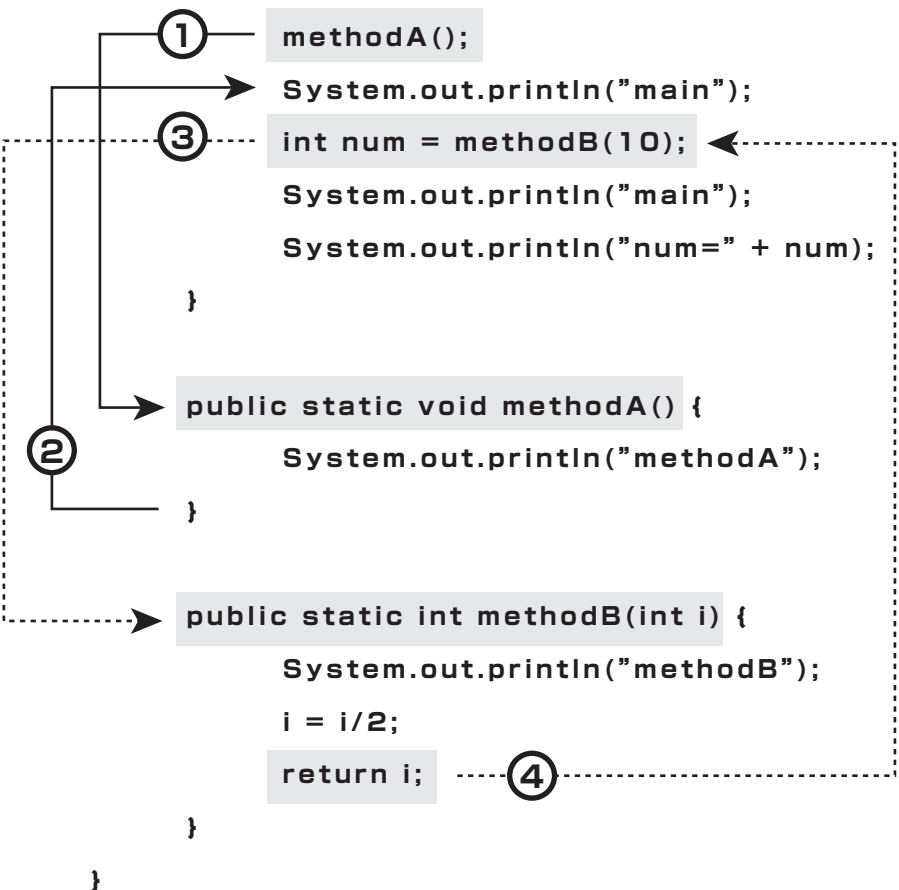
```
}
```

- **メソッド宣言はクラスの中に記述します。**  
左記の例だと Execute クラス宣言の { から } の中となります。
- **メソッド宣言の中にメソッド宣言を記述する事はできません。**
- **メソッド宣言を記述する順序は関係ありません。**  
プログラムが実行される時に最初に呼び出されるメソッドは「main」という名称のメソッドと規定されています。  
クラスの中でどの位置に main メソッドが記述されていても最初に呼び出されるメソッドは main メソッドであり、他のメソッドは明示的に呼び出さない限り実行される事はありません。
- **左記の例だと、メソッド宣言の数は3つとなります。**
  - ① main メソッド（戻り値：なし、引数：String 型配列）
  - ② methodA メソッド（戻り値：なし、引数：なし）
  - ③ methodB メソッド（戻り値：int 型、引数：int 型）

```
public class Execute {  
    public static void main(String[] args) {  
        System.out.println("main");  
        ① methodA(); ②  
        System.out.println("main");  
        ③ int num = methodB(10); ④  
        System.out.println("main");  
        System.out.println("num=" + num);  
    }  
  
    public static void methodA() {  
        System.out.println("methodA");  
    }  
  
    public static int methodB(int i) {  
        System.out.println("methodB");  
        i = i/2;  
        return i;  
    }  
}
```

- ① 戻り値なし (void 宣言) のメソッドの呼び出し  
methodA メソッドは戻り値なしの定義がなされているので、そのままメソッド名を記述します。
- ② 引数なし  
methodA メソッドは引数なしの定義がなされているので、メソッド名のあとに () のみ記述します。  
() が無いと変数と区別がつかず、エラーになるので注意しましょう。
- ③ 戻り値ありのメソッドの呼び出し  
methodB メソッドは int 型の戻り値が定義されているので、変数 = メソッド名 () の書式で、戻り値の代入を記述します。  
変数の型は戻り値の型に合わせます。
- ④ 引数あり  
methodB メソッドは int 型の引数を 1 つ要求しているので、メソッド名 (値) の書式で記述します。  
引数が複数ある場合は、[, ] で区切って複数記述します。  
その際必ず、型と数、順番を合わせる必要があります。

```
public class Execute {  
    public static void main(String[] args) {  
        System.out.println("main");  
        ① methodA();  
        System.out.println("main");  
        ③ int num = methodB(10);  
        System.out.println("main");  
        System.out.println("num=" + num);  
    }  
    public static void methodA() {  
        System.out.println("methodA");  
    }  
    ②  
    public static int methodB(int i) {  
        System.out.println("methodB");  
        i = i/2;  
        return i; ④  
    }  
}
```



左記プログラムの実行結果

```
>java Execute  
main  
methodA  
main  
methodB  
main  
num=5
```

- ① methodA を呼び出し、処理を methodA に移します。
- ② methodA の処理が終了したので、処理が main に戻ります。
- ③ methodB に引数 10 を渡して呼び出し、処理を methodB に移します。
- ④ return 文で戻り値に変数 i の値を設定し、処理を main に戻します。  
main に戻った後、変数 num に戻り値を代入します。

```
int num = methodB(10);
```

戻り値を num に代入