

第17章 コンストラクタ

目次

- コンストラクタの基本
- コンストラクタのオーバーロード

コンストラクタとは

オブジェクトを初期化したい場合は、コンストラクタを利用します。

オブジェクトが生成された直後に1度だけ実行される処理のこと。
コンストラクタを利用して、オブジェクト生成時にフィールドに
初期値を代入することができる。

コンストラクタの定義

通常のメソッドとの違い

- コンストラクタの名前は、クラス名と同じである。
- 戻り値の型を指定しない。

```
修飾子 クラス名(引数リスト){  
    処理;  
}
```

(例) Dogクラスのコンストラクタ

```
public class Dog {  
    private String gender;  
    private String breed;  
  
    public Dog() {  
        gender = "オス";  
        breed = "柴犬";  
        System.out.println("犬を作成しました。");  
    }  
}
```

コンストラクタ

コンストラクタを呼び出す

コンストラクタはオブジェクト生成時に呼び出される。

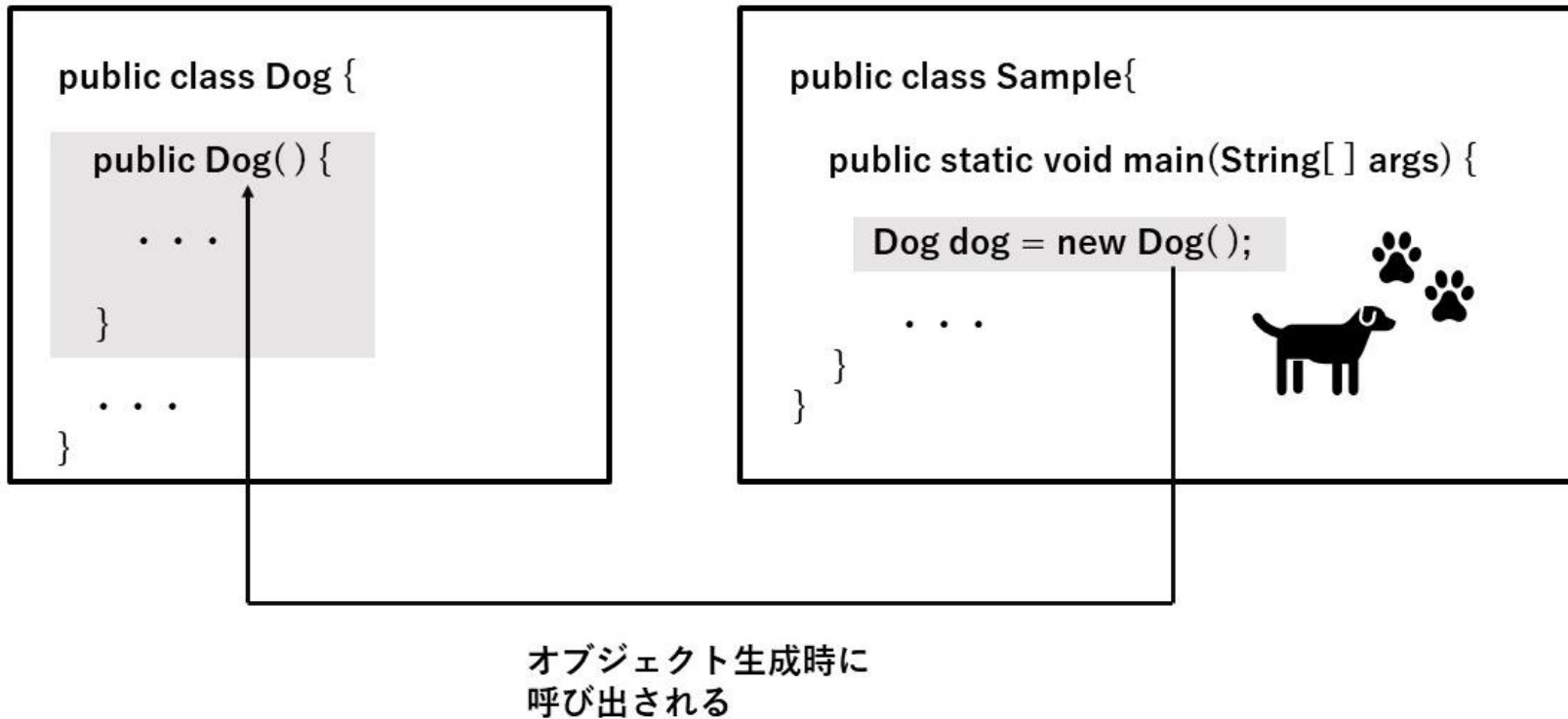
```
クラス名 オブジェクト名 = new クラス名(引数リスト);
```

(例) Dogクラスのコンストラクタ実行

```
public class Sample {  
    public static void main(String[] args) {  
        Dog dog = new Dog();  
    }  
}
```

犬を作成しました。

(例) Dogクラスのコンストラクタ実行



引数を持つコンストラクタ

コンストラクタは、メソッドと同様に引数を持つことができる。

```
public Dog(String gender) {
```

複数個の引数を定義する場合は、「,(カンマ)」区切りを行う。

```
public Dog(String gender, String breed) {
```

【Sample1701 引数ありのコンストラクタ】 を作成しましょう



Sample1701のポイント

Dog1701クラスでは、コンストラクタに
引数genderとbreedが定義されている。
各引数の値はフィールドに代入される処理が記述されている。

```
public Dog1701(String gender, String breed) {  
    this.gender = gender;  
    this.breed = breed;  
    System.out.println("犬を作成しました。");  
}
```

Sample1701のポイント

Sample1701クラスでは、オブジェクト生成時に
コンストラクタに実引数を指定している。

この処理を実行すると、

「性別がオスで、犬種が柴犬の犬のオブジェクト」が生成される。

```
Dog1701 dog = new Dog1701("オス", "柴犬");
```



<dog>
性別:オス
犬種:柴犬

【Sample1702 コンストラクタのオーバーロード】 を作成しましょう

コンストラクタも、引数の型や個数のいずれかが異なっていれば、オーバーロードできる。

Let's try!



Sample1702のポイント

Sample1702クラスでは、2つのオブジェクトを作成している
1つ目のオブジェクト生成では引数なしのコンストラクタを
呼び出している。

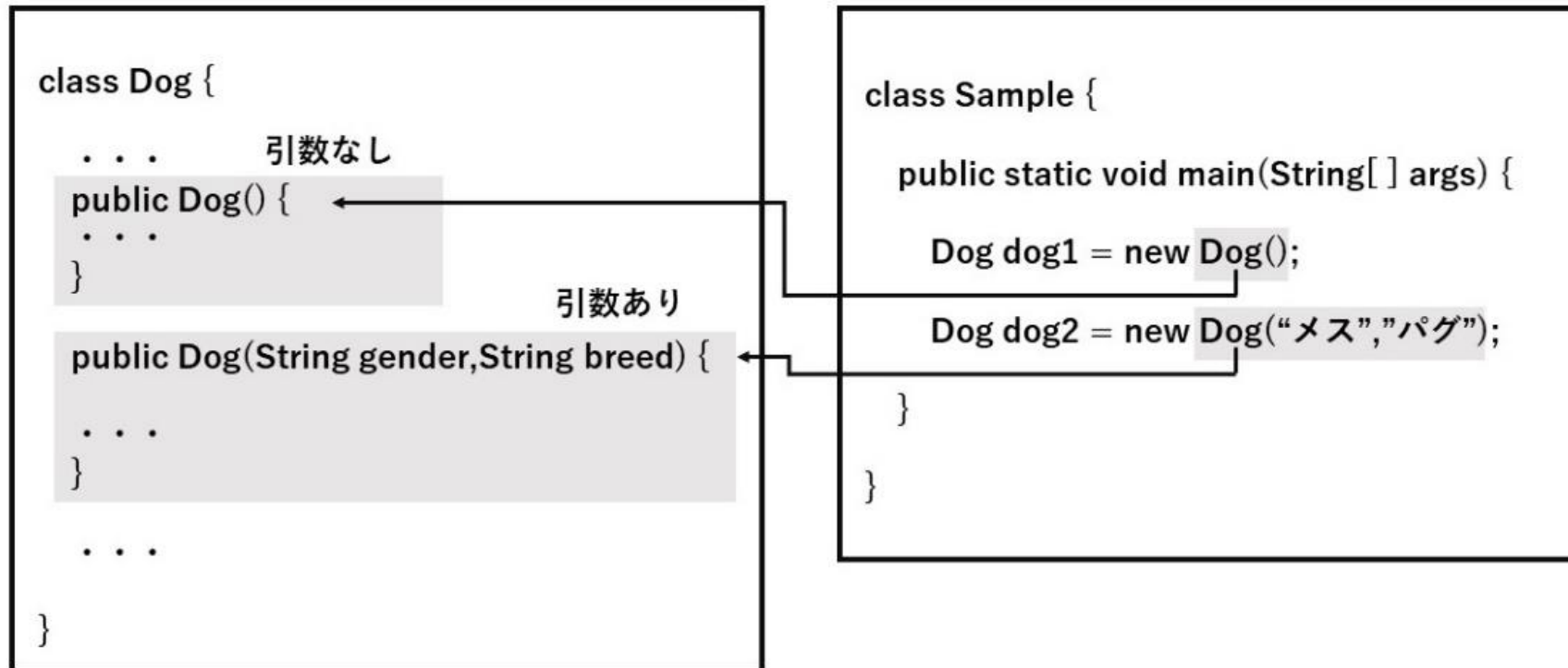
```
Dog1702 dog1 = new Dog1702();
```

Sample1702のポイント

2つ目のオブジェクト生成では引数なしの
コンストラクタを呼び出している。

```
Dog1902 dog2 = new Dog1902("メス", "トイプードル");
```

Sample1702のポイント



コンストラクタをオーバーロードして、
さまざまな初期設定処理が行えるようになります。

別のコンストラクタを呼び出す

コンストラクタ内では、同じクラスに定義された別のコンストラクタを呼び出せる。

```
this();
```

「this()」は同じクラスのコンストラクタのことを指しており、
「()」はコンストラクタの引数を意味する。
引数なしの場合、上記のように記述する。

別のコンストラクタを呼び出す

引数ありのコンストラクタを呼び出したい場合は、
下記のように「()」内に実引数を指定する。

```
this(“オス”, “柴犬”);
```

別のコンストラクタを呼び出す

「this()」はコンストラクタの1行目にしか記述できない。
2行目以降に記述するとコンパイルエラーとなるため注意。

this()を使って、コンストラクタ内で
別のコンストラクタを呼び出すことができます。
this()はコンストラクタの先頭に記述します。

【Sample1703 別のコンストラクタを呼び出す】 を作成しましょう



Sample1703のポイント

Dog1703クラスでは、引数ありのコンストラクタの先頭で
「this();」により、引数なしのコンストラクタを呼び出している。

```
this();  
this.breed = breed;
```

引数なしのコンストラクタ

デフォルトコンストラクタ

コンパイラが作成する見えないコンストラクタ。
コンストラクタが定義されていない場合は、
「デフォルトコンストラクタ」と呼ばれる
「引数のないコンストラクタ」が自動的に実行される。

クラス内にコンストラクタが1つでも定義されると、
コンパイラはデフォルトコンストラクタを作成しなくなる。

章のまとめ

- コンストラクタは、オブジェクトの生成とともに自動的に呼び出されます。
- コンストラクタの役割は、オブジェクトの初期化を行うことです。
- コンストラクタはクラス名と同じ名前を持ち、戻り値を持ちません。
- コンストラクタが定義されていないとき、引数を持たないデフォルトコンストラクタが自動的に定義され実行されます。
- コンストラクタを定義すると、デフォルトコンストラクタは作成されなくなります。
- メソッドと同様に、コンストラクタもオーバーロードをすることができます。