

- 1 SA31 Java文法 Lesson3 抽象クラスとインターフェース
- 2
- 3 [1] プロジェクトの作成
- 4 プロジェクト名「JavaLesson3」でJavaプロジェクトを作成する。
- 5
- 6 [2] 写経確認
- 7 2025/12/10(水)
- 8
- 9 [3] ソースコード
- 10 配布ソースコードのリストは以下の通り。
- 11
- 12 ■Ex1
- 13 1 AbstractAnimal.java
- 14 2 Dog.java
- 15 3 Cat.java
- 16 4 CallAnimals.java
- 17
- 18 ■Ex2
- 19 1 AnimalInterface.java
- 20 2 Dog.java
- 21 3 Cat.java
- 22 4 AnimalPrinter.java
- 23 5 CallAnimals.java

```
1 package local.hal.sa31.java.lesson3.ex1;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex1 Src1
5  *
6  * 抽象的な動物を表すクラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public abstract class AbstractAnimal {
11     /**
12      * その動物の愛称を表すフィールド。
13      */
14     private String _name;
15
16     /**
17      * 愛称フィールドのセッタ。
18      */
19     public void setName(String name) {
20         _name = name;
21     }
22
23     /**
24      * 愛称フィールドのゲッタ。
25      */
26     public String getName() {
27         return _name;
28     }
29
30     /**
31      * 自己紹介するメソッド。
32      */
33     public void selfIntro() {
34         System.out.println("わたしの名前は" + _name + "だよ～");
35     }
36
37     /**
38      * 鳴き声のメソッド。
39      *
40      * @return 鳴き声。
41      */
42     public abstract String call();
43 }
44
```

```
1 package local.hal.sa31.java.lesson3.ex1;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex1 Src2
5  *
6  * 犬を表すクラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class Dog extends AbstractAnimal {
11     @Override
12     public String call() {
13         return "わん、わん";
14     }
15 }
16
```

```
1 package local.hal.sa31.java.lesson3.ex1;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex1 Src3
5  *
6  * 猫を表すクラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class Cat extends AbstractAnimal {
11     @Override
12     public String call() {
13         return "にゃあ、にゃあ";
14     }
15 }
16
```

```
1 package local.hal.sa31.java.lesson3.ex1;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex1 Src4
5  *
6  * 実行クラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class CallAnimals {
11     public static void main(String[] args) {
12         // AbstractAnimal pet = new AbstractAnimal();
13         // pet.call();
14
15         Dog pet1 = new Dog();
16         pet1.setName("しろ");
17         pet1.selfIntro();
18         System.out.println(pet1.call());
19
20         Cat pet2 = new Cat();
21         pet2.setName("たま");
22         pet2.selfIntro();
23         System.out.println(pet2.call());
24     }
25 }
26
```

```
1 package local.hal.sa31.java.lesson3.ex2;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex2 Src1
5  *
6  * 動物を表すインターフェース。
7  *
8  * @author Shinzo SAITO
9  */
10 public interface AnimalInterface {
11     /**
12      * 自己紹介の元となる文字列定数。
13      */
14     String SELF_INTRO_BASE = "私の名前";
15     /**
16      * 動物の愛称を得るメソッド。
17      *
18      * @return 動物の愛称。
19      */
20     String getName();
21     /**
22      * 自己紹介するメソッド。
23      */
24     default void selfIntro() {
25         System.out.println(SELF_INTRO_BASE + "は" + getName() + "だよ～");
26     }
27     /**
28      * 動物の鳴き声を得るメソッド。
29      *
30      * @return 動物の鳴き声。
31      */
32     String call();
33     /**
34      * 無意味なメソッド。
35      */
36     void doNothing();
37 }
38
```

```
1 package local.hal.sa31.java.lesson3.ex2;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex2 Src2
5  *
6  * AnimalInterfaceを実装した犬クラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class Dog implements AnimalInterface {
11     @Override
12     public String getName() {
13         return "ぼち";
14     }
15
16     @Override
17     public void selfIntro() {
18         System.out.println(SELF_INTRO_BASE + "は" + getName() + "だぜい");
19     }
20
21     @Override
22     public String call() {
23         return "わんわん";
24     }
25
26     @Override
27     public void doNothing() {
28         String msg = "こんにちは" + "さようなら";
29         msg = "また明日";
30     }
31 }
32
```

```
1 package local.hal.sa31.java.lesson3.ex2;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex2 Src3
5  *
6  * AnimalInterfaceを実装した猫クラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class Cat implements AnimalInterface {
11     @Override
12     public String getName() {
13         return "たま";
14     }
15
16     @Override
17     public String call() {
18         return "にゃあにゃあ";
19     }
20
21     @Override
22     public void doNothing() {}
23 }
24
```



```
1 package local.hal.sa31.java.lesson3.ex2;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex2 Src4
5  *
6  * AnimalInterfaceの処理クラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class AnimalPrinter {
11     /**
12      * AnimalInterfaceの内容を表示するメソッド。
13      *
14      * @param animal AnimalInterfaceを実装したクラス。
15      */
16     public void show(AnimalInterface animal) {
17         System.out.println("----- 表示します");
18         animal.selfIntro();
19         String name = animal.getName();
20         String call = animal.call();
21         System.out.println(name + "は" + call);
22     }
23 }
24
```

```
1 package local.hal.sa31.java.lesson3.ex2;
2
3 /**
4  * SA31 Android用Java文法 Lesson3 Ex2 Src5
5  *
6  * 実行クラス。
7  *
8  * @author Shinzo SAITO
9  */
10 public class CallAnimals {
11     public static void main(String[] args) {
12         // AnimalInterface pet0 = new AnimalInterface();
13
14         Dog pet1 = new Dog();
15         Cat pet2 = new Cat();
16         // AnimalInterface pet1 = new Dog();
17         // AnimalInterface pet2 = new Cat();
18
19         AnimalPrinter printer = new AnimalPrinter();
20         printer.show(pet1);
21         printer.show(pet2);
22     }
23 }
24
```