

5.3.4 클래스 상속과 오버라이딩을 이용한 간단한 게임

- 클래스 상속과 메소드 오버라이딩을 이용하는 간단한 게임을 구현해 보았음
- 애완동물의 기본 속성을 갖는 클래스인 "Pat"을 부모클래스로 구현하고 이를 상속받아 "Dog", "Cat", "Rabbit" 클래스를 생성하고, 각 클래스를 객체로 생성하여 밥먹기, 잠자기, 운동하기, 공격하기 등의 동작을 통해 파워, 에너지 등의 멤버 변수의 값이 어떻게 동작하는지 알아봄

Pat 클래스: 애완동물의 속성을 갖는 부모 클래스

- 애완동물의 기본 속성을 갖는 부모 클래스를 구성함
- 클래스의 이름은 "Pat"으로 정하고, 애완동물의 속성 값을 갖는 멤버 변수로는 애완동물의 종류, 이름, 나이, 파워, 에너지 등을 저장할 수 있도록 정함
- 애완동물의 동작을 지원하는 메소드로는 밥 먹기, 잠자기, 운동하기, 나이 먹기, 공격하기, 현재 애완동물의 상태 출력하기 등의 기능을 수행하도록 구성함

5.3.4 클래스 상속과 오버라이딩을 이용한 간단한 게임

Dog 클래스: 애완동물 강아지 클래스

- Pat 클래스를 상속받아 강아지의 속성을 갖는 Dog 클래스임
- Dog 클래스의 경우 밥 먹기 동작을 수행하면 기본 파워는 200, 에너지는 300으로 생성되며, 밥 먹기를 수행하면 에너지가 15 증가하고, 잠자기는 에너지 5 증가, 운동하기는 에너지 10 감소, 파워 10 증가, 공격하기는 에너지 20 감소, 파워 30 증가의 동작을 수행함

Cat 클래스: 애완동물 고양이 클래스

- Pat 클래스를 상속받아 고양이의 속성을 갖는 Cat 클래스임
- Cat 클래스의 경우 밥 먹기 동작을 수행하면 기본 파워는 100, 에너지는 200으로 생성되며, 밥 먹기를 수행하면 에너지가 15 증가하고, 잠자기는 에너지 5 증가, 운동하기는 에너지 10 감소, 파워 10 증가, 공격하기는 에너지 15 감소, 파워 20 증가의 동작으로 부모 클래스인 Pat과 동일한 동작을 수행함
- 이 경우에는 각 메소드를 오버라이딩하지 않아도 무방함

5.3.4 클래스 상속과 오버라이딩을 이용한 간단한 게임

Rabbit 클래스: 애완동물 토끼 클래스

- Pat 클래스를 상속받아 토끼의 속성을 갖는 Rabbit 클래스임
- Rabbit 클래스의 경우 밥 먹기 동작을 수행하면 기본 파워는 30, 에너지는 100으로 생성되며, 밥 먹기를 수행하면 에너지가 5 증가하고, 잠자기는 에너지 3 증가, 운동하기는 에너지 3 감소, 파워 7 증가, 공격하기는 에너지 10 감소, 파워 10 증가의 동작을 수행함

간단한 게임 만들기

- 부모 클래스인 "Pat"과 이를 상속하여 "Dog", "Cat", "Rabbit" 클래스들을 생성하고, 메인 클래스에서는 원하는 애완동물을 선택하여 생성한 후, 밥 먹기, 잠자기, 운동하기, 공격하기 등의 동작을 수행하도록 구현해봄

출처: 프로그램 새내기를 위한 자바 언어 프로그래밍, 기한재

34

```

애완동물을 선택하세요.
1. 강아지, 2. 고양이, 3. 토끼3
애완동물의 이름을 입력하세요. 토끼이
시작
토끼(토끼이)가 만들어졌습니다.
메뉴
1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝
1
나이: 0
파워: 30
에너지: 105
메뉴
1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝
2
나이: 0
파워: 30
에너지: 108
메뉴
1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝
3
나이: 0
파워: 37
에너지: 105
메뉴
1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝
4
나이: 0
파워: 47
에너지: 95
메뉴
1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝
5

```

애완동물들 선택하세요.

1. 강아지, 2. 고양이, 3. 토끼1

애완동물의 이름을 입력하세요. 땡땡이

시작

강아지(땡땡이)가 만들어졌습니다.

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

1

나이: 0

파워: 200

에너지: 315

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

2

나이: 0

파워: 200

에너지: 320

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

3

나이: 0

파워: 220

에너지: 300

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

4

나이: 0

파워: 250

에너지: 280

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

5

애완동물들 선택하세요.

1. 강아지, 2. 고양이, 3. 토끼²

애완동물의 이름을 입력하세요. **고냥이**

시작

고양이(고냥이)가 만들어졌습니다.

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

1

나이: 0

파워: 100

에너지: 215

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

2

나이: 0

파워: 100

에너지: 220

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

3

나이: 0

파워: 110

에너지: 210

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

4

나이: 0

파워: 130

에너지: 195

메뉴

1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝

5

PatGrow.java ×

```
1 import java.util.Scanner;
2 class Pat{
3     public String race; //애완동물의 종류
4     public String name; //애완동물 이름
5     public int age; //나이
6     public int power; //파워
7     public int energy; //에너지
8     int count = 0;
9     public Pat() {
10         System.out.println("시작");
11     }
12     public void eat() {
13         energy += 15;
14         count++;
15         aging();
16     }
17     public void sleep() {
18         energy += 5;
19         count++;
20         aging();
21     }
22     public void exercise() {
23         energy -= 10;
24         power += 10;
25     }
26     public void aging() {
27         if(count>3) {
28             age++;
29             count = 0;
30         }
31     }
32     public void attack() {
33         energy -= 15;
34         power += 20;
35     }
36     public void print() {
37         System.out.println("나이: "+age);
38         System.out.println("파워: "+power);
39         System.out.println("에너지: "+energy);
40     }
41 }
```

```

42 class Dog extends Pat{
43     public int count;
44     public Dog(String name) {
45         this.name = name;
46         race = "강아지";
47         age = 0;
48         power = 200;
49         energy = 300;
50         count = 0;
51         System.out.println("강아지("+name+")가 만들어졌습니다.");
52     }
53     public void eat() {
54         energy += 15;
55         count++;
56         aging();
57     }
58     public void sleep() {
59         energy += 5;
60         count++;
61         aging();
62     }
63     public void exercise() {
64         energy -= 20;
65         power += 20;
66     }
67     public void aging() {
68         if(count>3) {
69             age++;
70             count = 0;
71         }
72     }
73     public void attack() {
74         energy -= 20;
75         power += 30;
76     }
77     public void print() {
78         System.out.println("나이: "+age);
79         System.out.println("파워: "+power);
80         System.out.println("에너지: "+energy);
81     }
82 }

83 class Cat extends Pat{
84     public int count;
85     public Cat(String name) {
86         this.name = name;
87         race = "고양이";
88         age = 0;
89         power = 100;
90         energy = 200;
91         count = 0;
92         System.out.println("고양이("+name+")가 만들어졌습니다.");
93     }
94 }

```

```

95 class Rabbit extends Pat{
96     public int count;
97     public Rabbit(String name) {
98         this.name = name;
99         race = "트끼";
100         age = 0;
101         power = 30;
102         energy = 100;
103         count = 0;
104         System.out.println("트끼("+name+")가 만들어졌습니다.");
105     }
106     public void eat() {
107         energy += 5;
108         count++;
109         aging();
110     }
111     public void sleep() {
112         energy += 3;
113         count++;
114         aging();
115     }
116     public void exercise() {
117         energy -= 3;
118         power += 7;
119     }
120     public void aging() {
121         if(count>3) {
122             age++;
123             count = 0;
124         }
125     }
126     public void attack() {
127         energy -= 10;
128         power += 10;
129     }
130     public void print() {
131         System.out.println("나이: "+age);
132         System.out.println("파워: "+power);
133         System.out.println("에너지: "+energy);
134     }
135 }

```

```

136 public class PatGrow {
137     public static void main(String[] args) {
138         Scanner sc = new Scanner(System.in);
139         Pat p = null;
140         System.out.println("애완동물들 선택하세요.");
141         System.out.print("1. 강아지, 2. 고양이, 3. 토끼");
142         int x = sc.nextInt();
143         System.out.print("애완동물의 이름을 입력하세요.");
144         String name = sc.next();
145         if(x==1) {
146             p = new Dog(name);
147         }else if(x==2) {
148             p = new Cat(name);
149         }else if(x==3) {
150             p = new Rabbit(name);
151         }
152         boolean flag = true;
153         while(flag) {
154             System.out.println("메뉴");
155             System.out.println("1. 밥먹기 2. 잠자기 3. 운동하기 4. 공격 5. 끝");
156             x = sc.nextInt();
157             switch (x) {
158                 case 1:
159                     p.eat();
160                     p.print();
161                     break;
162                 case 2:
163                     p.sleep();
164                     p.print();
165                     break;
166                 case 3:
167                     p.exercise();
168                     p.print();
169                     if(p.energy<=0) {
170                         System.out.println("게임 종료");
171                         flag = false;
172                     }
173                     break;
174                 case 4:
175                     p.attack();
176                     p.print();
177                     if(p.energy <=0) {
178                         System.out.println("게임 종료");
179                         flag = false;
180                     }
181                     break;
182                 case 5:
183                     flag = false;
184             }
185         }
186     }
187 }

```