

Part 1:

-Virtual Dog-

```
  _ _ _ _ _  
 / ( . . ) \   )  
  (*) _____ / |  
   /           |  
  /    | -- \   |  
 ( ) ( )   ( )
```

Please Enter a name for your pet
Dog:

Danny

Initializing Stats...

-Danny's Stats-

Hunger: 3

Cleanliness: 8

Happiness: 8

Health: 8

Danny was born!

Main Menu:

1. Feed
2. Wash
3. Play
4. Health
5. -Quit-

1

Danny has their stomach satisfied
by the bowl of soup you offer.

-Danny's Stats-

Hunger: 4

Cleanliness: 8

Happiness: 8

Health: 8

Main Menu:

1. Feed
2. Wash
3. Play
4. Health
5. -Quit-

2

Danny is sparkling clean after the
long bath you give them.

-Danny's Stats-

Hunger: 4

Cleanliness: 9

Happiness: 8

Health: 8

Main Menu:

1. Feed
2. Wash
3. Play
4. Health
5. -Quit-

3

Danny seems much more positive after you
give them a big hug.

-Danny's Stats-

Hunger: 4

Cleanliness: 9

Happiness: 9

Health: 8

Main Menu:

1. Feed
2. Wash
3. Play
4. Health
5. -Quit-

4

Danny looks much healthier after
their long rest.

-Danny's Stats-

Hunger: 4

Cleanliness: 9

Happiness: 9

Health: 9

Main Menu:

1. Feed
2. Wash
3. Play
4. Health
5. -Quit-

1

Danny has their stomach satisfied
by the bowl of soup you offer.

-Danny's Stats-

Hunger: 5

Cleanliness: 9

Happiness: 9

Health: 9

Main Menu:

1. Feed
2. Wash

```
3. Play
4. Health
5. -Quit-
2
Danny smells great after the long
shower they take.
-Danny's Stats-
Hunger: 5
Cleanliness: 10
Happiness: 9
Health: 9
Main Menu:
1. Feed
2. Wash
3. Play
4. Health
5. -Quit-
3
Danny's eyes sparkle after you
finish playing catch.
-Danny's Stats-
Hunger: 5
Cleanliness: 10
Happiness: 10
Health: 9
Main Menu:
1. Feed
2. Wash
3. Play
4. Health
5. -Quit-
4
Danny looks much healthier after
their long rest.
-Danny's Stats-
Hunger: 5
Cleanliness: 10
Happiness: 10
Health: 10
Main Menu:
1. Feed
2. Wash
3. Play
4. Health
5. -Quit-
```

```
1
Danny shouts with joy as you
present a watermelon slice.
-Danny's Stats-
Hunger: 6
Cleanliness: 10
Happiness: 10
Health: 10
Main Menu:
1. Feed
2. Wash
3. Play
4. Health
5. -Quit-
2
Danny is sparkling clean after the
long bath you give them.
-Danny's Stats-
Hunger: 6
Cleanliness: 11
Happiness: 10
Health: 10

Danny Died of old age
Quitting Program...
```

```

1 import java.util.Scanner;
2
3 public class Six {
4
5     public static void main(String[] args) {
6         //variable initialization
7         Scanner input = new Scanner(System.in);
8         int menuChoice = 0;
9
10        //prints a nice, formatted heading on the program
11        printPet();
12
13        //calls the constructor for the pet class and stores the name
14        // read in from input
15        VirtualPet pet = new VirtualPet(input.nextLine());
16        System.out.println("Initializing Stats...");
17
18        //runs a loop until the user chooses to quit the program or until
19        // the pet reaches the age of 10 and dies
20        do{
21            printStats(pet); //prints the stats in a nice format
22
23            //prints out the pet was born if the pet's age is 0
24            if(pet.getAge() == 0) {
25                System.out.println(pet.getName() + " was born!");
26            }
27
28            printMenu(); //prints the menu for the user to see
29            menuChoice = input.nextInt(); //stores the menu input
30            //updates the stats of the pet based on the menu input
31            menuStatUpdate(menuChoice, pet);
32
33        } while(menuChoice != 5 && pet.getAge() < 10);
34
35        printStats(pet);
36
37        System.out.println();
38
39        //outputs based on the pet age
40        if(pet.getAge() < 10) {
41            System.out.println(pet.getName() + " ended up being "
42                + pet.getAge() + " years old");
43        }
44        else {
45            System.out.println(pet.getName() + " Died of old age");
46        }

```

```

47
48     System.out.println("Quitting Program...");
49 }
50
51 //formats the stats to print out to the console
52 public static void printStats(VirtualPet pet) {
53     System.out.println("-" + pet.getName() + "'s Stats-"
54         + "\nHunger: " + pet.getHunger() + "\nCleanliness: "
55         + pet.getCleanliness() + "\nHappiness: " + pet.getHappiness()
56         + "\nHealth: " + pet.getHealth());
57 }
58
59 //formats the menu to print out to the console
60 public static void printMenu() {
61     System.out.println("Main Menu:\n1. Feed\n2. Wash\n3. Play\n"
62         + "4. Health\n5. -Quit-");
63 }
64
65 //formatted heading to the program
66 public static void printPet() {
67     System.out.println("-Virtual Dog-\r\n" + " _ _ \r\n" +
68         "/(. .)\ \r\n" +
69         " (*) ____/|\r\n" +
70         " /      |\r\n" +
71         " / |--\ \ |\r\n" +
72         "(_)(_) (_)\r\nPlease Enter a name for your pet"
73         + "\nDog:");
74 }
75
76 //switch statement that checks the menu input choice and will
77 // update the stat that matches the menu choice on the pet object
78 // that is passed in with the choice
79 public static void menuStatUpdate(int choice, VirtualPet pet) {
80     switch(choice) {
81         case 1: pet.feed();
82                 pet.birthday();
83                 break;
84         case 2: pet.wash();
85                 pet.birthday();
86                 break;
87         case 3: pet.play();
88                 pet.birthday();
89                 break;
90         case 4: pet.health();

```

```
91         pet.birthday();
92         break;
93     //case where the user quits does nothing and will exit the loop
94     case 5: break;
95     //case where invalid input is entered will not increase the age
96     // but will proceed in the program
97     default: System.out.println("Invalid input, please enter a valid "
98         + "number between 1 and 5");
99     }
100 }
101
102 }
```

```

1
2 public class VirtualPet {
3     private String name;
4     private int hunger;
5     private int cleanliness;
6     private int happiness;
7     private int health;
8     private int age;
9
10
11 public VirtualPet(String namef) {
12     //initialized variables
13     age = 0;
14     this.name = namef;
15     //randomly initialized variables
16     this.hunger = (int) (Math.random() * 10) + 1;
17     this.cleanliness = (int) (Math.random() * 10) + 1;
18     this.happiness = (int) (Math.random() * 10) + 1;
19     this.health = (int) (Math.random() * 10) + 1;
20 }
21
22 //getter methods
23 public String getName() {return name;}
24 public int getAge() {return age;}
25 public int getHunger() {return hunger;}
26 public int getCleanliness() {return cleanliness;}
27 public int getHappiness() {return happiness;}
28 public int getHealth() {return health;}
29
30 //setter methods
31 public void setName(String namef) { name = namef;}
32 public void setAge(int agef) { age = agef;}
33 public void setHunger(int hungerf) { hunger = hungerf;}
34 public void setCleanliness(int cleanlinessf) { cleanliness = cleanlinessf;}
35 public void setHappiness(int happinessf) { happiness = happinessf;}
36 public void setHealth(int healthf) { health = healthf;}
37
38 //stat update methods
39 public void feed() { //updates the hunger and outputs a random text dialog
40     setHunger(getHunger() + 1);
41     if((int) (Math.random()*2) == 0) {
42         System.out.println(getName() + " shouts with joy"
43             + " as you\npresent a watermelon slice.");
44     }
45     else {

```

```

46         System.out.println(getName() + " has their stomach"
47             + " satisfied\nby the bowl of soup you offer.");
48     }
49 }
50
51 public void wash() { //updates the cleanliness and outputs a random text dialog
52     setCleanliness(getCleanliness() + 1);
53     if((int)(Math.random()*2) == 0) {
54         System.out.println(getName() + " is sparkling clean"
55             + " after the\nlong bath you give them.");
56     }
57     else {
58         System.out.println(getName() + " smells great after"
59             + " the long\nshower they take.");
60     }
61 }
62
63 public void play() { //updates the happiness and outputs a random text dialog
64     setHappiness(getHappiness() + 1);
65     if((int)(Math.random()*2) == 0) {
66         System.out.println(getName() + "'s eyes sparkle after"
67             + " you\nfinish playing catch.");
68     }
69     else {
70         System.out.println(getName() + " seems much more positive"
71             + " after you\ngive them a big hug.");
72     }
73 }
74
75 public void health() { //updates the health and outputs a random text dialog
76     setHealth(getHealth() + 1);
77     if((int)(Math.random()*2) == 0) {
78         System.out.println(getName() + " looks much healthier"
79             + " after\ntheir long rest.");
80     }
81     else {
82         System.out.println(getName() + "'s muscles are looking"
83             + " larger\nafter exercising.");
84     }
85 }
86
87 public void birthday() { //makes the pet a year older
88     age += 1;
89 }
90 }

```

Part 2:

```
-Virtual Dog-

/(\. .)\  )
  (*)____/|
  /      |
  /      |--\ |
  ( ) ( ) ( )
Please Enter a name for your pet
Dog:
George
Initializing Stats...
-George's Stats-
Hunger: 7
Cleanliness: 3
Happiness: 7
Health: 1
George was born!
Main Menu:
1. Feed
2. Wash
3. Play
4. Health
5. -Quit-
1
George has their stomach satisfied
by the bowl of soup you offer. (+1 Hunger)
George is starting to smell bad. (-1 Health)
George can hear their stomach
growling. (-1 Hunger)
-George's Stats-
Hunger: 7
Cleanliness: 3
Happiness: 7
Health: 0

Insifficient stat: George has died
Quitting Program...
```



```

1 import java.util.Scanner;
2
3 public class Six {
4
5     public static void main(String[] args) {
6         //variable initialization
7         Scanner input = new Scanner(System.in);
8         int menuChoice = 0;
9         boolean dead = false;
10
11         //prints a nice, formatted heading on the program
12         printPet();
13
14         //calls the constructor for the pet class and stores the name
15         // read in from input
16         VirtualPet pet = new VirtualPet(input.nextLine());
17         System.out.println("Initializing Stats...");
18
19         //runs a loop until the user chooses to quit the program or until
20         // the pet reaches the age of 10 and dies
21         do{
22             printStats(pet); //prints the stats in a nice format
23
24             //prints out the pet was born if the pet's age is 0
25             if(pet.getAge() == 0) {
26                 System.out.println(pet.getName() + " was born!");
27             }
28
29             printMenu(); //prints the menu for the user to see
30             menuChoice = input.nextInt(); //stores the menu input
31             //updates the stats of the pet based on the menu input
32             dead = menuStatUpdate(menuChoice, pet);
33
34             } while(menuChoice != 5 && pet.getAge() < 10 && !dead);
35
36         printStats(pet);
37
38         System.out.println();
39
40         //outputs based on the pet age
41         if(pet.getAge() < 10 && !dead) {
42             System.out.println(pet.getName() + " ended up being "
43                 + pet.getAge() + " years old");
44         }
45         else if(dead) {
46             System.out.println("Insufficient stat: " + pet.getName()
47                 + " has died");
48         }
49         else {
50             System.out.println(pet.getName() + " Died of old age");
51         }
52
53         System.out.println("Quitting Program...");
54     }
55
56     //formats the stats to print out to the console

```

```

57 public static void printStats(VirtualPet pet) {
58     System.out.println("-" + pet.getName() + "'s Stats-"
59         + "\nHunger: " + pet.getHunger() + "\nCleanliness: "
60         + pet.getCleanliness() + "\nHappiness: " + pet.getHappiness()
61         + "\nHealth: " + pet.getHealth());
62 }
63
64 //formats the menu to print out to the console
65 public static void printMenu() {
66     System.out.println("Main Menu:\n1. Feed\n2. Wash\n3. Play\n"
67         + "4. Health\n5. -Quit-");
68 }
69
70 //formatted heading to the program
71 public static void printPet() {
72     System.out.println( "-Virtual Dog-\r\n" + " _ _\r\n" +
73         "/(. .)\ \r\n" +
74         " (*)____/|\r\n" +
75         " / _____|\r\n" +
76         " / |--\ \r\n" +
77         "(_) ( ) (_)\r\nPlease Enter a name for your pet"
78         + "\nDog:");
79 }
80
81 //switch statement that checks the menu input choice and will
82 // update the stat that matches the menu choice on the pet object
83 // that is passed in with the choice
84 public static boolean menuStatUpdate(int choice, VirtualPet pet) {
85     boolean death = false;
86     switch(choice) {
87         case 1: pet.feed();
88             death = checkAttributes(pet);
89             pet.birthday();
90             break;
91         case 2: pet.wash();
92             death = checkAttributes(pet);
93             pet.birthday();
94             break;
95         case 3: pet.play();
96             death = checkAttributes(pet);
97             pet.birthday();
98             break;
99         case 4: pet.health();
100             death = checkAttributes(pet);
101             pet.birthday();
102             break;
103         //case where the user quits does nothing and will exit the loop
104         case 5: break;
105         //case where invalid input is entered will not increase the age
106         // but will proceed in the program
107         default: System.out.println("Invalid input, please enter a valid "
108             + "number between 1 and 5");
109     }
110
111     return death;
112 }

```

```
113
114 //static method that checks to see if stats are below 5, if so then
115 // the low stats cause other stats to drop
116 public static boolean checkAttributes(VirtualPet petf) {
117     Attribute_Tool.checkCleanliness(petf);
118     Attribute_Tool.checkHappiness(petf);
119     Attribute_Tool.checkHealth(petf);
120     Attribute_Tool.checkHunger(petf);
121     return Attribute_Tool.checkDeath(petf);
122 }
123
124
125 }
```

```

1
2 public class VirtualPet {
3     private String name;
4     private int hunger;
5     private int cleanliness;
6     private int happiness;
7     private int health;
8     private int age;
9
10
11 public VirtualPet(String namef) {
12     //initialized variables
13     age = 0;
14     this.name = namef;
15     //randomly initialized variables
16     this.hunger = (int) (Math.random() * 10) + 1;
17     this.cleanliness = (int) (Math.random() * 10) + 1;
18     this.happiness = (int) (Math.random() * 10) + 1;
19     this.health = (int) (Math.random() * 10) + 1;
20 }
21
22 //getter methods
23 public String getName() {return name;}
24 public int getAge() {return age;}
25 public int getHunger() {return hunger;}
26 public int getCleanliness() {return cleanliness;}
27 public int getHappiness() {return happiness;}
28 public int getHealth() {return health;}
29
30 //setter methods
31 public void setName(String namef) { name = namef;}
32 public void setAge(int agef) { age = agef;}
33 public void setHunger(int hungerf) { hunger = hungerf;}
34 public void setCleanliness(int cleanlinessf) { cleanliness = cleanlinessf;}
35 public void setHappiness(int happinessf) { happiness = happinessf;}
36 public void setHealth(int healthf) { health = healthf;}
37
38 //stat update methods
39 public void feed() { //updates the hunger and outputs a random text dialog
40     setHunger(getHunger() + 1);
41     if((int) (Math.random()*2) == 0) {
42         System.out.println(getName() + " shouts with joy"
43             + " as you\npresent a watermelon slice. (+1 Hunger)");
44     }
45     else {
46         System.out.println(getName() + " has their stomach"
47             + " satisfied\nby the bowl of soup you offer. (+1 Hunger)");
48     }
49 }
50
51 public void wash() { //updates the cleanliness and outputs a random text dialog
52     setCleanliness(getCleanliness() + 1);
53     if((int) (Math.random()*2) == 0) {
54         System.out.println(getName() + " is sparkling clean"
55             + " after the\nlong bath you give them. (+1 Cleanliness)");
56     }

```

```

57         else {
58             System.out.println(getName() + " smells great after"
59                 + " the long\nshower they take. (+1 Cleanliness)");
60         }
61     }
62
63     public void play() { //updates the happiness and outputs a random text dialog
64         setHappiness(getHappiness() + 1);
65         if((int)(Math.random()*2) == 0) {
66             System.out.println(getName() + "'s eyes sparkle after"
67                 + " you\nfinish playing catch. (+1 Happiness)");
68         }
69         else {
70             System.out.println(getName() + " seems much more positive"
71                 + " after you\ngive them a big hug. (+1 Happiness)");
72         }
73     }
74
75     public void health() { //updates the health and outputs a random text dialog
76         setHealth(getHealth() + 1);
77         if((int)(Math.random()*2) == 0) {
78             System.out.println(getName() + " looks much healthier"
79                 + " after\ntheir long rest. (+1 Health)");
80         }
81         else {
82             System.out.println(getName() + "'s muscles are looking"
83                 + " larger\nafter exercising. (+1 Health)");
84         }
85     }
86
87     public void birthday() { //makes the pet a year older
88         age += 1;
89     }
90 }

```

```

1
2 public class Attribute_Tool {
3     //Checks to see if cleanliness is less than 5, if so it drops other stats
4     public static void checkCleanliness(VirtualPet petf) {
5         if(petf.getCleanliness() < 5) {
6             if((int)(Math.random()*2) == 0) {
7                 petf.setHealth(petf.getHealth() - 1);
8                 System.out.println(petf.getName() + " is starting to"
9                     + " smell bad. (-1 Health)");
10            }
11            else {
12                petf.setHappiness(petf.getHappiness() + 1);
13                System.out.println(petf.getName() + " is muddy from"
14                    + " rolling around\nin the dirt. (+1 Happiness)");
15            }
16        }
17    }
18
19    //Checks to see if happiness is less than 5, if so it drops other stats
20    public static void checkHappiness(VirtualPet petf) {
21        if(petf.getHappiness() < 5) {
22            if((int)(Math.random()*2) == 0) {
23                petf.setHunger(petf.getHunger() - 1);
24                System.out.println(petf.getName() + " is about to"
25                    + " gnaw his\nleg off. (-1 Hunger)");
26            }
27            else {
28                petf.setCleanliness(petf.getCleanliness() + 1);
29                System.out.println(petf.getName() + " forgot to put on"
30                    + " deodorant\nthis morning. (-1 Cleanliness)");
31            }
32        }
33    }
34
35    //Checks to see if the happiness is less than 5, if so it drops other stats
36    public static void checkHealth(VirtualPet petf) {
37        if(petf.getHealth() < 5) {
38            if((int)(Math.random()*2) == 0) {
39                petf.setHunger(petf.getHunger() - 1);
40                System.out.println(petf.getName() + " can hear their"
41                    + " stomach\ngrowling. (-1 Hunger)");
42            }
43            else {
44                petf.setCleanliness(petf.getCleanliness() + 1);
45                System.out.println(petf.getName() + " hasn't changed"
46                    + " their\nunderwear yet. (-1 Cleanliness)");
47            }
48        }
49    }
50
51    //Checks to see if the hunger is less than 5, if so it drops other stats
52    public static void checkHunger(VirtualPet petf) {
53        if(petf.getHunger() < 5) {
54            if((int)(Math.random()*2) == 0) {
55                petf.setHealth(petf.getHealth() - 1);
56                System.out.println(petf.getName() + " forgot to eat"

```

```

57         + " their\ndaily vitamins. (-1 Health)");
58     }
59     else {
60         petf.setHappiness(petf.getHappiness() - 1);
61         System.out.println(petf.getName() + " didn't get ice"
62             + " cream\nafter dinner. (-1 Happiness)");
63     }
64 }
65
66 }
67 //Checks to see if any stats are 0, if so then the pet is dead
68 public static boolean checkDeath(VirtualPet petf) {
69     if(petf.getHealth() <= 0 || petf.getCleanliness() <= 0 ||
70         petf.getHappiness() <= 0 || petf.getHunger() <= 0) {
71         return true;
72     }
73     return false;
74 }
75 }

```