

Universidad de las Fuerzas Armadas ESPE

Matriz - Sangolquí

Object Oriented Programming



HOMEWORKS

NRC:

14575

Estudiante:

Stefany Maricela Díaz Antun

Professor:

Edison Lascano

Noviembre 2023- Marzo 2024

-----> POLYMORPHISM by STEFANY DÍ

Animal:

Animal{id=1, scientificName=Iguanas,

Animal type: class ec.edu.espe.zooExe

Animal-->Turtle{Reptile{Vertebrate{

Animal{id=2, scientificName=Chelonioid

The screenshot shows an IDE window with two main panes. The top pane displays the output of a Java program, and the bottom pane shows the source code of the `Reptile.java` file.

Output Pane:

```
run:
-----> POLYMORPHISM by STEFANY DIAZ
Animal:
Animal{id=1, scientificName=Iguanas, bornDate=Sat Jan 13 11:33:50 ECT 2024, cageId=1}
Animal type: class ec.edu.espe.zooExercise.model.Animal
Animal-->Turtle{Reptile{Vertebrate{
Animal{id=2, scientificName=Chelonioidae, bornDate=Sat Jan 13 11:33:50 ECT 2024, cageId=2}spine=Exoskeleton, numberOfBones=380}typeOfReproduction=Oviparous}shell=Exoskeleton
Animal typeclass ec.edu.espe.zooExercise.model.Turtle
Animal-->Chameleon{Reptile{Vertebrate{
Animal{id=3, scientificName=Chamaeleonidae, bornDate=Sat Jan 13 11:33:50 ECT 2024, cageId=3}spine=Exoskeleton, numberOfBones=210}typeOfReproduction=Oviparous}color=Green
Animal typeclass ec.edu.espe.zooExercise.model.Chameleon
Other chameleonChameleon{Reptile{Vertebrate{
Animal{id=4, scientificName=Furcifer pardalis, bornDate=Sat Feb 02 00:00:00 ECT 3924, cageId=4}spine=Exoskeleton, numberOfBones=200}typeOfReproduction=Oviparous}color=Green
Chameleons change the color of their body to hide
Chameleon is regulating its temperature
Chameleon is regulating its temperature at26°C room temperature.

ZOO ANIMALS
[
Animal{id=1, scientificName=Iguanas, bornDate=Sat Jan 13 11:33:50 ECT 2024, cageId=1}, Turtle{Reptile{Vertebrate{
Animal{id=2, scientificName=Chelonioidae, bornDate=Sat Jan 13 11:33:50 ECT 2024, cageId=2}spine=Exoskeleton, numberOfBones=380}typeOfReproduction=Oviparous}shell=Exoskeleton
Animal{id=4, scientificName=Furcifer pardalis, bornDate=Sat Feb 02 00:00:00 ECT 3924, cageId=4}spine=Exoskeleton, numberOfBones=200}typeOfReproduction=Oviparous}color=Green
Total animals-->3
BUILD SUCCESSFUL (total time: 0 seconds)
```

Source Pane (Reptile.java):

```
1 package ec.edu.espe.zooExercise.model;
2
3 import java.util.Date;
4
5 /**
6  *
7  * @author Stefany Diaz, HoneyBadgers, DCCO - ESPE
8  */
9
10 public class Reptile extends Vertebrate{
11     private String typeOfReproduction;
12
13     public void regulateTemperature() {
14         System.out.println("The reptile regulates its temperature according to the environment");
15     }
16
17     @Override
18     public String toString() {
19         return "Reptile{" + super.toString() + "typeOfReproduction=" + typeOfReproduction + '}';
20     }
21
22     public Reptile(String typeOfReproduction, String spine, int numberOfBones, int id, String scientificName, Date bornDate, int cageId) {
23         super(spine, numberOfBones, id, scientificName, bornDate, cageId);
24         this.typeOfReproduction = typeOfReproduction;
25     }
26
27     /**
28      * @return the typeOfReproduction
29      */
30     public String getTypeOfReproduction() {
31         return typeOfReproduction;
32     }
33 }
```

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help Zoo_1 - Apache NetBeans IDE 19 Search (Ctrl-I)
<default config> 231,7/329,0MB
Animal.java Zoo.java Vertebrate.java Mammal.java
Source History
13 14
15 public class Zoo {
16     public static void main(String[] args) {
17         System.out.println("----->POLYMORPHISM by STEFANY DÍAZ");
18         Animal animal = new Mammal(numberOfMammaryGlands: 0, spine: "", numberOfBones: 0, id: 0, scientificName: "", new Date(), cageId: 0);
19         ArrayList<Animal> animals = new ArrayList<>();
20         animals.add(e: animal);
21         System.out.println("animal type is -->" + animal.getClass().getName());
22         System.out.println("animal--> " + animal);
23         System.out.println("animal type" + animal.getClass());
24
25         animal = new Platypus(poisonous: true, poisonGlands: 0, numberOfMammaryGlands: 0, spine: "spine small", numberOfBones: 0, id: 0, scientificName: "platypus scientific", new Date(), cageId: 0);
26         animals.add(e: animal);
27     }
28 }
Output - Zoo_1 (run)
----->POLYMORPHISM by STEFANY DÍAZ
animal type is -->ec.edu.espe.zoo.model.Mammal
animal--> Mammal[Vertebrate{
Animal[id=0, scientificName=, bornDate=Mon Jan 15 09:51:40 ECT 2024, cageId=0]spine=, numberOfBones=0]numberOfMammaryGlands=0}
animal typeclass ec.edu.espe.zoo.model.Mammal
animal--> Platypus[Mammal[Vertebrate{
Animal[id=0, scientificName=platypus scientific, bornDate=Mon Jan 15 09:51:40 ECT 2024, cageId=0]spine=spine small, numberOfBones=0]numberOfMammaryGlands=0]poisonous=true, poisonGlands=0}
animal typeclass ec.edu.espe.zoo.model.Platypus
animal--> Giraffe[Mammal[Vertebrate{
Animal[id=0, scientificName=girafus, bornDate=Mon Jan 15 09:51:40 ECT 2024, cageId=0]spine=big spine, numberOfBones=0]numberOfMammaryGlands=0}
animal typeclass ec.edu.espe.zoo.model.Giraffe
other giraffeGiraffe[Mammal[Vertebrate{
Animal[id=2, scientificName=Giraffe, bornDate=Thu Feb 01 00:00:00 ECT 3923, cageId=0]spine=long, numberOfBones=206]numberOfMammaryGlands=4}
Brushing the neck of the giraffe
Giraffe is breastfeeding
Giraffe is breastfeeding (re call number=2)
```

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help Zoo_1 - Apache NetBeans IDE 19 Search (Ctrl-I)
<default config> 196,7/329,0MB
Animal.java Zoo.java Vertebrate.java Mammal.java
Source History
4 import ec.edu.espe.zoo.model.Giraffe;
5 import ec.edu.espe.zoo.model.Platypus;
6 import ec.edu.espe.zoo.model.Vertebate;
7 import java.util.ArrayList;
8 import java.util.Date;
9
10 /**
11  *
12  * @author Stefany Diaz, HoneyBadgers, DCCO - ESPE
13  */
14 public class Zoo {
15     public static void main(String[] args) {
16         System.out.println("----->POLYMORPHISM by STEFANY DÍAZ");
17         Animal animal = new Vertebrate("Light", 0, 0, "Berb", new Date(), 0);
18         ArrayList<Animal> animals = new ArrayList<>();
19         animals.add(e: animal);
20         System.out.println("animal type is -->" + animal.getClass().getName());
21         System.out.println("animal--> " + animal);
22         System.out.println("animal type" + animal.getClass());
23
24         animal = new Platypus(poisonous: true, poisonGlands: 0, numberOfMammaryGlands: 0, spine: "spine small", numberOfBones: 0, id: 0, scientificName: "platypus scientific", new Date(), cageId: 0);
25         animals.add(e: animal);
26         System.out.println("animal--> " + animal);
27         System.out.println("animal type" + animal.getClass());
28
29         animal = new Giraffe(numberOfMammaryGlands: 0, spine: "big spine", numberOfBones: 0, id: 0, scientificName: "girafus", new Date(), cageId: 0);
30         System.out.println("animal--> " + animal);
31         System.out.println("animal type" + animal.getClass());
32         //animal.brushNeck();
33
34         Giraffe giraffe = new Giraffe(numberOfMammaryGlands: 4, spine: "long", numberOfBones: 206, id: 2, scientificName: "Giraffe", new Date(year: 2023, month: 1, date: 1), cageId: 0);
35     }
36 }
```

```
11
12 * @author Stefany Diaz, HoneyBadgers, DCCO - ESPE
13 */
14
15 ----->POLYMORPHISM by STEFANY DÍAZ");
16 (Alt-Enter shows hints)
17 Animal animal = new Vertebrate("Light", 0, 0, "Berb", new Date(), 0);
18 ArrayList<Animal> animals = new ArrayList<>();
19 animals.add(e: animal);
20 System.out.println("animal type is -->" + animal.getClass().getName());
21 System.out.println("animal--> " + animal);
22 System.out.println("animal type" + animal.getClass());
```

```

12  *
13  * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
14  */
15  public class Zoo {
16      public static void main(String[] args) {
17          System.out.println("-----> ABSTRACT CLASSES by STEFANY DÍAZ");
18          Animal animal = new Mammal(numberOfMammaryGlands: 0, spine: "", numberOfBones: 0, id: 0, scientificName: "");
19          ArrayList<Animal> animals = new ArrayList<>();
20          animals.add(e: animal);
21          System.out.println("animal type is -->" + animal.getClass().getName());
22          System.out.println("animal--> " + animal);
23          System.out.println("animal type" + animal.getClass());
24
25          animal = new Platypus(poisoness: true, poisonGlands: 0, numberOfMammaryGlands: 0, spine: "spine sma

```

Output - Zoo_1 (run) ×

```

run:
-----> ABSTRACT CLASSES by STEFANY DÍAZ
animal type is -->ec.edu.espe.zoo.model.Mammal
animal--> Mammal{Vertebrate{
Animal{id=0, scientificName=, bornDate=Mon Jan 15 09:56:12 ECT 2024, cageId=0}spine=, numberOfBones=0}numb

```

```

* @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
*/

```

Mammal is abstract; cannot be instantiated

```

-----> ABSTRACT CLASSES by STEFANY DÍAZ
Animal animal = new Mammal(0, "", 0, 0, "", new Date(), 0);
ArrayList<Animal> animals = new ArrayList<>();
animals.add(e: animal);
System.out.println("animal type is -->" + animal.getClass().getName());

```

```

12  *
13  * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
14  */
15  public class Zoo {
16      public static void main(String[] args) {
17          System.out.println("-----> ABSTRACT CLASSES by STEFANY DÍAZ");
18          Animal animal = new Giraffe(numberOfMammaryGlands: 0, spine: "heavy", numberOfBones: 0, id: 0, scientificName: "gir");
19          ArrayList<Animal> animals = new ArrayList<>();
20          animals.add(e: animal);
21          System.out.println("animal type is -->" + animal.getClass().getSimpleName());
22          System.out.println("animal--> " + animal);
23          System.out.println("animal type" + animal.getClass());
24
25          animal = new Platypus(poisoness: true, poisonGlands: 0, numberOfMammaryGlands: 0, spine: "spine sma

```

Output - Zoo_1 (run) ×

```

run:
-----> ABSTRACT CLASSES by STEFANY DÍAZ
animal type is -->Giraffe
animal--> Giraffe{Mammal{Vertebrate{
Animal{id=0, scientificName=gir, bornDate=Mon Jan 15 10:00:38 ECT 2024, cageId=0}spine=heavy,
animal typeclass ec.edu.espe.zoo.model.Giraffe

```

```
5  /**
6   Platypus is not abstract and does not override abstract method feed() in Animal
7   ----
8   (Alt-Enter shows hints)
9
10 public class Platypus extends Mammal{
11     private boolean poisoness;
12     private int poisonGlands;
13
14 7   * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
15 8   */
16 9   Giraffe is not abstract and does not override abstract method feed() in Animal
17 10  ----
18 11  (Alt-Enter shows hints)
19
20 public class Giraffe extends Mammal{
21
22 13
23 14     public Giraffe(int numberOfMammaryGlands, String spine, int
24 15         super(numberOfMammaryGlands, spine, numberOfBones, id, s
25 16     }
```

