Universidad de las Fuerzas Armadas ESPE

Matriz - Sangolquí

Object Oriented Programming

WORKSHOP

NRC:

14575

Estudiante:

Stefany Maricela Díaz Antun

Professor:

Edison Lascano

Noviembre 2023- Marzo 2024

```
11      *
12      * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
13      */
14
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()
18          ArrayList<Animal> animals = new ArrayList<>();
19          animals.add(e: animal);
20          System.out.println("animal type is -->" + animal.getClass().getN
21          System.out.println("animal--> " + animal);
22          System.out.println("animal type" + animal.getClass());
```

Vertebrate is abstract; cannot be instantiated

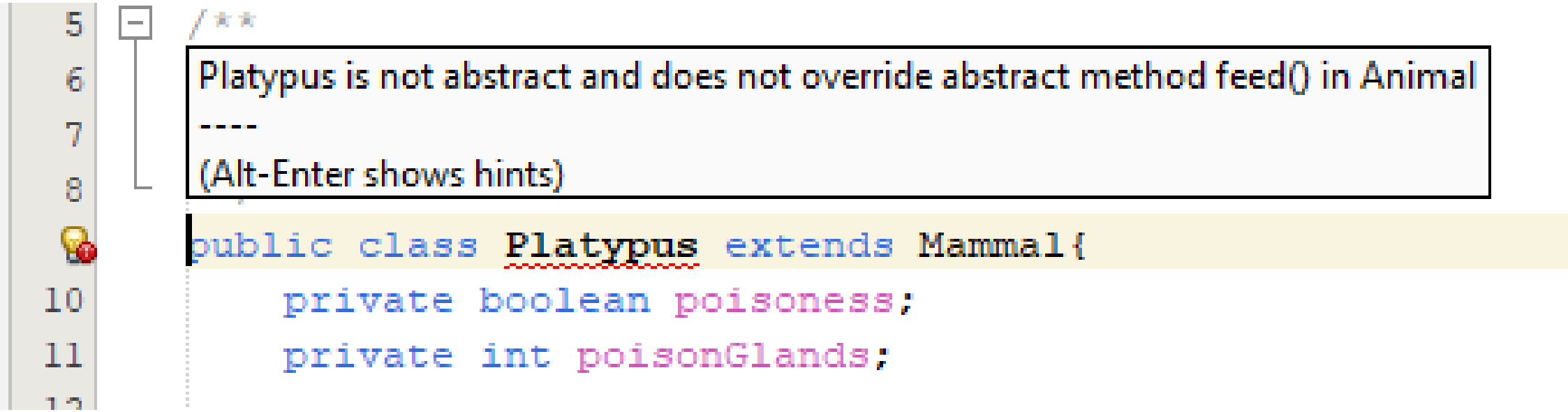
(Alt-Enter shows hints)

```
 * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
 */
Mammal is abstract; cannot be instantiated
----
(Alt-Enter shows hints)
                                    tring[] args) {
                                    ------> ABSTRACT CLASSES by STEFANY DíAZ"
        Animal animal = new Mammal(0, "", 0, 0, "", new Date(), 0);
        ArrayList<Animal> animals = new ArrayList<>();
        animals.add(e: animal);
```

```java
 *
 * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
 */
public class Zoo {
    public static void main(String[] args) {
        System.out.println(x: "------> ABSTRACT CLASSES by STEFANY DíAZ");
        Animal animal = new Giraffe(numberOfMammaryGlands: 0, spine: "heavy", numberOfBones:
        ArrayList<Animal> animals = new ArrayList<>();
        animals.add(e: animal);
        System.out.println("animal type is -->" + animal.getClass().getSimpleName
        System.out.println("animal--> " + animal);
        System.out.println("animal type" + animal.getClass());

        animal = new Platypus(poisoness: true, poisonGlands: 0, numberOfMammaryGlands: 0, spine
```

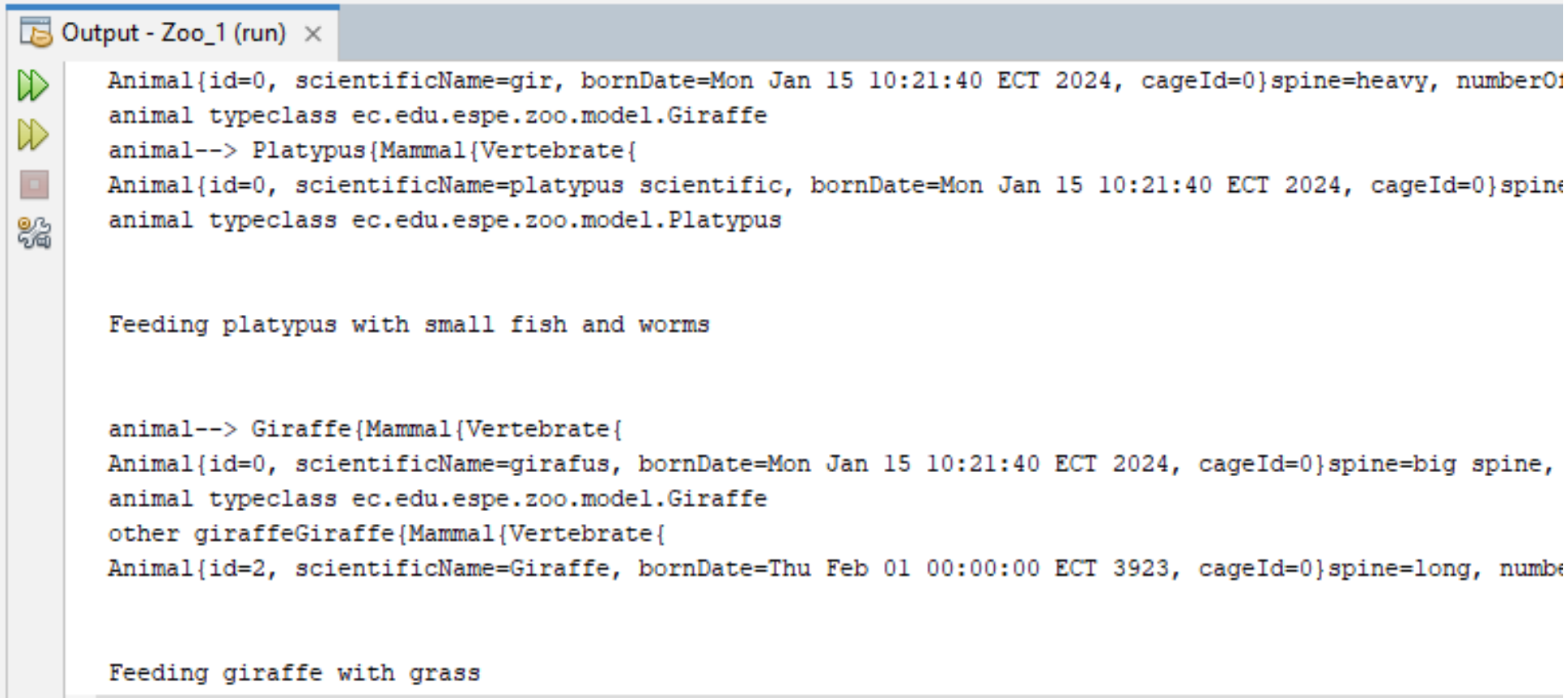
Output - Zoo_1 (run)

```
run:
------> ABSTRACT CLASSES by STEFANY D?AZ
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
/**
Platypus is not abstract and does not override abstract method feed() in Animal
---
(Alt-Enter shows hints)
public class Platypus extends Mammal{
10
private boolean poisoness;
11
private int poisonGlands;

```
 * @author Stefany Díaz, HoneyBadgers, DCCO - ESPE
 */
Giraffe is not abstract and does not override abstract method feed() in Animal
----
(Alt-Enter shows hints)
public class Giraffe extends Mammal{

    public Giraffe(int numberOfMammaryGlands, String spine, int
        super(numberOfMammaryGlands, spine, numberOfBones, id, s
    }
```


Animal.java
Zoo.java
Vertebrate.java
Mammal.java
Platypus.java
Giraffe.java
Source
History
System.out.println("\n");
animal.feed();
System.out.println("\n");
animal = new Giraffe(numberOfMammaryGlands: 0, spine: "big spine", numberOfBones: 0, id: 0,
System.out.println("animal--> " + animal);
System.out.println("animal type" + animal.getClass());
//animal.brushNeck();
Giraffe giraffe = new Giraffe(numberOfMammaryGlands: 4, spine: "long", numberOfBones: 206,


Output - Zoo_1 (run)
Animal{id=0, scientificName=gir, bornDate=Mon Jan 15 10:21:40 ECT 2024, cageId=0}spine=heavy, numberOf
animal typeclass ec.edu.espe.zoo.model.Giraffe
animal--> Platypus{Mammal{Vertebrate{
Animal{id=0, scientificName=platypus scientific, bornDate=Mon Jan 15 10:21:40 ECT 2024, cageId=0}spine
animal typeclass ec.edu.espe.zoo.model.Platypus
Feeding platypus with small fish and worms
animal--> Giraffe{Mammal{Vertebrate{
Animal{id=0, scientificName=girafus, bornDate=Mon Jan 15 10:21:40 ECT 2024, cageId=0}spine=big spine,
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, numbe
Feeding giraffe with grass

```
38
39          animal.assignCage(cageId: 20);
40          Giraffe giraffe = new Giraffe(numberOfMammaryGlands: 4, spine: "long", numberOf
41          animals.add(e: giraffe);
42          System.out.println("other giraffe" + giraffe);
43          giraffe.assignCage(cageId: 25);
44          System.out.println(x: "\n");
45          giraffe.feed();
46          System.out.println(x: "\n");
47
48          giraffe.brushNeck();
49
```

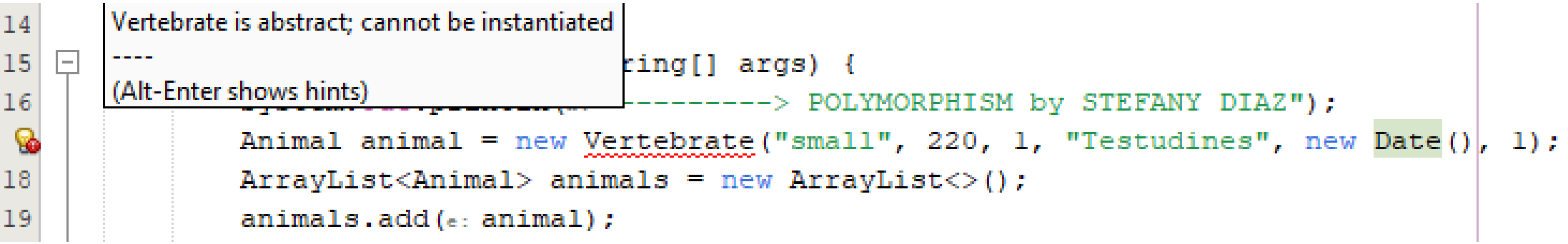
Output - Zoo_1 (run)

```
Feeding platypus with small fish and worms


animal--> Giraffe{Mammal{Vertebrate{
Animal{id=0, scientificName=girafus, bornDate=Mon Jan 15 10:47:40 ECT 2024, cageId=0}sp:
animal typeclass ec.edu.espe.zoo.model.Giraffe
Assigning cage number --> 20
other giraffeGiraffe{Mammal{Vertebrate{
Animal{id=2, scientificName=Giraffe, bornDate=Thu Feb 01 00:00:00 ECT 3923, cageId=3}sp:
Assigning cage number --> 25


Feeding giraffe with grass
```

```
12      */
13
14      public static void main(String[] args) {
15          System.out.println("---------> POLYMORPHISM by STEFANY DIAZ");
16          Animal animal = new Animal(1, "Iguanas", new Date(), 1);
17          ArrayList<Animal> animals = new ArrayList<>();
18          animals.add(e: animal);
```

Animal is abstract; cannot be instantiated

(Alt-Enter shows hints)


Vertebrate is abstract; cannot be instantiated

(Alt-Enter shows hints)
ring[] args) {
---------> POLYMORPHISM by STEFANY DIAZ");
Animal animal = new Vertebrate("small", 220, 1, "Testudines", new Date(), 1);
ArrayList<Animal> animals = new ArrayList<>();
animals.add(e: animal);

```
15
16      String[] args) {
17          "---------> ABSTRACT CLASSES by STEFANY DIAZ");
18          Animal animal = new Reptile("Oviparous", "Exoeskeleton", 350, 2, "Lacertilia", new Date(), 2);
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());
```

Reptile is abstract; cannot be instantiated

(Alt-Enter shows hints)