|  |  |
| --- | --- |
| **Practicum Case** |  |
| COMP6708  Object Oriented Programming |
| **Computer Science** | **O221-COMP6708-RV01-08** |
| ***Valid on*** *Odd Semester Year 2021/2022* | **Revision 00** |

## Learning Outcomes

* The additional features of OOP
* A program using additional features of OOP

## Topic

* Polymorphism

## Subtopics

* Introduction to Polymorphism
* Final Method
* Ad hoc Polymorphism (Overloading)
* Overriding

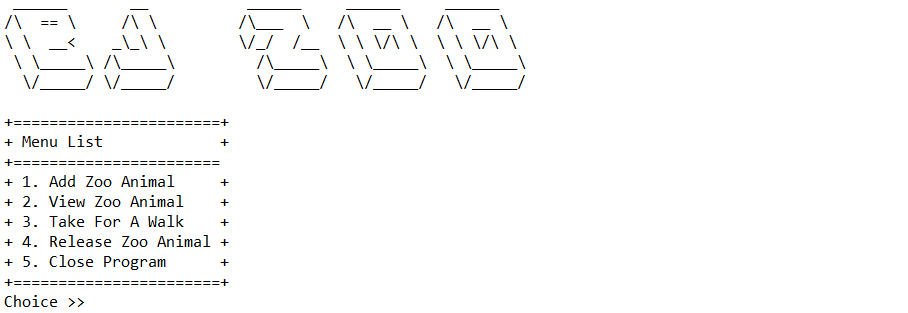
## Soal

*Case*

**BlueJack Zoo**

**BlueJack Zoo** is the current trending zoo in BlueJackVille. Many investors had already planned to offer them many of those endangered species from **Feline** and **Canine** genus. This is already the best possible solution toward the brimming and bright future of BlueJack Zoo. Unfortunately, they haven’t developed a convenient app that could instantly note down every animal transaction that has ever happen in the zoo (i.e., **rereleasing them in the wild**, **taking in baby animals**, **and so on**). Now you, as the most trustful programmer, are asked to build a simple yet convenient app to solve their problem using **Java Programming Language**.

* In the beginning, the program will show the title



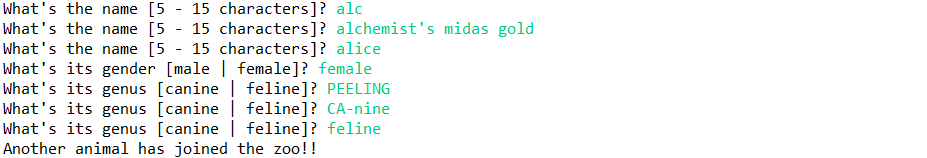
* The program will consist of **5 menus**:

1. Add Zoo Animal
2. View Zoo Animal
3. Take For A Walk
4. Release Zoo Animal
5. Close Program

* The program will ask the user to input **choose menu**, which must be inputted **between 1** and **5**



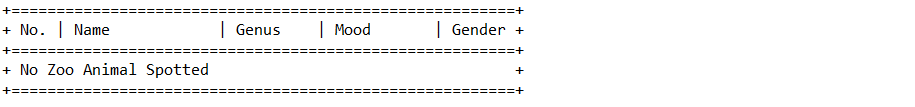
* If the user chooses menu 1, then:
* The program will ask the user to input **animal detail**, which consists of:
* **Name**, which **length** must be **between** **5** and **15** **characters**
* **Gender**, which must be **between** **male** and **female** (**case sensitive**)
* **Genus**, which must be **between** **canine** and **feline** (**case sensitive**)



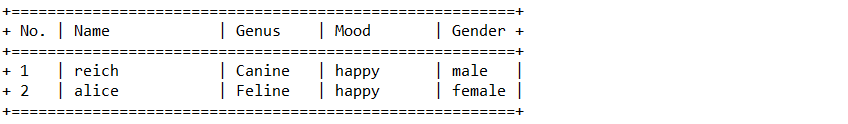
* If the genus type is **canine**, then the program will ask the user to input **its walking speed**, which must be **between** **1** and **80**

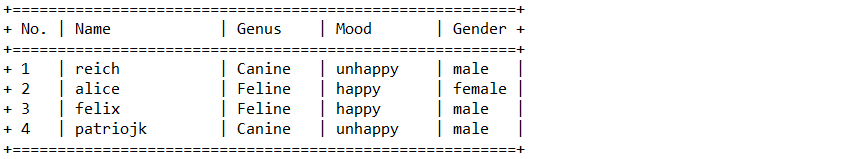


* After all required input has been filled, **add the data to a vector/array** list
* After that, display a message indicating the corresponding animal has been added to the zoo and return to the main menu
* If the user chooses menu 2, then:
* If there are **no animal**, display list with empty content

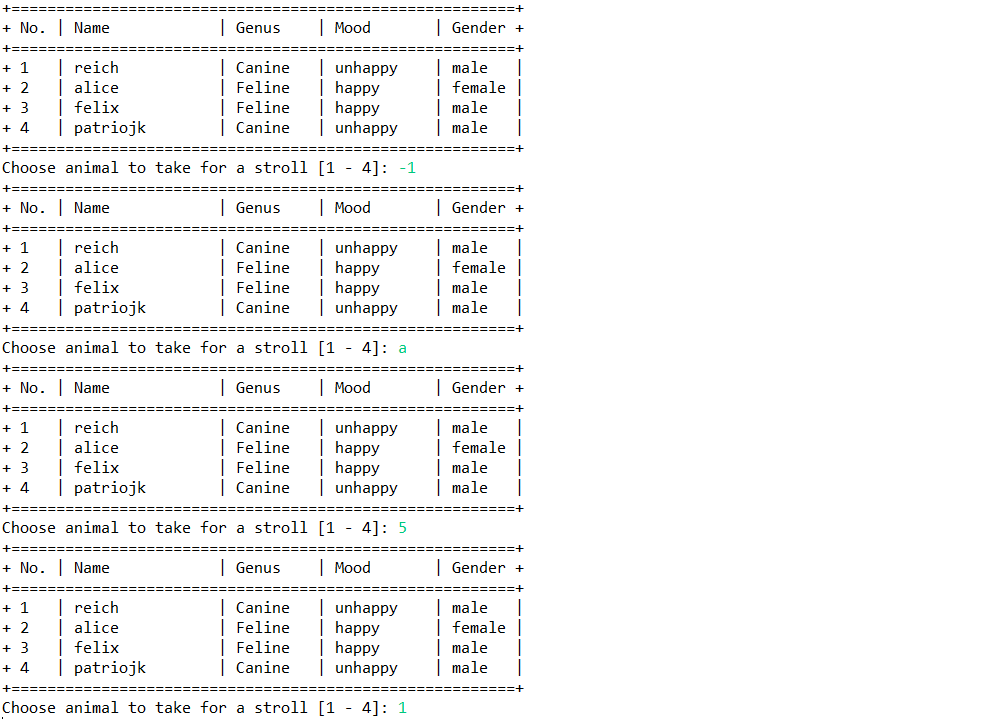


* Otherwise, display list of current active zoo animals
* **Mood** is gotten from the relationship from each genus towards its kind
* If the genus type is “**canine**”:
* If there’s only **1** canine in the zoo, set mood to“**happy**”
* Else, if the current canine is **less than** the most canine ever held in the zoo, set the mood to “**happy**”
* Else set the mood to “**unhappy**”
* If the genus type is “**feline**”:
* If there’s only **1** feline in the zoo, set mood to“**unhappy**”
* Else, if the current feline is **less than** the most feline ever held in the zoo, set the mood to “**unhappy**”
* Else set the mood to “**happy**”





* After that, return to the main menu
* If the user chooses menu 3, then:
* If there are **no animal**, display list with **empty** content
* Otherwise, the program will:
* **Display** list of current active zoo animals
* Ask the user to input:
* Choice option, which must be **numeric** and **between** **1** and the **count of active zoo animal**
* Roaming time, which must be **numeric** and **between** **0** and **50**
* After fulfilling all validation, **update** the selected animal roam time with following condition:
* If the roaming time is **0**, then set the roam time to **60**
* Otherwise, set the roaming time to the **inputted value**



* After that, display total roaming range
* If the chosen animal is a “**feline**”
* Generate a **random** number between **0** and **1**
* If the number is 0, then show the message “**Felines doesn’t feel like roaming at all today..**”



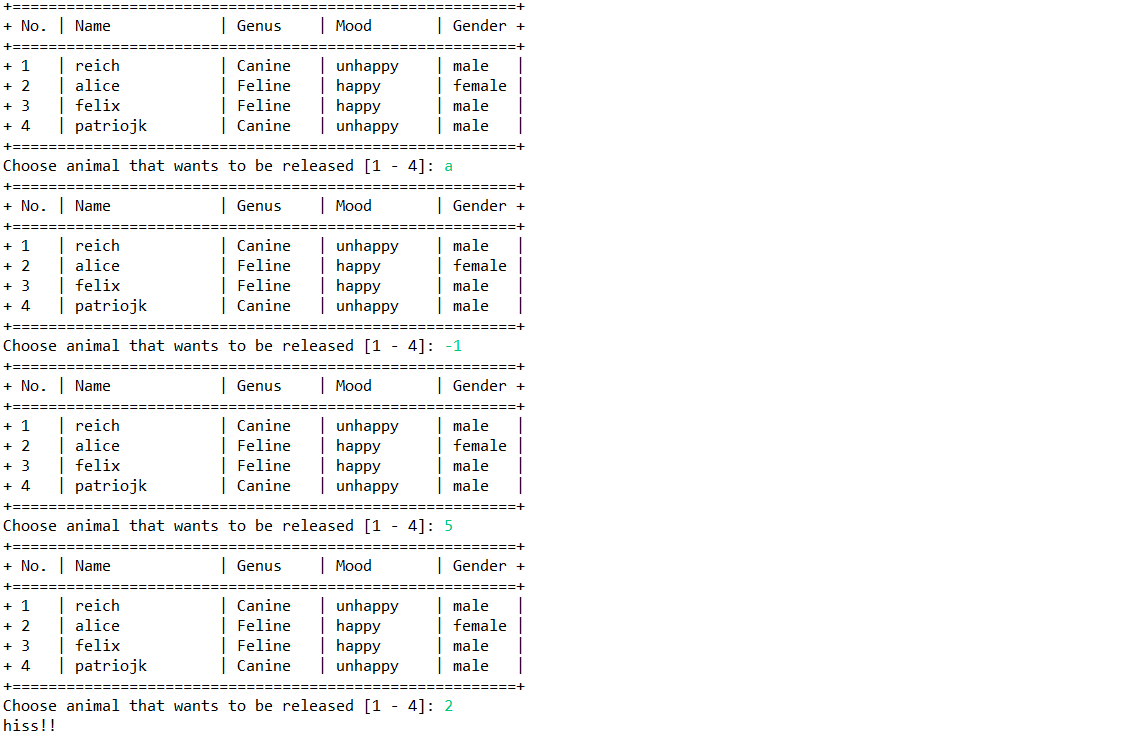
* If the number is 1:
* Generate total range based on the following formula

* If the roam time is 0, set roam time to 30
* Otherwise, **divide** the inputted duration by 2
* If the chosen animal is a “**canine**”
* Generate total range based on the following formula

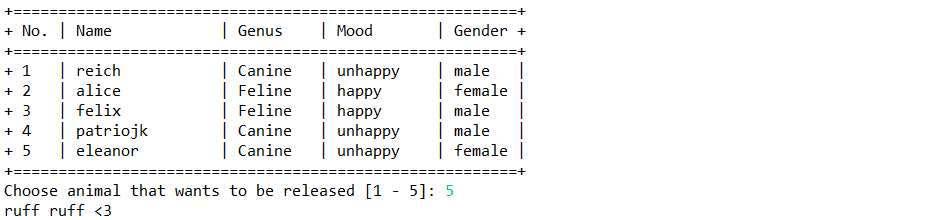
* If the roam time is 0, set roam time to 60
* Otherwise, set the roam time based on roaming time

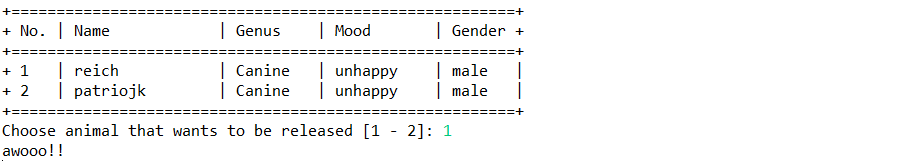


* Finally, return to the main menu
* If user choose menu 4, then:
* If there are **no animal**, display list with emptycontent
* Otherwise, the program will:
* **Display** list of current active zoo animals
* Ask the user to input **choice option**, which must be **numeric** and between **1** and the **count of active zoo animal**

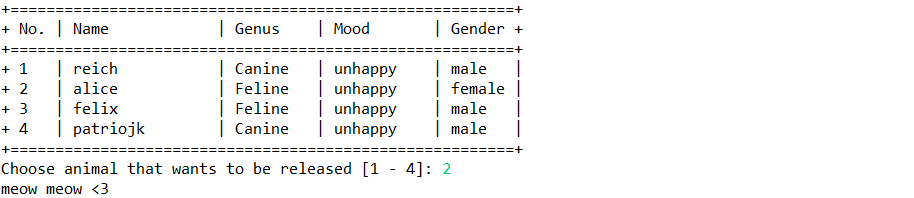


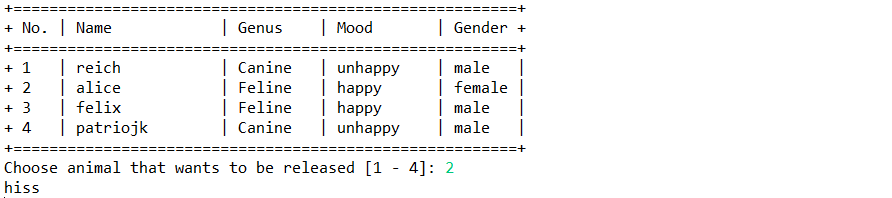
* After that, display the parting message from the animal
* Generate a **random** number with the size of **the current active** zoo animal list
* If the chosen animal is a “**canine**”
* If the random number index on the list is also a **canine**
* Validate the random number index’s gender
* If both genders are different from each other, display message“**ruff ruff <3**”and remove **both** animal
* Else, display message“**awooo!!**” and remove the **picked** animal





* If the chosen animal is a “**feline**”
* If the random number index on the list is also a **feline**
* Validate the random number index’s gender
* If both genders are different from each other, display message“**meow meow <3**”and remove **both** animal
* Else, display message“**hiss!!**” and remove the **picked** animal





* Finally, return to the main menu
* If user choose menu 5, then the program will be closed

**Please ask your teaching assistant if there are any related questions.**