Jeffrey Ngo

CS 162

Project 2

October 27, 2019

Design

Menu

a. Create a menu to start or exit the zoo game
b. Input validation for user input
c. If start game, ask user to buy 1 or 2 of each animal to start game
d. Ask user if they want to buy another animal before day ends
e. Ask user to keep playing after each day or exit

Input Validation

a. Input must be greater than zero, can't negative, no characters, letters.
b. Handle float values
c. Must no crash program

Animal Class

a. Contains setters and getters
b. Age of each animal (baby < 3 days <= adult)
c. Cost of each animal (tiger $10,000, penguin $1000, turtle $100)
d. Number of babies of each animal (tiger 1, penguin 5, turtle 10)
e. Base food cost for each animal (tiger $50, penguins $10, turtle $5)
f. Payoff of each animal (tiger $2000, penguin $100, turtle $5)

Zoo Class

a. Contains start function if user chooses to play
b. Cash balance starts at $100,000
c. To start user must buy 1 or 2 of each animal (tiger, penguin, turtle)
d. Set each animals' age to 1
e. Deduct animals cost from cash balance when bought
f. Set array capacity of each animal to 10
g. Expand the array if animal reaches the capacity

h. Each day increase animals age, feed animal and deduct cost from balance, and pick a random event to happen
i. Random events sickness, attendance boom, baby animal is born, or nothing.
j. Pick randomly out of the three animals to it reproduce.

Tiger Class

a. Inherits from the base animal class

Penguin Class

a. Inherits from the base animal class

Turtle Class

a. Inherits from the base animal class

Changes and Problems encountered

I created a separate function to setAnimalAge, setAnimalCost, setNumberOfBabies, setAnimalBaseFoodCost and setAnimalPayoff for each animal. This allowed me to call the function each time an animal is bought instead of typing the whole thing each time. This helped shortened my project by eliminating repetition. I also found breaking down functions into smaller sections helpful in reaching what I initially wanted to function to do. For example, at first, I tried creating the random event function as a single function but found it lengthy and hard to manage. I decided to create separate functions for each event and a function to call to those events if they were chosen. I find it much easier to read and follow.

Reflection

I found this project difficult as the first and took a lot of time. The areas that I struggled with the most with was the arrays. I believe this project focused on a good amount on arrays. We had to modify them by adding to arrays, resize it if it reaches capacity, and loop through arrays to get the necessary information to increase, age, food cost, and profit. I believe this project has helped me get a better understanding of arrays and how to modify them.

Test Plan

Test Plan	Expected	Actual
Menu	a. Start and exit game b. Ask user to buy starting number of animals c. Ask user to buy another animal d. Ask user to keep playing e. Input validation	Created 4 different menu Starts or exit game. If game starts, ask user to buy 1 or 2 of each animal. Ask user to buy another animal Ask user to keep playing.
Input validation	a. Must be greater than zero b. Must be positive c. Must exclude characters and letters d. Handle float value e. Doesn't crash	I expected the input validation to not handle float values since I used the same method on previous labs that didn't handle float, but it seems to handle float values.
Animal Class	a. Setters and getters for: b. Age c. Cost d. Number of babies e. Food cost f. payoff	Create a class that has setters and getters that can be inherited by other classes such as the tiger, penguin and turtle class
Tiger Class	a. Inherits from the base class Animal	Inherits from the base class Animal
Penguin Class	a. Inherits from the base class Animal	Inherits from the base class Animal
Turtle Class	a. Inherits from the base class Animal	Inherits from the base class Animal
Zoo Class	a. Game flow menus and function to start the game b. Buy Animals c. Feed animals d. Increase age e. Contains function that set info f. Modify arrays by resizing	Starts with $100,000 User starts by buying 1 or 2 of each animal Deduct cost from cash balance Deduct feeding cost Increase age of animal so that they can reproduce Resize array if capacity reaches limit Removes sick animals

	g. Random event functions	Bonus cash from attendance User can exit the program