import java.util.ArrayList;

import java.util.Scanner;

public class Driver {

private static ArrayList<Dog> dogList = new ArrayList<Dog>();

//do array for monkey list and species

private static ArrayList<Monkey> monkeyList = new ArrayList<Monkey>();

private static ArrayList<String> monkeySpecies = new ArrayList<String>();

// Instance variables (if needed)

//new dog object

Object listDog = new Object();

public static void main(String[] args) {

initializeDogList();

initializeMonkeyList();

//add array of monkey species

initializeMonkeySpecies();

// Add a loop that displays the menu, accepts the users input

// and takes the appropriate action.

// For the project submission you must also include input validation

// and appropriate feedback to the user.

// Hint: create a Scanner and pass it to the necessary

// methods

Scanner scnr = new Scanner(System.in);

String userInput = scnr.nextLine();

//display menu

displayMenu();

//while input != quit

while (!userInput.equalsIgnoreCase("Q")) {

if(userInput.equals("1")) {

intakeNewDog(scnr);

userInput = scnr.nextLine();

}

if (userInput.equals("2")) {

intakeNewMonkey(scnr);

userInput = scnr.nextLine();

}

if (userInput.equals("3")) {

reserveAnimal(scnr);

userInput = scnr.nextLine();

}

// Hint: Menu options 4, 5, and 6 should all connect to the printAnimals() method.

if (userInput.equals("4")) {

printAnimals("dog");

userInput = scnr.nextLine();

}

if (userInput.equals("5")) {

printAnimals("monkey");

userInput = scnr.nextLine();

}

if (userInput.equals("6")) {

printAnimals("available");

userInput = scnr.nextLine();

}

//print invalid entry

else {

System.out.println("You entered an invalid input. Please try again.");

displayMenu();

userInput = scnr.nextLine();

}

}

//end while

//if user quits message

if(userInput.equalsIgnoreCase("q")) {

System.out.println("Thank you for using program. Goodbye.");

scnr.close();

System.exit(0);

}

}

//end main

// This method prints the menu options

public static void displayMenu() {

System.out.println("\n\n");

System.out.println("\t\t\t\tRescue Animal System Menu");

System.out.println("[1] Intake a new dog");

System.out.println("[2] Intake a new monkey");

System.out.println("[3] Reserve an animal");

System.out.println("[4] Print a list of all dogs");

System.out.println("[5] Print a list of all monkeys");

System.out.println("[6] Print a list of all animals that are not reserved");

System.out.println("[q] Quit application");

System.out.println();

System.out.println("Enter a menu selection");

}

//end display menu

// Adds dogs to a list for testing

public static void initializeDogList() {

Dog dog1 = new Dog("Spot", "German Shepherd", "male", "1", "25.6", "05-12-2019", "United States", "intake", false, "United States");

Dog dog2 = new Dog("Rex", "Great Dane", "male", "3", "35.2", "02-03-2020", "United States", "Phase I", false, "United States");

Dog dog3 = new Dog("Bella", "Chihuahua", "female", "4", "25.6", "12-12-2019", "Canada", "in service", true, "Canada");

dogList.add(dog1);

dogList.add(dog2);

dogList.add(dog3);

}

//end dog list

// Adds monkeys to a list for testing

//Optional for testing

public static void initializeMonkeyList() {

//complete monkey list - similar to dog list for testing

/\*name, gender, age, weight, acquisitionDate, acquisitionCountry, trainingStatus,

reserved, inServiceCountry, tailLength, bodyLength, species, height

\*/

Monkey monkey1 = new Monkey ("Tre", "male", "22", "200", "08-31-2022", "United States", "Phase 1", false, "United States", "3", "5", "Tamarin", "6" );

//use monkey1

monkeyList.add(monkey1);

}

//end monkey list

//create species list

public static void initializeMonkeySpecies() {

//capuchin, guenon, macaque,marmoset, squirrel monkey, tamarin

String monkeySpecies1 = new String ("Capuchin");

String monkeySpecies2 = new String ("Guenon");

String monkeySpecies3 = new String ("Macaque");

String monkeySpecies4 = new String ("Marmoset");

String monkeySpecies5 = new String ("Squirrel Moneky");

String monkeySpecies6 = new String ("Tamarin");

//add them

monkeySpecies.add(monkeySpecies1);

monkeySpecies.add(monkeySpecies2);

monkeySpecies.add(monkeySpecies3);

monkeySpecies.add(monkeySpecies4);

monkeySpecies.add(monkeySpecies5);

monkeySpecies.add(monkeySpecies6);

}

// Complete the intakeNewDog method

// The input validation to check that the dog is not already in the list

// is done for you

public static void intakeNewDog(Scanner scanner) {

System.out.println("What is the dog's name?");

String name = scanner.nextLine();

for(Dog dog: dogList) {

if(dog.getName().equalsIgnoreCase(name)) {

System.out.println("\n\nThis dog is already in our system\n\n");

return; //returns to menu

}

//end if

// Add the code to instantiate a new dog and add it to the appropriate list

else {

try {

//set dog constructors

/\*name, breed, gender, age, weight, acquisitionDate, acquisitionCountry,

trainingStatus, reserved, inServiceCountry

\*/

dog.setName(name);

System.out.println("What is the dog's breed?");

dog.setBreed(scanner.nextLine());

System.out.println("What is the dog's gender?");

dog.setGender(scanner.nextLine());

System.out.println("What is the dog's age?");

dog.setAcquisitionDate(scanner.nextLine());

System.out.println("What is the dog's weight?");

dog.setWeight(scanner.nextLine());

System.out.println("What is the dog's acquisition date?");

dog.setAcquisitionDate(scanner.nextLine());

System.out.println("What is the dog's acquisition country?");

dog.setAcquisitionLocation(scanner.nextLine());

System.out.println("What is the dog's training status?");

dog.setTrainingStatus(scanner.nextLine());

System.out.println("Is the dog reserved?");

dog.setReserved(scanner.nextBoolean());

//boolean if statement

if (dog.getReserved() == true) {

System.out.println("What country is the dog reserved in?");

dog.setInServiceCountry(scanner.nextLine());

}

//display menu

displayMenu();

return;

}

//end try

//need a 'finally' to complete block statement

catch (Exception except) {

System.out.println(except.getMessage());

System.out.println("Try again.");

displayMenu();

return;

}

//end catch

}

//end else

}

//end for

}

//end intake new dog

// Complete intakeNewMonkey

//Instantiate and add the new monkey to the appropriate list

// For the project submission you must also validate the input

// to make sure the monkey doesn't already exist and the species type is allowed

public static void intakeNewMonkey(Scanner scanner) {

//change output message

//copy from dog intake -- change all 'dog' to 'monkey'

System.out.println("What is the monkey's name?");

String name = scanner.nextLine();

for(Monkey monkey: monkeyList) {

if(monkey.getName().equalsIgnoreCase(name)) {

System.out.println("\n\nThis monkey is already in our system\n\n");

return; //returns to menu

}

//end if

// Add the code to instantiate a new monkey and add it to the appropriate list

else {

try {

//set monkey constructors

/\*name, gender, age, weight, acquisitionDate, acquisitionCountry,

trainingStatus, reserved, inServiceCountry, tailLength, bodyLength,

species, height

\*/

monkey.setName(name);

System.out.println("What is the monkey's species?");

String species = scanner.nextLine().toLowerCase();

//if invalid species

if (!monkeySpecies.contains(species)) {

throw new Exception ("You entered an invalid species. Please try again.");

}

else {

monkey.setSpecies(species);

}

System.out.println("What is the monkey's gender?");

monkey.setGender(scanner.nextLine());

System.out.println("What is the monkey's age?");

monkey.setAcquisitionDate(scanner.nextLine());

System.out.println("What is the monkey's weight?");

monkey.setWeight(scanner.nextLine());

System.out.println("What is the monkey's acquisition date?");

monkey.setAcquisitionDate(scanner.nextLine());

System.out.println("What is the monkey's acquisition country?");

monkey.setAcquisitionLocation(scanner.nextLine());

System.out.println("What is the monkey's training status?");

monkey.setTrainingStatus(scanner.nextLine());

System.out.println("Is the monkey reserved?");

monkey.setReserved(scanner.nextBoolean());

//boolean if statement

if (monkey.getReserved() == true) {

System.out.println("What country is the monkey reserved in?");

monkey.setInServiceCountry(scanner.nextLine());

}

//display menu

displayMenu();

return;

}

//end try

//need a 'finally' to complete block statement

catch (Exception except) {

System.out.println(except.getMessage());

System.out.println("Try again.");

displayMenu();

return;

}

//end catch

}

//end else

}

//end for

}

//end intake new monkey

// Complete reserveAnimal

// You will need to find the animal by animal type and in service country

public static void reserveAnimal(Scanner scanner) {

//change output message

System.out.println("Select the species of animal.");

String species = scanner.nextLine();

//if it equals 'dog'

if (species.equalsIgnoreCase("dog")) {

for (Dog dog: dogList) {

System.out.println("Enter name of the dog.");

String name = scanner.nextLine();

if (dog.getName().equalsIgnoreCase(name)) {

System.out.println("Set if dog is reserved. True or false?");

dog.setReserved(scanner.nextBoolean());

return;

}

//end if

else {

System.out.println("Dog is not in program.");

return;

}

//end else

}

//end for

}

//end if

//if it equals monkey

else if (species.equalsIgnoreCase("monkey")) {

//copy from dog if

for (Monkey monkey: monkeyList) {

System.out.println("Enter name of the monkey.");

String name = scanner.nextLine();

if (monkey.getName().equalsIgnoreCase(name)) {

System.out.println("Set if monkey is reserved. True or false?");

monkey.setReserved(scanner.nextBoolean());

return;

}

//end if

else {

System.out.println("Monkey is not in program.");

return;

}

//end else

}

//end for

}

//end else if

}

//end reserve animal

// Complete printAnimals

// Include the animal name, status, acquisition country and if the animal is reserved.

// Remember that this method connects to three different menu items.

// The printAnimals() method has three different outputs

// based on the listType parameter

// dog - prints the list of dogs

// monkey - prints the list of monkeys

// available - prints a combined list of all animals that are

// fully trained ("in service") but not reserved

// Remember that you only have to fully implement ONE of these lists.

// The other lists can have a print statement saying "This option needs to be implemented".

// To score "exemplary" you must correctly implement the "available" list.

public static void printAnimals(String input) {

//if dog - list

if(input.equalsIgnoreCase("dog")) {

for (Dog dog: dogList) {

System.out.println(dog.getName());

}

//end for

}

//end if

//if monkey - list

//copy from dog - list

if(input.equalsIgnoreCase("monkey")) {

for (Monkey monkey: monkeyList) {

System.out.println(monkey.getName());

}

//end for

}

//end if

//if available - list

if(input.equalsIgnoreCase("available")) {

for (Dog dog: dogList) {

if (dog.getReserved() == false) {

System.out.println(dog.getName() + " is a " + dog.getBreed());

}

//end if

}

//end for

//do same for monkey

for (Monkey monkey: monkeyList) {

if (monkey.getReserved() == false) {

System.out.println(monkey.getName() + " is a " + monkey.getSpecies());

}

//end if

}

//end for

}

//end if

}

//end print animals

}

//end driver class