

Итоговая контрольная работа

Veranika Lis 6380 group

Информация о проекте

Необходимо организовать систему учета для питомника в котором живут домашние и Pack animals.

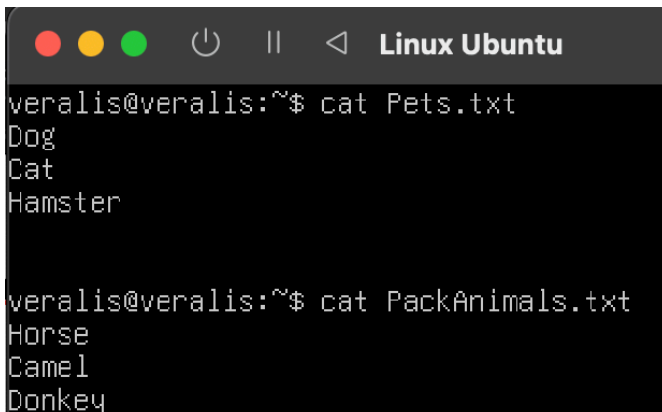
Как сдавать проект

Для сдачи проекта необходимо создать отдельный общедоступный репозиторий(Github, gitlub, или Bitbucket). Разработку вести в этом репозитории, использовать пул реквесты на изменения. Программа должна запускаться и работать, ошибок при выполнении программы быть не должно. Программа, может использоваться в различных системах, поэтому необходимо разработать класс в виде конструктора

Задание

Операционные системы и виртуализация (Linux)

1.Использование команды cat в Linux



```
veralis@veralis:~$ cat Pets.txt
Dog
Cat
Hamster

veralis@veralis:~$ cat PackAnimals.txt
Horse
Camel
Donkey
```

- Создать два текстовых файла: "Pets"(Домашние животные) и "Pack animals"(вьючные животные), используя команду `cat` в терминале Linux. В первом файле перечислить собак, кошек и хомяков. Во втором — лошадей, верблюдов и ослов.

```
veralis@veralis:~$ cat HumanFriends.txt
Dog
Cat
Hamster

Horse
Camel
Donkey
```

- Объединить содержимое этих двух файлов в один и просмотреть его содержимое.

- Переименовать получившийся файл в "Human Friends"(.).

Пример конечного вывода после команды "ls" :

Desktop Documents Downloads HumanFriends.txt Music PackAnimals.txt
Pets.txt Pictures Videos

```
veralis@veralis:~$ ls
app.py  Dockerfile  Human_Friends.txt  my_project  PackAnimals.txt  Pets.txt  snap  task  testfolder
```

2. Работа с директориями в Linux

```
veralis@veralis:~$ cd FriendsDirectory
veralis@veralis:~/FriendsDirectory$ ls
Human_Friends.txt
```

- Создать новую директорию и переместить туда файл "Human Friends".

3. Работа с MySQL в Linux. "Установить MySQL на вашу вычислительную машину"

- Подключить дополнительный репозиторий MySQL и установить один из пакетов из этого репозитория.

4. Управление deb-пакетами

- Установить и затем удалить deb-пакет, используя команду `dpkg`.

5. История команд в терминале Ubuntu

```

Last login: Thu Oct 10 14:12:51 on console
[veranikalis@Veranikas-MacBook-Air ~ % brew services start mysql
Service `mysql` already started, use `brew services restart mysql` to restart.
[veranikalis@Veranikas-MacBook-Air ~ % brew services list
Name Status User      File
mysql started veranikalis ~/Library/LaunchAgents/homebrew.mxcl.mysql.plist
[veranikalis@Veranikas-MacBook-Air ~ % mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.3.0 Homebrew

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

[mysql> history > имя_файла.txt
[      -> echo "Лис Вероника Леонидовна, 10.10.2024, группа 6380" >> имя_файла.txt
[      -> ^C

^C
[mysql> exit;
Bye

```

- Сохранить и выложить историю ваших терминальных команд в Ubuntu.

В формате: Файла с ФИО, датой сдачи, номером группы(или потока)

Объектно-ориентированное программирование

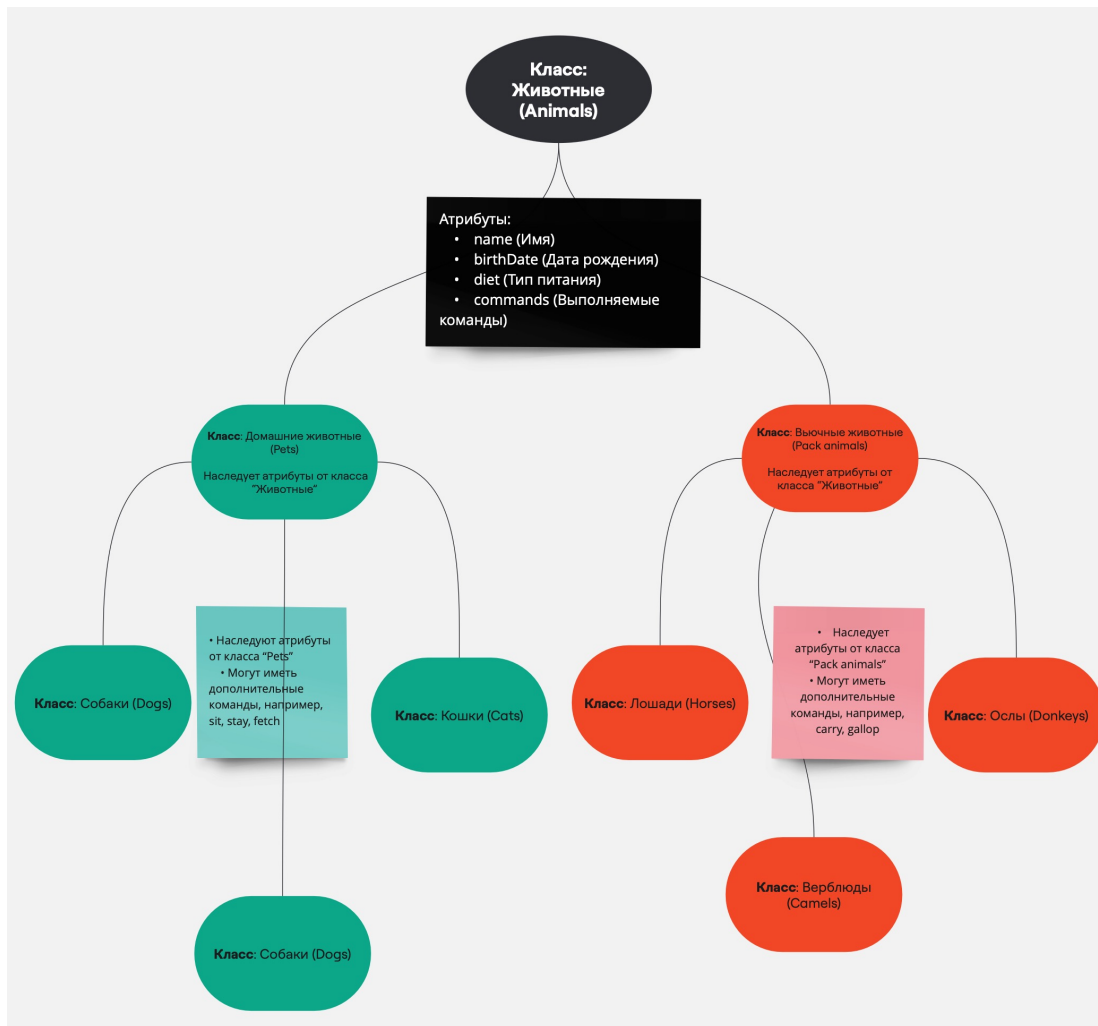
6. Диаграмма классов

- Создать диаграмму классов с родительским классом "Животные", и двумя подклассами: "Pets" и "Pack animals".

В составы классов которых в случае Pets войдут классы: собаки, кошки, хомяки, а в класс Pack animals войдут: Лошади, верблюды и ослы).

Каждый тип животных будет характеризоваться (например, имена, даты рождения, выполняемые команды и т.д)

Диаграмму можно нарисовать в любом редакторе, такими как Lucidchart, Draw.io, Microsoft Visio и других.



7. Работа с MySQL (Задача выполняется в случае успешного выполнения задачи “Работа с MySQL в Linux. “Установить MySQL на вашу машину”

7.1. После создания диаграммы классов в 6 пункте, в 7 пункте база данных "Human Friends" должна быть структурирована в соответствии с этой диаграммой. Например, можно создать таблицы, которые будут соответствовать классам "Pets" и "Pack animals", и в этих таблицах будут поля, которые характеризуют каждый тип животных (например, имена, даты рождения, выполняемые команды и т.д.).

7.2 - В ранее подключенном MySQL создать базу данных с названием "Human Friends".

- Создать таблицы, соответствующие иерархии из вашей диаграммы классов.

- Заполнить таблицы данными о животных, их командах и датами рождения.

```

83 • CREATE TABLE animals_1_to_3_years AS
84   SELECT id, name, TIMESTAMPDIFF(MONTH, birth_date, CURDATE()) AS age_in_months
85   FROM animals
86   WHERE TIMESTAMPDIFF(YEAR, birth_date, CURDATE()) BETWEEN 1 AND 3;
87
88 • CREATE TABLE all_animals AS
89   SELECT a.id, a.name, a.birth_date, a.diet, p.commands, pa.load_capacity,
90          cp.type AS combined_type, cp.characteristic
91   FROM animals a
92   LEFT JOIN pets p ON a.id = p.animal_id
93   LEFT JOIN pack_animals pa ON a.id = pa.animal_id
94   LEFT JOIN combined_pack_animals cp ON pa.id = cp.id;
95
96 • SELECT * FROM all_animals;

```

00% 27:96

Result Grid Filter Rows: Search Export:

	id	name	birth_date	diet	commands	load_capacity	combined_ty...	characteris...
	1	Барсик	2020-05-10	Carnivore	sit, stay	NULL	NULL	NULL
	2	Шарик	2019-07-15	Carnivore	fetch, roll over	NULL	NULL	NULL
	3	Гектор	2018-04-22	Herbivore	NULL	500.00	Donkey	10.50
	3	Гектор	2018-04-22	Herbivore	NULL	500.00	Horse	25.00
	4	Макс	2021-03-12	Herbivore	NULL	450.00	Donkey	10.50

- Удалить записи о верблюдах и объединить таблицы лошадей и ослов.
- Создать новую таблицу для животных в возрасте от 1 до 3 лет и вычислить их возраст с точностью до месяца.
- Объединить все созданные таблицы в одну, сохраняя информацию о принадлежности к исходным таблицам.

Пример заполненной таблицы для теста:

Лист "Pets"

I	D	Name	Type	BirthDate	Commands
1		Fido	Dog	2020-01-01	Sit, Stay, Fetch
2		Whiskers	Cat	2019-05-15	Sit, Pounce
3		Hammy	Hamster	2021-03-10	Roll, Hide
4		Buddy	Dog	2018-12-10	Sit, Paw, Bark
5		Smudge	Cat	2020-02-20	Sit, Pounce, Scratch
6		Peanut	Hamster	2021-08-01	Roll, Spin
7		Bella	Dog	2019-11-11	Sit, Stay, Roll

ID	Name	Type	BirthDate	Commands
8	Oliver	Cat	2020-06-30	Meow, Scratch, Jump

Лист "PackAnimals"

ID	Name	Type	BirthDate	Commands
1	Thunder	Horse	2015-07-21	Trot, Canter, Gallop
2	Sandy	Camel	2016-11-03	Walk, Carry Load
3	Eeyore	Donkey	2017-09-18	Walk, Carry Load, Bray
4	Storm	Horse	2014-05-05	Trot, Canter
5	Dune	Camel	2018-12-12	Walk, Sit
6	Burro	Donkey	2019-01-23	Walk, Bray, Kick
7	Blaze	Horse	2016-02-29	Trot, Jump, Gallop
8	Sahara	Camel	2015-08-14	Walk, Run

8. ООП и Java

- Создать иерархию классов в Java, который будет повторять диаграмму классов созданную в задаче 6(Диаграмма классов) .

9. Программа-реестр домашних животных

- Написать программу на Java, которая будет имитировать реестр домашних животных.

Должен быть реализован следующий функционал:

9.1. Добавление нового животного

- Реализовать функциональность для добавления новых животных в реестр.

Животное должно определяться в правильный класс (например, "собака", "кошка", "хомяк" и т.д.)

9.2. Список команд животного

- Вывести список команд, которые может выполнять добавленное животное (например, "сидеть", "лежать").

9.3. Обучение новым командам

- Добавить возможность обучать животных новым командам.

9.4 Вывести список животных по дате рождения

9.5. Навигация по меню

- Реализовать консольный пользовательский интерфейс с меню для навигации между вышеуказанными функциями.

10. Счетчик животных

Создать механизм, который позволяет вывести на экран общее количество созданных животных любого типа (Как домашних, так и вьючных), то есть при создании каждого нового животного счетчик увеличивается на "1".

AnimalShelterSystem

ООП и Java

1. View (MenuView) отвечает за отображение меню, получение ввода от пользователя и вывод сообщений.

2. Controller (AnimalShelterController) отвечает за логику работы: вызывает методы сервиса и взаимодействует с пользователем через View.

3. Service (AnimalService) содержит логику добавления животных, работу с базой данных или счётчиком.

4. Model (Animal, Dog, Cat, etc.) содержит классы животных и их характеристики.

/src

 /controller

 AnimalShelterController.java

 /model

 Animal.java

 PackAnimal

 Pet

 /packAnimals

 Horse.java

 /pets

 Dog.java

 Cat.java

 /service

 AnimalService.java

 Counter.java

 /view

 MenuView.java

Main.java