МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

|  |
| --- |
| КАФЕДРА АЭРОКОСМИЧЕСКИХ КОМПЬЮТЕРНЫХ И ПРОГРАММНЫХ СИСТЕМ |

КУРСОВАЯ РАБОТА

ЗАЩИЩЕНА С ОЦЕНКОЙ

РУКОВОДИТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доц., канд.т.н, доц. |  |  |  | К. А. Курицын |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К КУРСОВОЙ РАБОТЕ  ООП. ПАТТЕРЫ ПРОЕКТИРОВАНИЯ |
|  |
| по дисциплине: ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 1642 |  |  |  | Д. А. Иванов |
|  | номер группы |  | подпись, дата |  | инициалы, фамилия |
| Студенческий билет № | 2017/3375 | |  |  |  |

Санкт-Петербург 2019

Содержание

**1. Задание3**

**2. Функциональные требования3**

**3. Требования к оборудованию и стороннему ПО 3**

**4. Технические спецификации 3**

**5. Взаимодействие с внешними и внутренними системами 3**

**6. Программа и методика испытаний 4**

**7. Литература 4**

**Приложение 1. Описание формата данных для ввода.5**

**Приложение 2. UML-диаграмма классов.5**

**Приложение 3. Результат работы программы.5**

**Приложение 4. Листинг программы.7**

1. **Задание.**

Создать класс «Животное», в результате использования которого клиент может получить: кошка, собака, мышь, лошадь, змея, рыбка. Клиент имеет возможность выбора типа создания животного и некоторых его характеристик (порода, окрас, кличка, пол, дата рождения, размеры, имя владельца, наименование питомника, характер). С консоли или из соответствующего файла. Использовать паттерн «Фабричный метод». Результат создания животных и все их характеристики записываются в выходные файлы по типам животных: «cat», «dog», «mouse», «horse», «snake», «fish».

1. **Функциональные требования.**
2. Считывание данных, введённых с клавиатуры. Как с консоли, так и с графического интерфейса. Требования по формату данных определены в прил. 1.
3. Выбор файла для записи считанных данных.
4. Запись данных в соответствующий файл .txt.
5. Использовать паттерн проектирования «Фабричный метод».
6. Программа должна быть реализована на языке программирования Java.
7. **Требования к оборудованию и стороннему ПО**

* Консольное приложение/Графический интерфейс: ОС Lubuntu 19.04. IntelliJ IDEA 2019.1. JRE 8u211.

1. **Технические спецификации.**
2. Графический интерфейс программы вызывается с помощью .jar файла animalForm.jar.
3. Отображается окно формы для ввода данных.
4. Пользователю предлагается выбрать тип животного с помощью переключателя (radio button).
5. Пользователь вводит необходимые данные в соответствии с требованиями прил. 1.
6. По нажатию кнопки Apply программа выведет результат ввода данных: «Информация успешно записана файл», или «Заполните все поля», если не все поля были заполнены, или «Неверный формат поля “Рост”», если данные введены не в соответствии требованиям из прил. 1.
7. Введённые данные сохраняются в соответствующий файл.
8. **Взаимодействие с внешними и внутренними системами.**

* Программа взаимодействует с файловой системой.
* Взаимодействие с потоками ввода/вывода Java. (System.in/System.out). Вывод данных на консоль с помощью функции println() из пакета System.out, ввод с консоли – с помощью вспомогательного класса Scanner, объект которого создаётся передачей потока System.in в качестве параметра.

1. **Программа и методика испытаний.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Сценарий проверки** | **Ожидаемый результат** | **ФТ.** | **Техн. Спец.** | **Результат** |
| 1 | Запуск приложения. | Отображение полей ввода, выбора животного и кнопки «применить» | 2.1. | 4.1. | Рисунок 3.1. |
| 2 | Выбор животного. | Выбрано необходимое животное с помощью переключателя | 2.2. | 4.3. | Рисунок 3.2. |
| 3 | Заполнение пользователем всех полей с данными. | Все поля формы заполнены в соответствии с требованиями | 2.1. | 4.4. | Рисунок 3.2. |
| 4 | Запись данных в файл. | Данные о выбранном животном успешно записаны в файл | 2.3. | 4.5. | Рисунок 3.2. |
| 5 | Ввод некорректных данных. | Сообщение об ошибке при вводе некорректных данных. | 2.1. | 4.5. | Рисунок 3.3. Рисунок 3.4. |

1. **Литература.**

7.1. Шилдт, Г. Java 8. Полное руководство, 9-е изд.: Пер. с англ. - М.: ООО «И.Д. Вильямс», 2015. – 1376 с.

7.2. Шилдт, Г. Java: методики программирования Шилдта.: Пер. с англ. - М.: ООО «И.Д. Вильямс», 2008. – 512 с.

**1. Приложение – 1. Описание формата данных для ввода.**

Порода: String.

Окрас: String.

Кличка: String.

Пол: String.

Дата рождения: String (dd/mm/yyyy - автоматически)

Размеры: целое число.

Имя владельца: String.

Наименование питомника: String.

Характер: String.

**2. Приложение – 2. UML-диаграмма классов.**

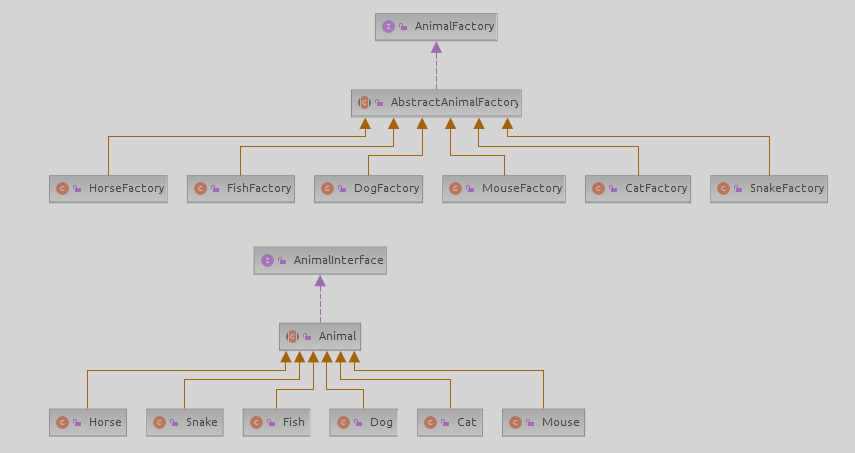


Рис 2.1. - UML-диаграмма наследования классов.

**3. Приложение – 3. Результат работы программы.**

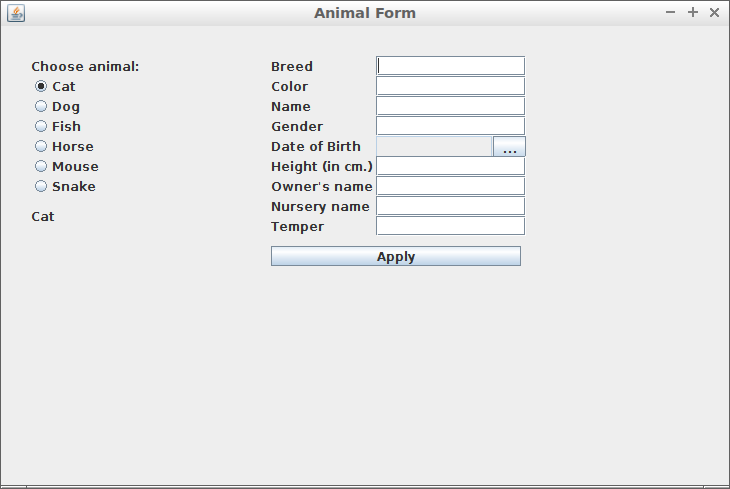
****

Рис. 3.1. – форма для ввода данных пользователем.

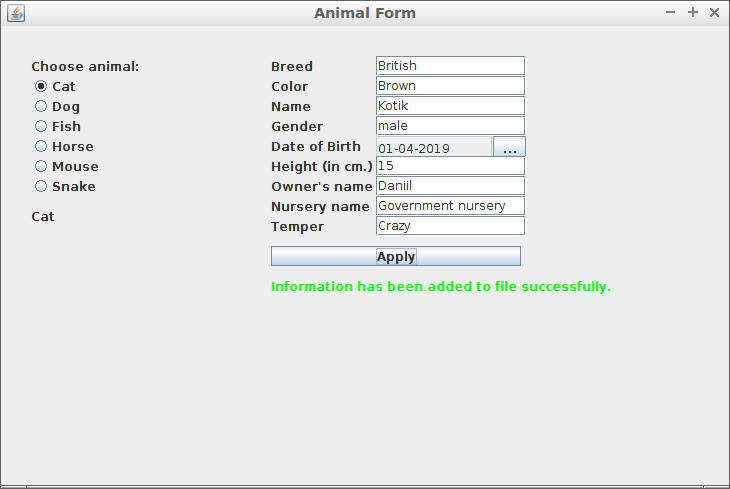


Рис. 3.2. – данные введены корректно.

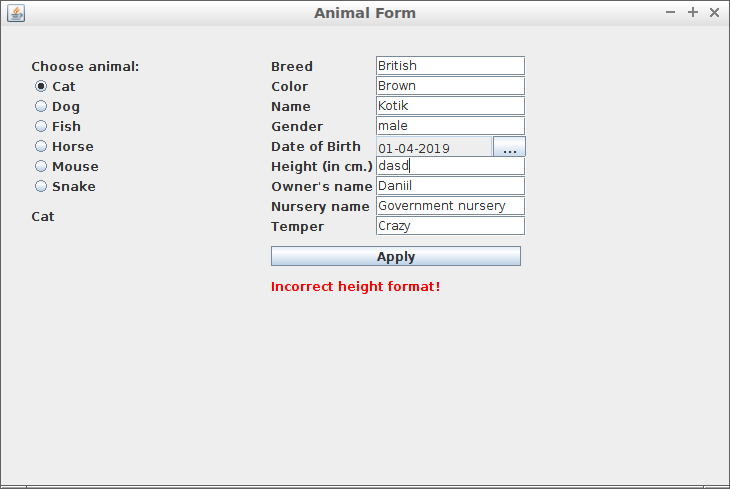


Рис. 3.3. – пользователем введены некорректные данные (Строка вместо целого числа).

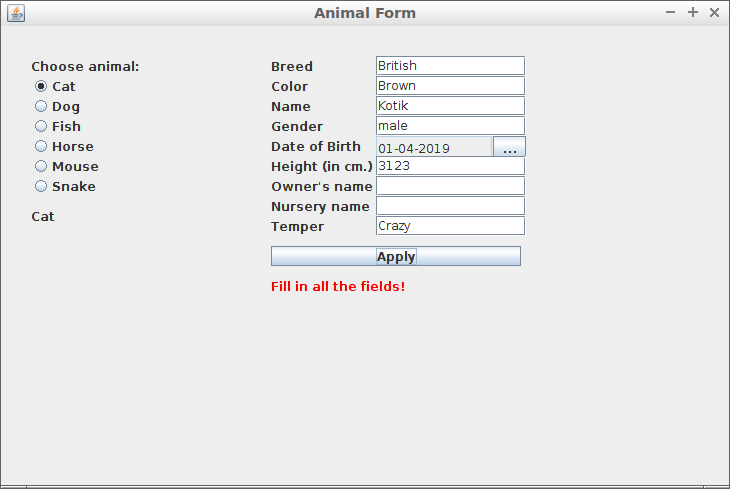


Рис. 3.4. – пользователь ввёл не все данные.

Файл cat.txt:

id: 2117131325

Breed: British

Color: Brown

Name: Kotik

Gender: male

Date of Birth: 01-04-2019

Height: 15 (cm.)

Owner's name: Daniil

Nursery name: Government nursery

Temper: Crazy

Программа работает верно, что подтверждено снимками экрана.

**4. Приложение – 4. Листинг программы.**

Листинг программы расположен по адресу:

<https://github.com/daniiliv/CourseworkTP>