

Министерство образования и молодежной политики Свердловской
области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»



Отчёт по программе «Практическое занятие 8.1»

Выполнил: Сагутдинова Илона

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

Задание:

Вариант 7. Класс «Собака»: кличка, порода, рост по холке, окрас шерсти, вес, характер.

Входные данные:

N - выбор действия, число, int.

Name - имя собаки, строка, string.

Breed - порода собаки, строка, string.

Color - окрас собаки, строка, string.

Weight - вес собаки, дробное, double.

Height - рост по холке собаки, дробное, double.

Haract - характер собаки, строка, string.

Выходные данные:

В зависимости от выбранного действия, добавление данных собаки, их обновление, показ информации или удаление.

Блок-схема:

Листинг программы:

```
fun main() {
    try {
        var dog: Dog? = null
        while (true) {
            println("Выберите действие:")
            println("1. Добавить собаку")
            println("2. Показать информацию о собаке")
            println("3. Обновить информацию о собаке")
            println("4. Удалить собаку")
            println("5. Выйти")
            val n = readLine()!!.toInt()
            when (n) {
                1 -> {
                    dog = addDog()
                    println("Собака добавлена!")
                }

                2 -> {
                    if (dog != null) {
                        println("Информация о собаке:")
                        dog.displayInfo()
                    } else {
                        println("Собака не добавлена.")
                    }
                }

                3 -> {
                    if (dog != null) {
                        println("Введите новую кличку собаки:")
                    }
                }
            }
        }
    }
}
```

```

3 -> {
    if (dog != null) {
        println("Введите новую кличку собаки:")
        val newName = readLine()!!.toString()
        println("Введите новую породу собаки:")
        val newBreed = readLine()!!.toString()
        println("Введите новый окрас собаки:")
        val newColor = readLine()!!.toString()
        println("Введите новый вес собаки (кг):")
        val newWeight = readLine()!!.toDouble()
        println("Введите новый рост собаки (см):")
        val newHeight = readLine()!!.toDouble()
        println("Выберите характер собаки:")
        println("1. Дружелюбный")
        println("2. Агрессивный")
        println("3. Спокойный")
        println("4. Активный")
        println("5. Умная")
        val newHaract = ""
        dog.updateInfo(newName, newBreed, newColor, newWeight, newHeight, newHaract)
    } else {
        println("Собака не добавлена.")
    }
}

4 -> {
    if (dog != null) {
        dog = null
        println("Собака удалена!")
    }
}

```

```

        else
        {
            println("Чтобы удалить собаку, ее надо добавить.")
        }
    }

    5 -> {
        println("Выход из программы.")
        return
    }

    else -> println("Неверный ввод. Попробуйте снова.")
}

}

catch (e:Exception)
{
    println("Неверный ввод")
}
}

```

```

class Dog (
    var name: String,
    var breed: String,
    var color: String,
    var weight: Double,
    var height: Double,
    var haract: String,
){
    fun displayInfo() {
        println("Кличка: $name")
        println("Порода: $breed")
        println("Окрас: $color")
        println("Вес: $weight кг")
        println("Рост: $height см")
        println("Характер: $haract")
    }

    fun updateInfo(newName: String, newBreed: String, newColor: String, newWeight: Double, newHeight: Double, newHaract: String) {
        name = newName
        breed = newBreed
        color = newColor
        weight = newWeight
        height = newHeight
        val n = readLine()!!.toInt()
        haract = when (n){
            1 -> "Дружелюбный"
            2 -> "Агрессивный"
            3 -> "Спокойный"
            4 -> "Активный"
            5-> "Умная"
            else -> "Неверный ввод"
        }
    }
}

```

```

        println("Информация о собаке обновлена!")
    }
}

fun addDog(): Dog {
    println("Введите кличку собаки:")
    val name = readLine()!!.toString()

```

```

println("Введите породу собаки:")
val breed = readLine()!!.toString()
println("Введите окрас собаки:")
val color = readLine()!!.toString()
var weight: Double
while (true) {
    println("Введите вес собаки (кг):")
    val weightInput = readLine()!!.toDouble()
    if (weightInput != null && weightInput > 0 && weightInput < 170 && weightInput >1) {
        weight = weightInput
        break
    } else {
        println("Вес должен быть положительным, не равняться нулю и иметь реальные размеры")
    }
}
var height: Double
while (true) {
    println("Введите рост собаки (см):")
    val heightInput = readLine()!!.toDouble()
    if (heightInput != null && heightInput > 0 && heightInput <=110 && heightInput >=10) {
        height = heightInput
        break
    } else {
        println("Рост должен быть положительным, не равняться нулю и иметь реальные размеры")
    }
}
println("Выберите характер собаки:")

```

```

println("1. Дружелюбный")
println("2. Агрессивный")
println("3. Спокойный")
println("4. Активный")
println("5. Умная")
val n = readLine()!!.toInt()
val harac = when (n){
    1 -> "Дружелюбный"
    2 -> "Агрессивный"
    3 -> "Спокойный"
    4 -> "Активный"
    5-> "Умная"
    else -> "Неверный ввод"
}
var haract: String
haract = harac
return Dog(name, breed, color, weight, height, haract)
}

```

Тестовые ситуации:

Выберите действие: 1. Добавить собаку 2. Показать информацию о собаке 3. Обновить информацию о собаке 4. Удалить собаку 5. Выйти 1 Введите кличку собаки: петя Введите породу собаки: бульдог Введите окрас собаки: черный Введите вес собаки (кг): 24.1 Введите рост собаки (см): 23 Выберите характер собаки: 1. Дружелюбный 2. Агрессивный 3. Спокойный 4. Активный 5. Умная 4 Собака добавлена! Выберите действие: 1. Добавить собаку 2. Показать информацию о собаке 3. Обновить информацию о собаке	4. Удалить собаку 5. Выйти 2 Информация о собаке: Кличка: петя Порода: бульдог Окрас: черный Вес: 24.1 кг Рост: 23.0 см Характер: Активный Выберите действие: 1. Добавить собаку 2. Показать информацию о собаке 3. Обновить информацию о собаке 4. Удалить собаку 5. Выйти 3 Введите новую кличку собаки: люся Введите новую породу собаки: бульдог Введите новый окрас собаки: белая Введите новый вес собаки (кг): 36 Введите новый рост собаки (см): 34 Выберите характер собаки: 1. Дружелюбный	2. Агрессивный 3. Спокойный 4. Активный 5. Умная 1 Информация о собаке обновлена! Выберите действие: 1. Добавить собаку 2. Показать информацию о собаке 3. Обновить информацию о собаке 4. Удалить собаку 5. Выйти 2 Информация о собаке: Кличка: люся Порода: бульдог Окрас: белая Вес: 36.0 кг Рост: 34.0 см Характер: Дружелюбный Выберите действие: 1. Добавить собаку 2. Показать информацию о собаке 3. Обновить информацию о собаке 4. Удалить собаку 5. Выйти 4 Собака удалена! Выберите действие:
---	--	--

```
Выберите действие:
1. Добавить собаку
2. Показать информацию о собаке
3. Обновить информацию о собаке
4. Удалить собаку
5. Выйти
2
Собака не добавлена.
Выберите действие:
1. Добавить собаку
2. Показать информацию о собаке
3. Обновить информацию о собаке
4. Удалить собаку
5. Выйти
5
Выход из программы.

Process finished with exit code 0
```

Выберите действие:

1. Добавить собаку
2. Показать информацию о собаке
3. Обновить информацию о собаке
4. Удалить собаку
5. Выйти

1

Введите кличку собаки:

петя

Введите породу собаки:

бульдог

Введите окрас собаки:

черный

Введите вес собаки (кг):

-1

Вес должен быть положительным, не равняться нулю и иметь реальные размеры

Введите вес собаки (кг):

1

Вес должен быть положительным, не равняться нулю и иметь реальные размеры

Введите вес собаки (кг):

24

Введите рост собаки (см):

-1

Рост должен быть положительным, не равняться нулю и иметь реальные размеры

Введите рост собаки (см):

45

Выберите характер собаки:

1. Дружелюбный

2. Агрессивный

3. Спокойный

4. Активный

5. Умная

4

Собака добавлена!

Выберите действие:

1. Добавить собаку

2. Показать информацию о собаке

3. Обновить информацию о собаке

4. Удалить собаку

5. Выйти

4

Собака удалена!

Выберите действие:

1. Добавить собаку

2. Показать информацию о собаке

3. Обновить информацию о собаке

4. Удалить собаку

5. Выйти

3

Собака не добавлена.

Выберите действие:

1. Добавить собаку

2. Показать информацию о собаке

3. Обновить информацию о собаке

4. Удалить собаку

5. Выйти

Вывод: Я научилась работать с классами в kotlin.