

**Федеральное государственное автономное образовательное  
учреждение**

**высшего образования**

**«Национальный исследовательский университет ИТМО»**

**Факультет Систем Управления и Робототехники**



**Вариант №42459**

**Лабораторная работа №3-4**

**по дисциплине**

**Программирование**

**Выполнил Студент группы R3142**

**Васильев И. Д.**

**Преподаватель:**

**Кобелев Р. П.**

## 1. Текст задания

### Лабораторная работа #3-4

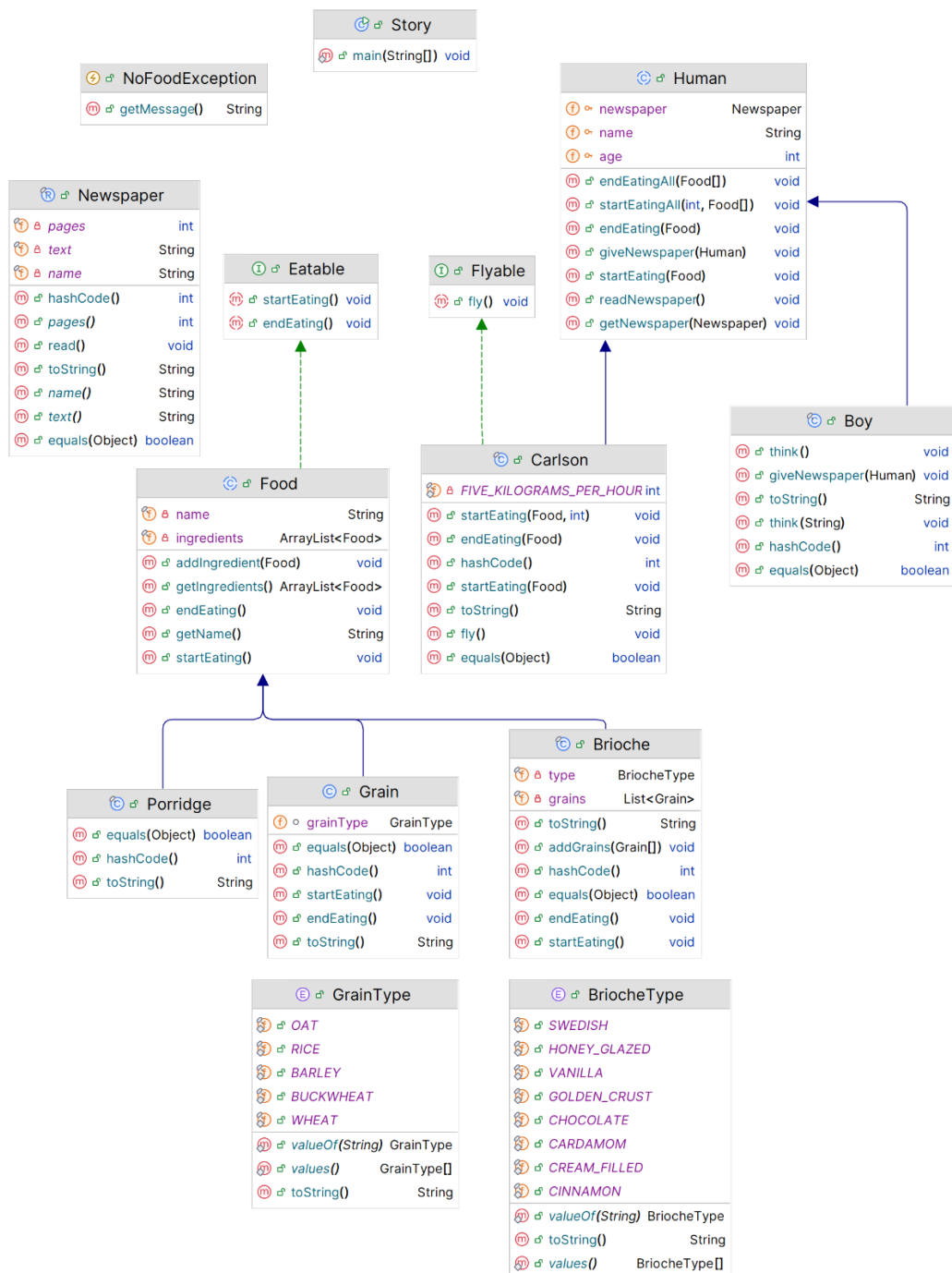
В соответствии с выданным вариантом на основе предложенного текстового отрывка из литературного произведения создать объектную модель реального или воображаемого мира, описываемого данным текстом. Должны быть выделены основные персонажи и предметы со свойственным им состоянием и поведением. На основе модели написать программу на языке Java.

Введите вариант:

**Описание предметной области, по которой должна быть построена объектная модель:**

Только после того как вся каша до последней крупинки оказалась у Карлсона в тарелке, он приступил к еде, да так, что за ушами трещало. На несколько минут все звуки в кухне заглушило громкое чавканье, которое всегда раздается, когда кто-нибудь очень жадно уплетает кашу. Пока Карлсон заглатывал одну за другой булочки, Малыш тихо сидел и обдумывал, как ему поумнее предостеречь своего друга. "Может, правильнее всего просто дать Карлсону газету? Пусть сам все прочтет", -- решил он и после некоторого колебания придвинул газету к Карлсону.

## 2. UML-диаграмма классов



### 3. Исходный код

<https://github.com/Garik-prog/itmo-labs-2025.git>

### 4. Результат работы программы

```
C:\Users\User\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2025.2.4\lib\idea_rt.jar=2399" -D
Carlson started eating...
    Grain wheat is cracking!
    Grain barley is cracking!
    Grain oat is cracking!
It is cracking behind ears!
    Grain wheat was eating
    Grain barley was eating
    Grain oat was eating
Вкуснейшая каша consists of [Grain { grainType = wheat } , Grain { grainType = barley } , Grain { grainType = oat } ] was eating full!
Carlson is chewing loudly... As always...
Carlson started eating...
Start eating Brioche Cardamom
    Grain wheat is cracking!
Carlson started eating...
Brioche Chocolate without grains!
Little Boy sits quietly and thinking about his friend...
Carlson ended eating...
    Grain wheat was eating
Brioche Cardamom with grains was eating full!
Carlson ended eating...
Brioche Chocolate without grains was eaten!
Brioche Chocolate without grains was eating full!
Little Boy thought: "Maybe it would be better to just give Carlson the newspaper?
- The boy hesitated a bit, but decided to let him read it himself.
Carlson ended eating...
    Grain wheat was eating
Brioche Cardamom with grains was eating full!
Carlson ended eating...
Brioche Chocolate without grains was eaten!
Brioche Chocolate without grains was eating full!
Little Boy pushed the newspaper towards Carlson

Carlson has started reading newspaper...
Reading newspaper 'Washington Post'
    Reading page[1]
    Reading page[2]
    Reading page[3]
    Reading page[4]
    Reading page[5]
    Reading page[6]
    Reading page[7]
    Reading page[8]
    Reading page[9]
    Reading page[10]
Text { Lorem ipsum dolor sit amet, co... } has been read!
Carlson has been reading newspaper in 2 seconds!

Process finished with exit code 0
```

## 5. Анализ реализаций метода

Обратившись к DeepSeek:

у меня есть record Newspaper(int pages, String name, String text). Мне необходимо реализовать метод read() так, чтобы в терминал выводился название газеты, номер прочитанной страницы, причём после каждой страницы должна быть пауза в 200 миллисекундах



Вот реализация метода `read()` для вашего record `Newspaper` :

java

Copy

Download

```
import java.util.concurrent.TimeUnit;

public record Newspaper(int pages, String name, String text) {

    public void read() {
        System.out.println("Чтение газеты: \"" + name + "\"");
        System.out.println("Всего страниц: " + pages);
        System.out.println("=====");

        for (int i = 1; i <= pages; i++) {
            System.out.println("Газета: " + name + " | Страница: " + i + "/" + pages);

            // Пауза 200 миллисекунд
            try {
                TimeUnit.MILLISECONDS.sleep(200);
            } catch (InterruptedException e) {
                Thread.currentThread().interrupt();
                System.out.println("Чтение прервано!");
                return;
            }
        }

        System.out.println("=====");
        System.out.println("Газета \"" + name + "\" прочитана!");
    }
}
```

## Обратившись к GigaChat:

у меня есть record Newspaper(int pages, String name, String text). Мне необходимо реализовать метод read() так, чтобы в терминал выводился название газеты, номер прочитанной страницы, причём после каждой страницы должна быть пауза в 200 миллисекундах  
язык программирования -- джава

Ответ 

Вот реализация метода `read()` для вашего класса `Newspaper` на Java:

```
java Копировать

import java.util.concurrent.TimeUnit;

public class Main {
    public static void main(String[] args) throws InterruptedException {
        // Создаем экземпляр газеты
        var newspaper = new Newspaper(
            3,                // Количество страниц
            "Известия",       // Название газеты
            """
                Статья №1\nСтраница 1\n
                Статья №2\nСтраница 2\n
                Статья №3\nСтраница 3""");

        // Читаем газету
        newspaper.read();
    }
}

record Newspaper(int pages, String name, String text) {
    public void read() throws InterruptedException {
        String[] pagesArray = this.text().split("\\n\\n"); // Разбиваем текст на отдельные страницы

        for (int pageNumber = 0; pageNumber < pages(); pageNumber++) {
            System.out.println(name() + ", страница " + (pageNumber + 1));
            System.out.println(pagesArray[pageNumber]);

            TimeUnit.MILLISECONDS.sleep(200); // Пауза в 200 миллисекунд
        }
    }
}
```

## Моя реализация:

```
public void read() {
    System.out.println("Reading newspaper '" + name + "'");

    try {
        for (int i = 1; i <= pages; i++) {
            System.out.println("\tReading page[" + i + "]");
            Thread.sleep(200);
        }
    }
```

```

    } catch (InterruptedException e) {
        e.printStackTrace();
    }

    System.out.println("Text { " + text.substring(0, Math.min(30, text.length())).concat("...") + " }
has been read!");
}

```

#### **Анализ:**

Моя реализация, очевидно, отличается выводом информации, а также, в сравнении с предложенным вариантом от **GigaChat**, у меня и **DeepSeek’a** лучше выбрано начало отсчёта в цикле, отвечающем за перебор всех страниц, поскольку нумеровать страницы с 1 намного логичнее, чем с 0, но на логику функции это никак не влияет. Также методы нейросетей используют задержку из библиотеки TimeUnit, что позволяет проще работать с разными единицами измерения времени, но в конкретном случае никакой разницы нет.

### **6. Выводы по работе**

За время работы над проектом я:

- Научился переводить текстовое описание в объектную модель
- Научился правильно строить иерархии классов и применять принципы SOLID в реальных проектах
- Освоил record, enum
- Убедился, что ИИ помогает, но его код нужно проверять и улучшать
- Научился собирать проекты с помощью Maven