Факультет Программной Инженерии и Компьютерной Техники

Вариант 13310

Выполнил:

Бармичев Григорий Андреевич

Группа P3110

Проверил:

Мустафаева Айнур Вугар Кызы

Преподаватель практик

**Содержание**

[Задание 3](#_Toc184498687)

[Исходный код программы 4](#_Toc184498688)

[Диаграмма классов 4](#_Toc184498689)

[Результат программы 4](#_Toc184498690)

[Вывод 4](#_Toc184498691)

## Задание

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.

|  |
| --- |
| **Потом Карлсон стал прощаться: Час спустя Малыш уже лежал в своей кроватке, а рядом стояла корзинка, где спал Бимбо. Все -- и мама, и папа, и Боссе, и Бетан -- пришли к нему в комнату пожелать спокойной ночи. Малыша уже одолевал сон. Но он еще не спал, а думал о Карлсоне. Что он сейчас делает, Карлсон? Может, как раз что-нибудь мастерит... скворечник или еще что...** |

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

**Этапы выполнения работы:**

1. Получить вариант
2. Нарисовать UML-диаграмму, представляющую классы и интерфейсы объектной модели и их взаимосвязи;
3. Придумать сценарий, содержащий действия персонажей, аналогичные приведенным в исходном тексте;
4. Согласовать диаграмму классов и сценарий с преподавателем;
5. Написать программу на языке Java, реализующую разработанные объектную модель и сценарий взаимодействия и изменения состояния объектов. При запуске программа должна проигрывать сценарий и выводить в стандартный вывод текст, отражающий изменение состояния объектов, приблизительно напоминающий исходный текст полученного отрывка.
6. Продемонстрировать выполнение программы на сервере helios.
7. Ответить на контрольные вопросы и выполнить дополнительное задание.

Текст, выводящийся в результате выполнения программы не обязан дословно повторять текст, полученный в исходном задании. Также не обязательно реализовывать грамматическое согласование форм и падежей слов выводимого текста.

Стоит отметить, что цель разработки объектной модели состоит не в выводе текста, а в эмуляции объектов предметной области, а именно их состояния (поля) и поведения (методы). Методы в разработанных классах должны изменять состояние объектов, а выводимый текст должен являться побочным эффектом, отражающим эти изменения.

**Требования к объектной модели, сценарию и программе:**

1. В модели должны быть представлены основные персонажи и предметы, описанные в исходном тексте. Они должны иметь необходимые атрибуты и характеристики (состояние) и уметь выполнять свойственные им действия (поведение), а также должны образовывать корректную иерархию наследования классов.
2. Объектная модель должна реализовывать основные принципе ООП - инкапсуляцию, наследование и полиморфизм. Модель должна соответствовать принципам SOLID, быть расширяемой без глобального изменения структуры модели.
3. Сценарий должен быть вариативным, то есть при изменении начальных характеристик персонажей, предметов или окружающей среды, их действия могут изменяться и отклоняться от базового сценария, приведенного в исходном тексте. Кроме того, сценарий должен поддерживать элементы случайности (при генерации персонажей, при задании исходного состояния, при выполнении методов).
4. Объектная модель должна содержать как минимум один корректно использованный элемент каждого типа из списка:
   * абстрактный класс как минимум с одним абстрактным методом;
   * интерфейс;
   * перечисление (enum);
   * запись (record);
   * массив или ArrayList для хранения однотипных объектов;
   * проверяемое исключение.
5. В созданных классах основных персонажей и предметов должны быть корректно переопределены методы equals(), hashCode() и toString(). Для классов-исключений необходимо переопределить метод getMessage().
6. Созданные в программе классы-исключения должны быть использованы и обработаны. Кроме того, должно быть использовано и обработано хотя бы одно unchecked исключение (можно свое, можно из стандартной библиотеки).
7. При необходимости можно добавить внутренние, локальные и анонимные классы.

**Содержание отчёта по работе:**

1. Текст задания.
2. Диаграмма классов объектной модели.
3. Исходный код программы (можно в виде ссылки на репозиторий).
4. Результат работы программы.
5. Выводы по работе.

## Исходный код программы

Ссылка на репозиторий:

## Диаграмма классов

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

## Результат программы

Потом Карлсон прощаться.

Час спустя Малыш лежал в кроватке.

Корзинка стоять рядом.

Все --[мама, папа, Боссе, Бетан]-- пришел в комната пожелать спокойной ночи.

Малыш одолевал сон. Но Малыш еще не спал.

Малыш думал о Карлсон.

Что Карлсон Сейчас делает? Может, Карлсон что-нибудь мастерит... скворечник или еще что...

## Вывод

Во время выполнения лабораторной работы я научился декомпозировать программу на классы и интерфейсы. Улучшил свои навыки работы с полями, методами и классами и UML диаграммами.

Потом Карлсон стал прощаться

Час спустя Малыш лежал в кроватке

Рядом стояла корзинка

Все -- и мама, и папа, и Боссе, и Бетан -- пришли в комнату пожелать спокойной ночи.

Малыша одолевал сон. Но Малыш ещё не спал.

Малыш думал о Карлсоне.

Что Карлсон делает? Карлсон мастерит что-нибудь… скворечник или еще что...