

Санкт-Петербургский Национальный Исследовательский Университет
ИТМО

Факультет Программной Инженерии и Компьютерной Техники



УНИВЕРСИТЕТ ИТМО

Лабораторная работа № 3

По дисциплине

Программирование

Вариант №2711

Выполнил студент группы Р3108:

Петров Вячеслав Маркович

Преподаватель:

Письмак Алексей Евгеньевич

Санкт-Петербург 2023 г.

Текст задания

Программа должна удовлетворять следующим требованиям:

1. Доработанная модель должна соответствовать принципам SOLID.
2. Программа должна содержать как минимум два интерфейса и один абстрактный класс (номенклатура должна быть согласована с преподавателем).
3. В разработанных классах должны быть переопределены методы `equals()`, `toString()` и `hashCode()`.
4. Программа должна содержать как минимум один перечисляемый тип (`enum`).

Порядок выполнения работы:

1. Доработать объектную модель приложения.
2. Перерисовать диаграмму классов в соответствии с внесёнными в модель изменениями.
3. Согласовать с преподавателем изменения, внесённые в модель.
4. Модифицировать программу в соответствии с внесёнными в модель изменениями.

Отчёт по работе должен содержать:

1. Текст задания.
2. Диаграмма классов объектной модели.
3. Исходный код программы.
4. Результат работы программы.
5. Выводы по работе.

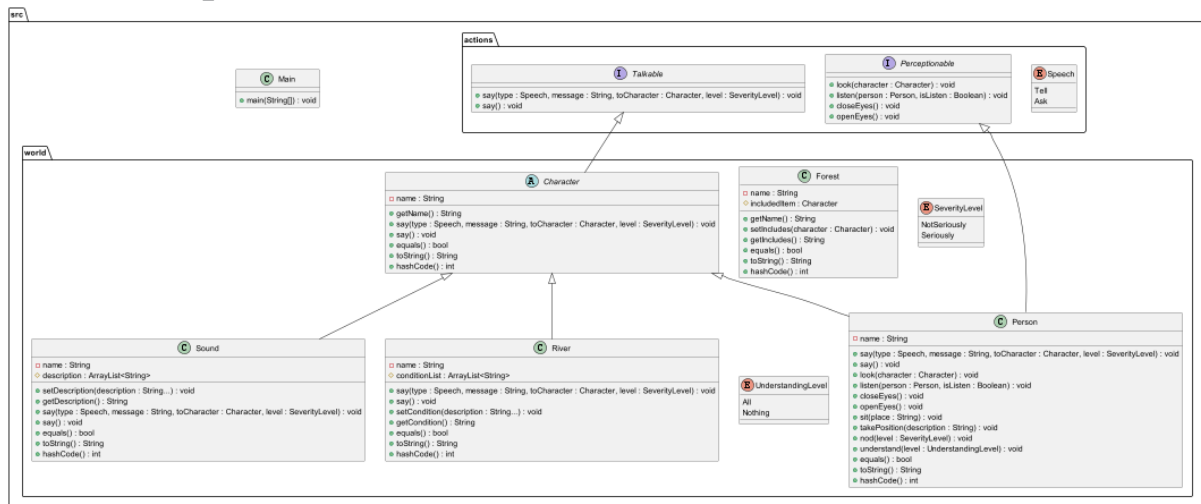
Вопросы к защите лабораторной работы:

1. Принципы объектно-ориентированного программирования SOLID и STUPID.
2. Класс `Object`. Реализация его методов по умолчанию.
3. Особенности реализации наследования в Java. Простое и множественное наследование.
4. Понятие абстрактного класса. Модификатор `abstract`.
5. Понятие интерфейса. Реализация интерфейсов в Java, методы по умолчанию. Отличия от абстрактных классов.
6. Перечисляемый тип данных (`enum`) в Java. Особенности реализации и использования.
7. Методы и поля с модификаторами `static` и `final`.
8. Перегрузка и переопределение методов. Коварианты возвращаемых типов данных.
9. Элементы функционального программирования в синтаксисе Java. Функциональные интерфейсы, лямбда-выражения. Ссылки на методы.

Текст задания:

Долгое время они глядели вниз на реку, ничего не говоря, и Река тоже ничего не говорила, потому что ей было очень спокойно и хорошо в этот солнечный полдень. Кролик и Пятачок сидели возле парадной двери дома Винни-Пуха и слушали, что говорит Кролик. Винни-Пух тоже сидел с ними. Был дремотный летний полдень, и Лес был полон тихих, неясных звуков, и все они, казалось, говорили Пуху: "Не слушай Кролика. Слушай меня". Поэтому Винни занял самую удобную позицию для того, чтобы не слушать Кролика, и лишь время от времени открывал глаза и говорил: "Ах", а потом закрывал глаза снова и говорил: "Верно, верно!" Сам же Кролик то и дело очень серьезно спрашивал: "Ты понимаешь, что я имею в виду, Пятачок?", а Пятачок не менее серьезно кивал в ответ, чтобы показать, что он все-все понимает.

UML диаграмма



<https://github.com/petrovviacheslav/myitmo/blob/main/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5/Viacheslav/lab3/diagram.png>

Исходный код программы

<https://github.com/petrovviacheslav/myitmo/tree/main/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5/Viacheslav/lab3/src>

Результат работы программы

Кролик смотрел на река Река
Пятачок смотрел на река Река
Кролик ничего не говорил
Пятачок ничего не говорил
Река ничего не говорила
Река было хорошо, спокойно

Кролик сидел рядом с входной дверью
Пятачок сидел рядом с входной дверью
Винни-Пух сидел с ними

Лес включает в себя звук звуки
звуки говорили персонаж Винни-Пуху "Не слушай Кролика. Слушай меня."

Винни-Пух занял удобную позицию
Винни-Пух не слушал персонаж Кролик
Винни-Пух открывал глаза
Винни-Пух говорил персонаж Кролику "Ах!"
Винни-Пух закрывал глаза
Винни-Пух говорил персонаж Кролику "Верно, верно!"

Кролик серьёзно спросил у персонаж Пятачок "Ты понимаешь, что я имею в виду, Пятачок?"
Пятачок серьёзно кивал в ответ
Пятачок показывает, что он все-все понимает

Выводы по работе

При выполнении лабораторной работы я изучил принципы SOLID, научился использовать интерфейсы, еипты, абстрактные классы и выстраивать взаимодействие между ними. Кроме того, я более детально изучил ООП: использовал переопределение и перегрузку методов.