

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 13:09:47
Уникальный программный ключ:
65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ
Проректор по учебной работе
О.Г. Доктинова
« 24 » 12 2017 г.
О.Г. Доктинова
государственный
университет»
(ЮЗГУ)

**АВТОМАТИЗАЦИЯ МОДЕЛИРОВАНИЯ ПРЕДМЕТНОЙ
ОБЛАСТИ ПРИ КОНСТРУИРОВАНИИ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ**

Методические указания по выполнению лабораторной работы
по дисциплине «Конструирование программного обеспечения»
для студентов направления подготовки 09.03.04 «Программная
инженерия»

Курск 2017

УДК 004.65

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

Автоматизация моделирования предметной области при конструировании программного обеспечения: методические указания по выполнению лабораторной работы по дисциплине "Конструирование программного обеспечения" для студентов направления подготовки 09.03.04 "Программная инженерия" / Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2017. – 16 с.: ил. 16.

Изложена последовательность действий со средой разработки Eclipse Rapyrus при моделировании предметной области на языке UML в виде диаграммы классов.

Материал предназначен для студентов направления подготовки бакалавров 09.03.04 «Программная инженерия», а также будет полезен студентам всех направлений подготовки, изучающим технологии проектирование программных систем.

Текст печатается в авторской редакции.

Подписано в печать *27.12.17*. Формат 60x84 1/16.
Усл. печ. л. *0,6*. Уч.-изд. л. *0,5*. Тираж 100 экз. Заказ *4391*. Бесплатно.
Юго-Западный государственный университет
305040, Курск, ул.50 лет Октября, 94.

Содержание

1	Цель лабораторной работы	4
2	Порядок выполнения лабораторной работы	5
2.1	Общее описание	5
2.2	Установка Parugus	6
2.3	Основные возможности Parugus	8
2.4	Пример работы в Parugus	13
3	Содержание отчета по лабораторной работе	15
4	Вопросы к защите лабораторной работы	16

1 Цель лабораторной работы

Целью лабораторной работы является приобретение знаний умений и навыков создания диаграмм классов при работе с модулем Papyrus в среде разработки Eclipse Neon 3.

При использовании Papyrus можно создавать диаграммы классов, которые в свою очередь являются отправной точкой процесса разработки. Они помогают при анализе, позволяя аналитику общаться с клиентом в «привычных» ему терминах и стимулируя процесс выявления важных деталей в проблеме, которую требуется решить.

2 Порядок выполнения лабораторной работы

2.1 Общее описание

Класс – это категория или группа вещей, которая имеет сходные атрибуты и общие свойства. Атрибуты описывают перечень значений, в рамках которых указываются свойства объектов этого класса. Операция – это то, что может выполнять класс, либо то, что вы можете выполнять над данным классом.

В UML имена классов чаще всего включают несколько слов. Каждое слово в имени класса начинается с прописной буквы, пробелы между словами отсутствуют. Имена атрибутов и операций строятся по тем же правилам, но первая буква является строчной.

Класс представляется прямоугольником, разделенным на три области (рисунок 1):

- верхняя часть содержит имя класса
- средняя часть - атрибуты
- нижняя часть – операции

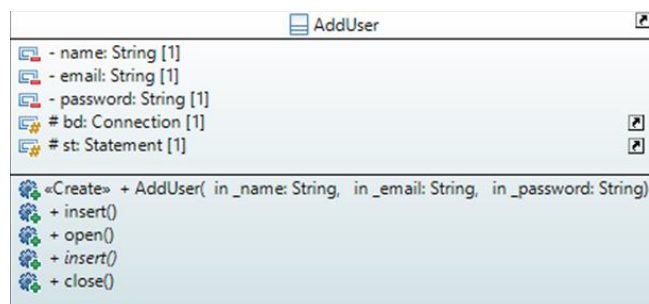


Рисунок 1

2.2 Установка Papyrus

1. Help => Eclipse Marketplace (рисунок 2).

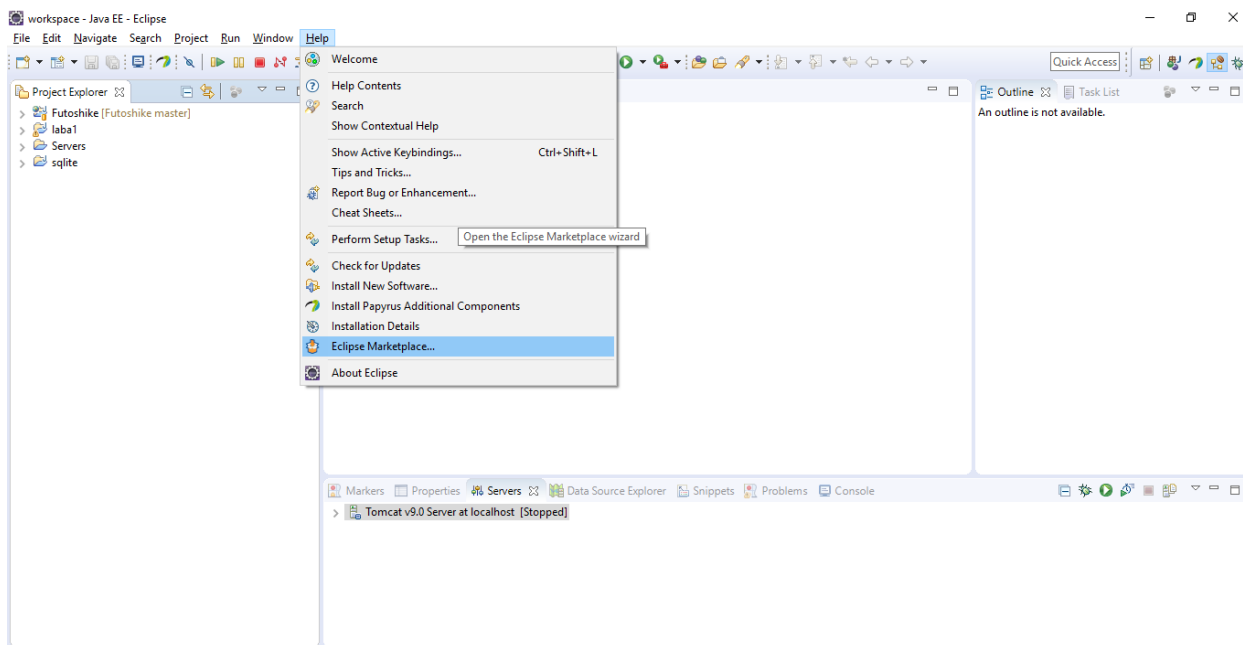


Рисунок 2

2. В появившемся окне в строке Find вводится Papyrus Designer и нажимается кнопка Install

3. Для добавления Диаграммы классов в проект нажимается правой кнопкой на проект. Выбирается New, затем Other. В появившемся меню ищется вкладку Papyrus. И выбирается Papyrus model (рисунок 3).

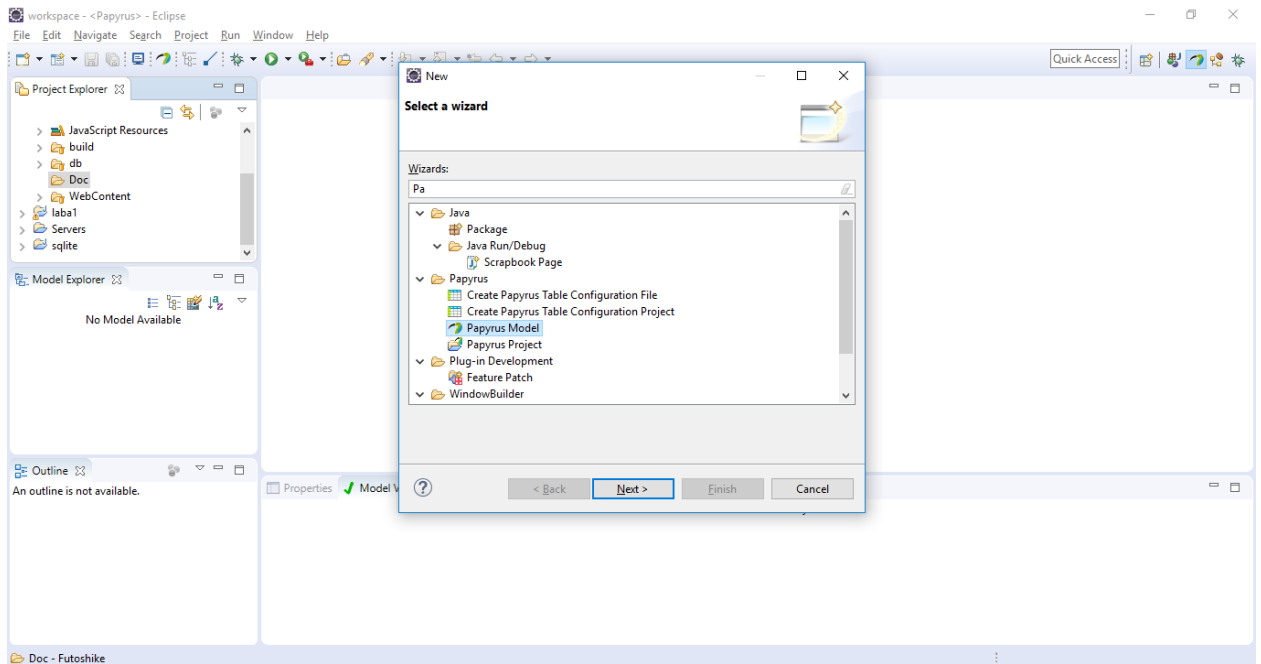


Рисунок 3

4. При создании Papyrus model язык выбираем UML и тип Class Diagram.

2.3 Основные возможности Papyrus

Рассмотрим основные возможности Papyrus при создании диаграммы классов:

Создание класса – с помощью инструмента Class на панели инструментов Palette (рисунок 4).

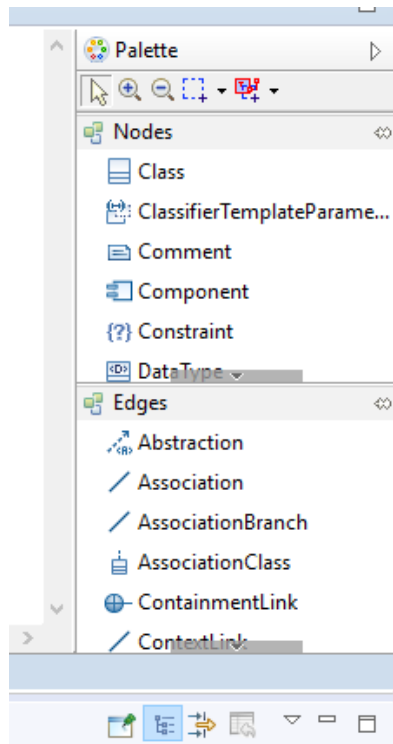


Рисунок 4

Для добавления атрибута необходимо навести курсор на класс, и подождать несколько секунд до появления следующего меню, где следует выбрать Add Property Class Attribute Label (рисунок 5).



Рисунок 5

Что бы изменить вновь созданный атрибут необходимо дважды кликнуть по нему в нижней панели (рисунок 6)

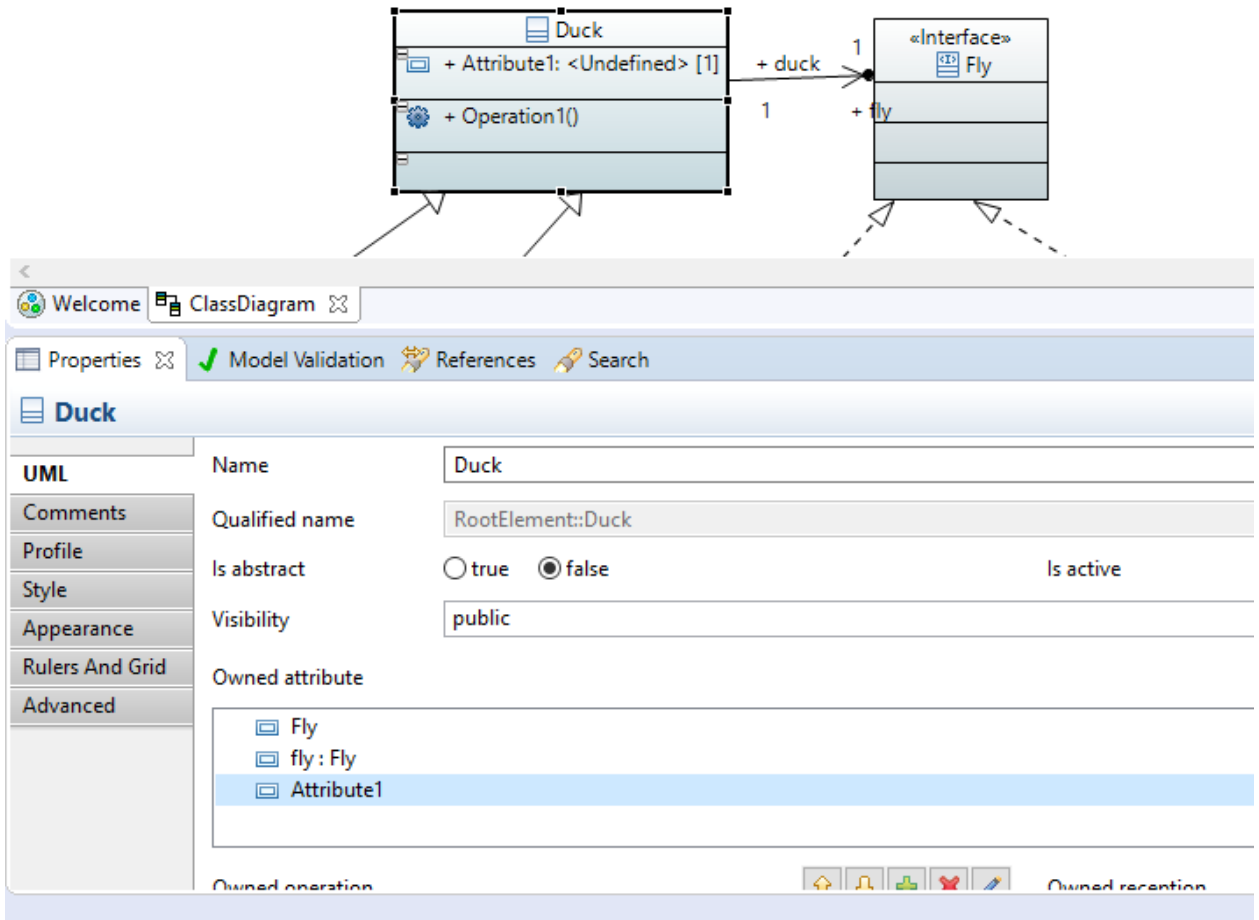


Рисунок 6

В появившемся меню можно изменить все возможные параметры атрибута (рисунок 7).

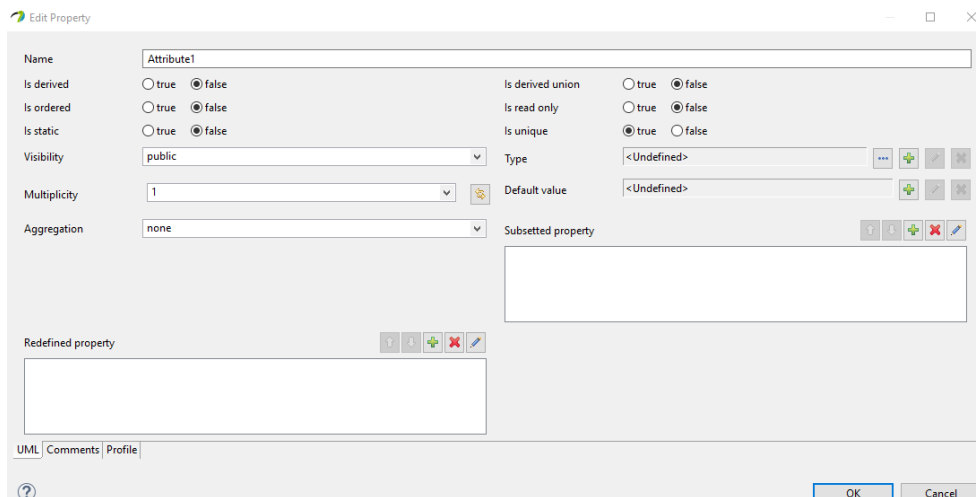


Рисунок 7

Для добавления метода необходимо навести курсор на класс, и подождать несколько секунд до появления следующего меню, где следует выбрать Add Operation Class Operation Label (рисунок 8).

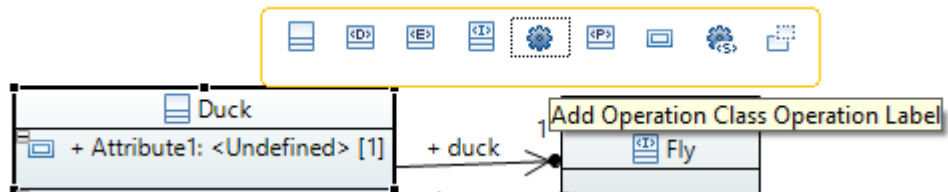


Рисунок 8

Что бы изменить вновь созданный метод необходимо дважды кликнуть по нему в нижней панели (рисунок 9)

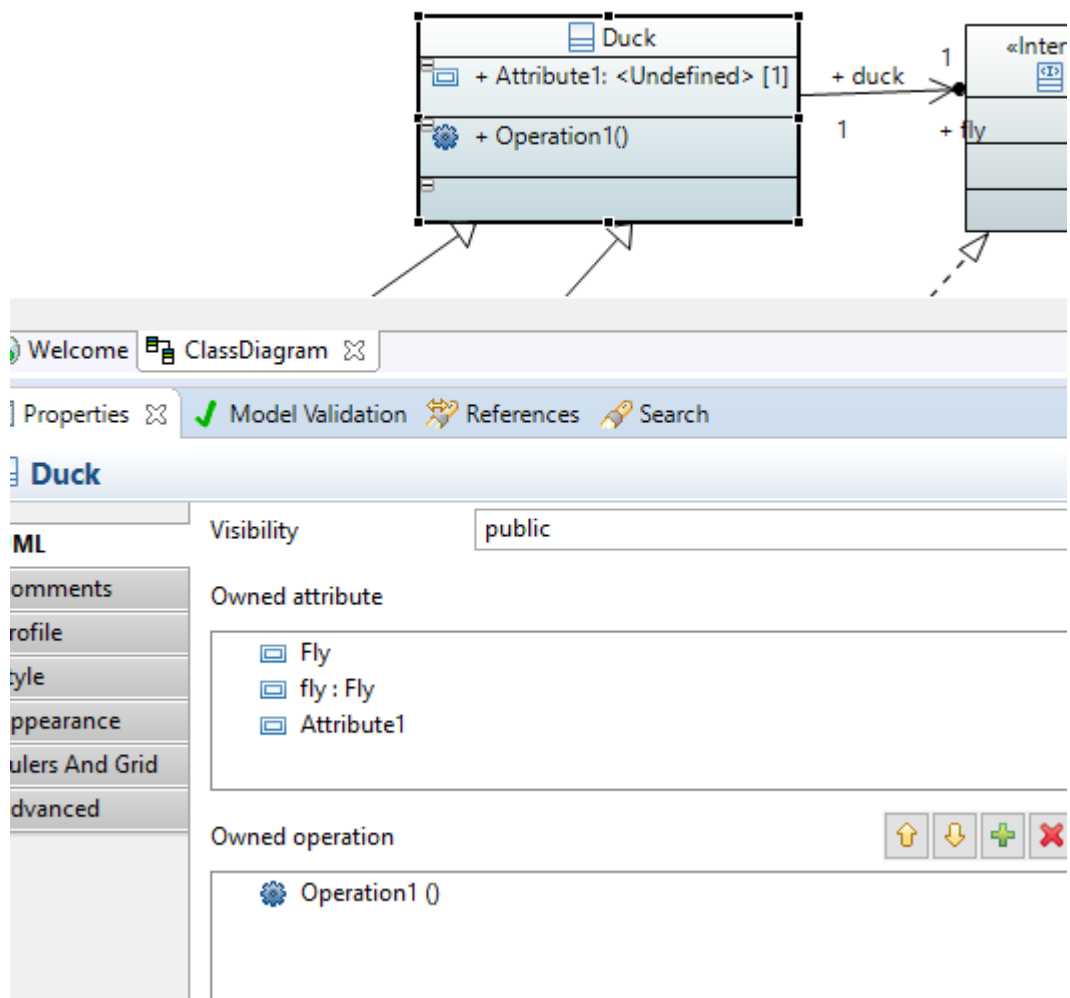


Рисунок 9

В появившемся меню можно изменить все возможные параметры метода (рисунок 10).

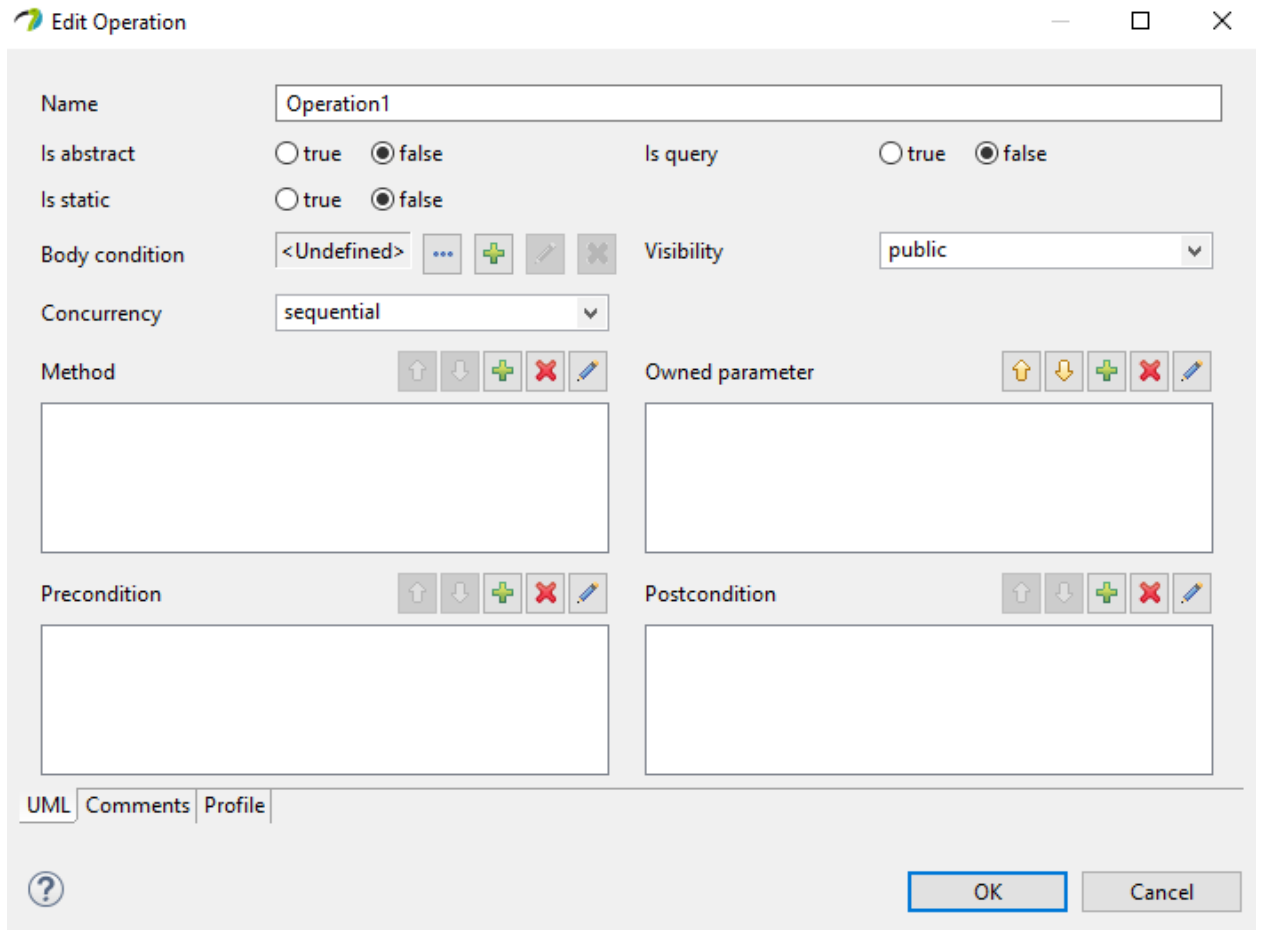


Рисунок 10

Для создания диаграммы классов, кроме самих классов и их атрибутов и методов, необходимы еще связи между классами. Ниже перечислены самые необходимые и часто используемые.

Ассоциация: если классы взаимодействуют друг с другом, то такое взаимодействие называется ассоциацией и изображается в Rarus при помощи инструмента Association.

Наследование: класс может наследовать атрибуты и операции другого класса. Наследующий класс является дочерним по отношению к родительскому, от которого он наследуется. Связь

родительского и дочернего классов соответствует отношению является видом. В Papyrus для изображения наследования используется инструмент Generalization.

Зависимость: взаимосвязь при использовании одного класса другим называется зависимостью. Наиболее общий случай зависимости – это использование одного класса в сигнатуре операции другого класса. Для изображения зависимости применяется инструмент Dependency.

Агрегация – взаимосвязь, при которой класс состоит из некоторого количества классов-компонентов, называется агрегацией. Компоненты и класс, который они составляют, находятся в ассоциации часть - целое. Агрегацию можно представить в виде дерева, корнем которого является «целое», а листьями – его компоненты. В Papyrus применяется инструмент Aggregation.

Взаимосвязь между классом и его интерфейсом называется отношением реализации (инструмент Realization) (Рисунок 11).

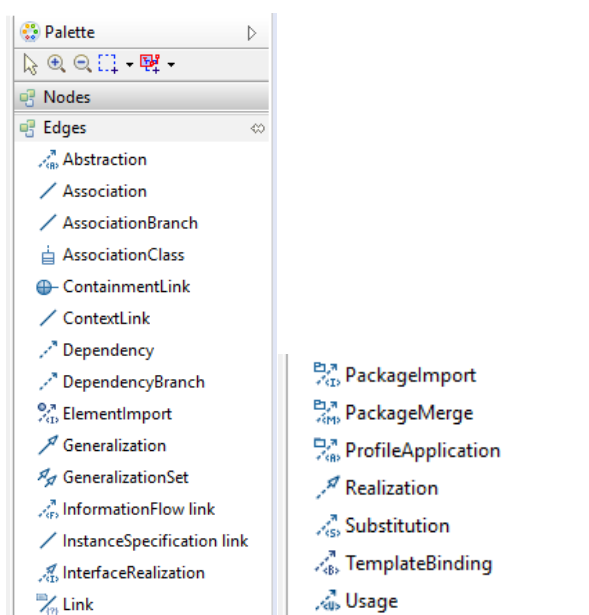


Рисунок 11

2.4 Пример работы в Rarugus

В первую очередь создаются два класса (рисунок 12).

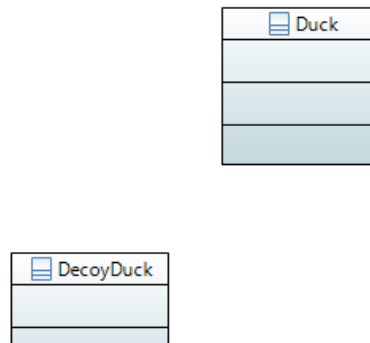


Рисунок 12

Для создания между ними наследственной связи выбирается в Palette инструмент Generalization. И соединяются, как показано на рисунке 13.

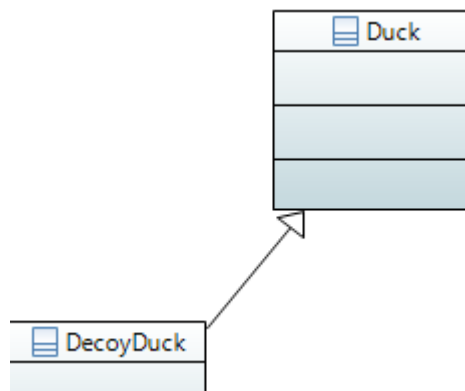


Рисунок 13

Для создания ассоциации между классами выбирается в Palette инструмент Association. . И соединяются, как показано на рисунке 14.

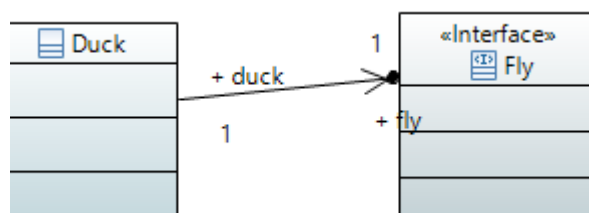


Рисунок 14

Для реализации интерфейса классом выбирается в Palette инструмент Realization. . И соединяются, как показано на рисунке 15.

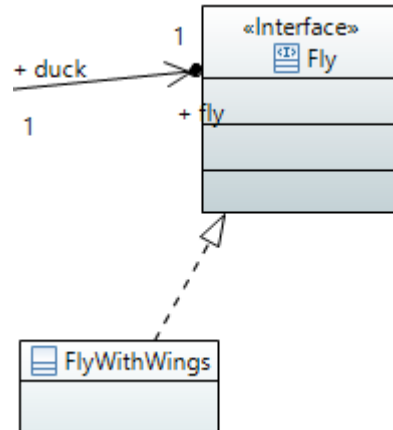


Рисунок 15

Теперь на основе вышеизложенных примеров самостоятельно создайте диаграмму классов в Rarugus на основе рисунка 16.

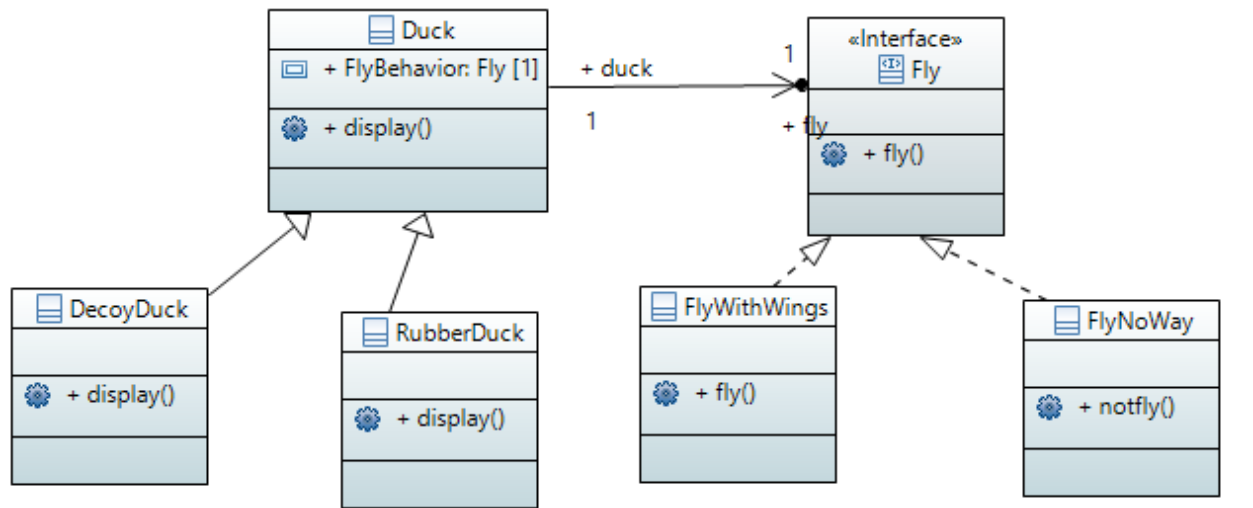


Рисунок 16

3 Содержание отчета по лабораторной работе

В сводный отчет по лабораторным работам в качестве одного из разделов или подразделов включается диаграмма классов индивидуального задания.

4 Вопросы к защите лабораторной работы

1. Зачем нужна диаграмма классов?
2. Для чего нужна связь Association?
3. Для чего нужна связь Generalization?
4. Для чего нужна связь Dependency?
5. Для чего нужна связь Aggregation?
6. Для чего нужна связь Realization?
7. Приведите пример полной записи атрибута внутри диаграммы классов.