Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт информационных технологий и анализа данных Центр программной инженерии

ОТЧЕТ

к лабораторной работе №1 по дисциплине:

|  |
| --- |
| «Технология разработки программных комплексов» |
| Шаблоны проектирования |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент |  | | | | |
| группы: | ИСТб-19-2 |  |  |  | Комогорцева Ю.В. |
| Проверил: | шифр группы  доцент |  | подпись |  | Фамилия ИО  Бахвалова З.А. |
|  | должность |  | подпись |  | Фамилия ИО |

Иркутск. 2022 г.

**Цель работы**

Ознакомиться с основными шаблонами проектирования, научиться применять их при проектировании и разработке ПО.

**Общее задание:**

Разработать каркас системы в соответствии с выданным вариантом задания.

При проектировании архитектуры **необходимо** использовать обозначенный в задании шаблон проектирования, иные варианты реализации задания **не принимаются**.

Программа должна функционировать в соответствии с приведённым в варианте задания примером работы.

Выбранный язык программирования должен поддерживать объектно-ориентированную парадигму. Выбор остального окружения производится студентом самостоятельно.

**Индивидуальный вариант**

8 Вариант. Паттерн «Фасад».

Вы принимаете участие в создании пакета для статистического анализа данных. В ваши задачи входит создание архитектуры модуля проведения описательного анализа. В ходе анализа производится расчёт среднего выборочного, медианы и моды (отдельный класс «средние величины»), расчет значений квартилей (отдельный класс «процентили распределения»), а также расчет дисперсии и среднеквадратического отклонения (отдельный класс «показатели рассеивания»).

Необходимо реализовать каркас модуля. Для обеспечения удобного интерфейса доступа ко всему описательному анализу главный класс модуля должен иметь метод «Произвести описательный анализ», который производит все вышеперечисленные операции. При запуске программа должна вызывать этот метод главного класса модуля и выводить названия операций, которые были произведены. Программа должна быть закрыта для изменения, но открыта для расширения. **Вычислений вышеприведенных характеристик производить не требуется, выводятся только названия операций.**

Пример работы программы:

Было рассчитано среднее выборочное

Была рассчитана медиана

Была рассчитана мода

Были рассчитаны значения квартилей

Были рассчитаны дисперсия и среднеквадратическое отклонение.

**1 Краткая характеристика паттерна «Фасад»**

Шаблон «Фасад» структурный шаблон проектирования, позволяющий упростить работу с подсистемой путем сведения всех возможных внешних вызовов к одному объекту, делегирующему их соответствующим объектам системы.

Фасад — это структурный паттерн проектирования, который предоставляет простой интерфейс к сложной системе классов, библиотеке или фреймворку.

При использовании паттерна Фасад мы создаем класс, который упрощает и унифицирует набор более сложных классов, образующих некую подсистему. В отличие от многих других паттернов, Фасад относительно прост; в нем нет умопомрачительных абстракций, в которых приходится подолгу разбираться. Но от этого он не становится менее полезным; паттерн Фасад предотвращает появление сильных связей между клиентом и подсистемой и, как вы вскоре увидите, способствует соблюдению нового принципа объектно-ориентированного проектирования.

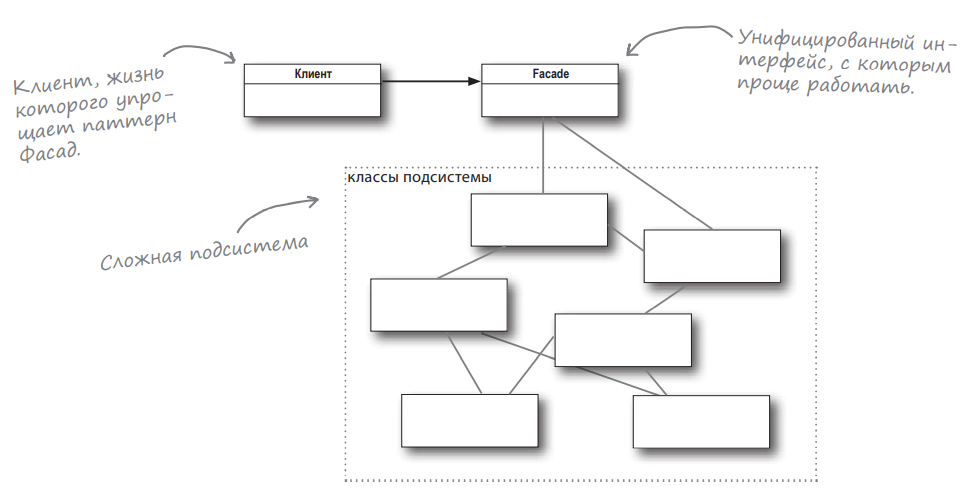


Рисунок 1 – Модель состава в форме схемы

Применение:

* Когда имеется сложная система, и необходимо упростить с ней работу. Фасад позволит определить одну точку взаимодействия между клиентом и системой.
* Когда надо уменьшить количество зависимостей между клиентом и сложной системой. Фасадные объекты позволяют отделить, изолировать компоненты системы от клиента и развивать и работать с ними независимо.
* Когда нужно определить подсистемы компонентов в сложной системе. Создание фасадов для компонентов каждой отдельной подсистемы позволит упростить взаимодействие между ними и повысить их независимость друг от друга.

Достоинства:

Недостатки:

**2 Шаги реализации**

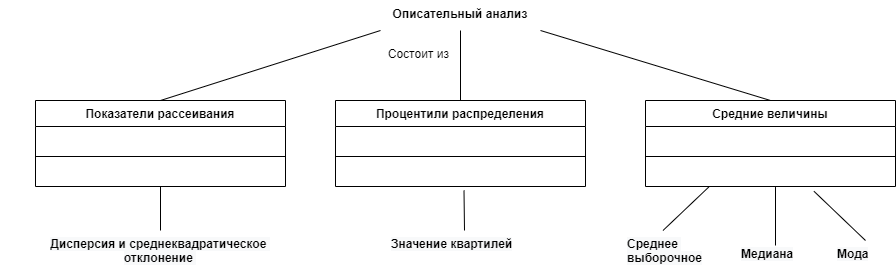
Даны классы показателей описательного анализа:

Рисунок 2 – Модель состава в форме схемы

С первого взгляда, можно было бы реализовать этот модуль через наследование классов. Например, класс «Значения квартилей» наследовался бы от класса «Процентили распределения». Однако в таком случае архитектура модуля проведения описательного анализа данных не будет соответствовать принципу открытости/закрытости: Программа должна быть закрыта для изменения, но открыта для расширения. В связи с этим необходимо добавить интерфейс IValues, в котором будет абстрагироваться расчет величин методом Calculation(). А конкретные способы расчета будут определены в реализациях этого интерфейса. Тем самым, при необходимости мы сможем добавить новые классы ,которые будут реализовывать интерфейс IValue, но не сможем внести изменения в уже существующие методы основных классов.

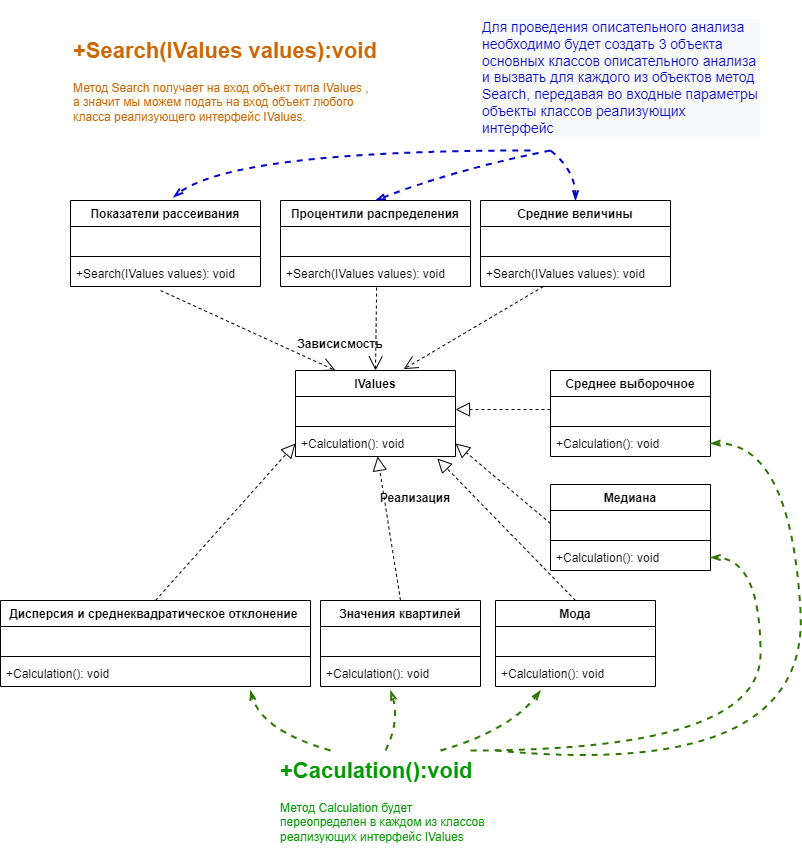


Рисунок 3 – Модель состава в форме схемы

На этом этапе архитектура модуля проведения описательного анализа выглядит завершенной, но это сложная подсистема одной большой, и нам необходимо упростить с ней работу. Фасад позволит упростить взаимодействие между другими подсистемами.

Класс фасада будет содержать 3 приватных поля объектов основных классов подсистемы описательного анализа. А также нам необходим метод MakeDescriptiveAnalysis() , который будет вызывать выполнение описательного анализа. Конечно, можно код этого метода напрямую прописывать в месте использования описательного анализа, но если это делать много раз, то код будет не эффективен. Поэтому что бы избежать повторения кода в месте использования описательного анализа мы будем создавать объект класса фасада и вызывать метод MakeDescriptiveAnalysis().

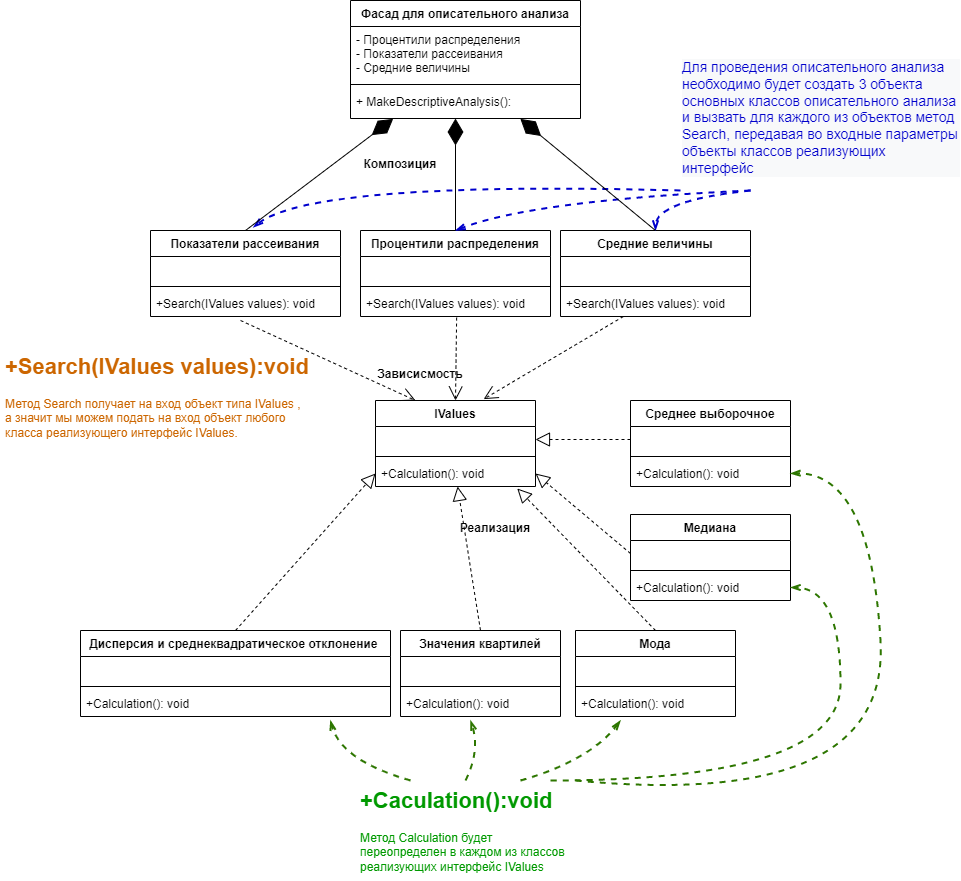


Рисунок 4 – Модель состава в форме схемы

**3 UML-диаграмма классов**

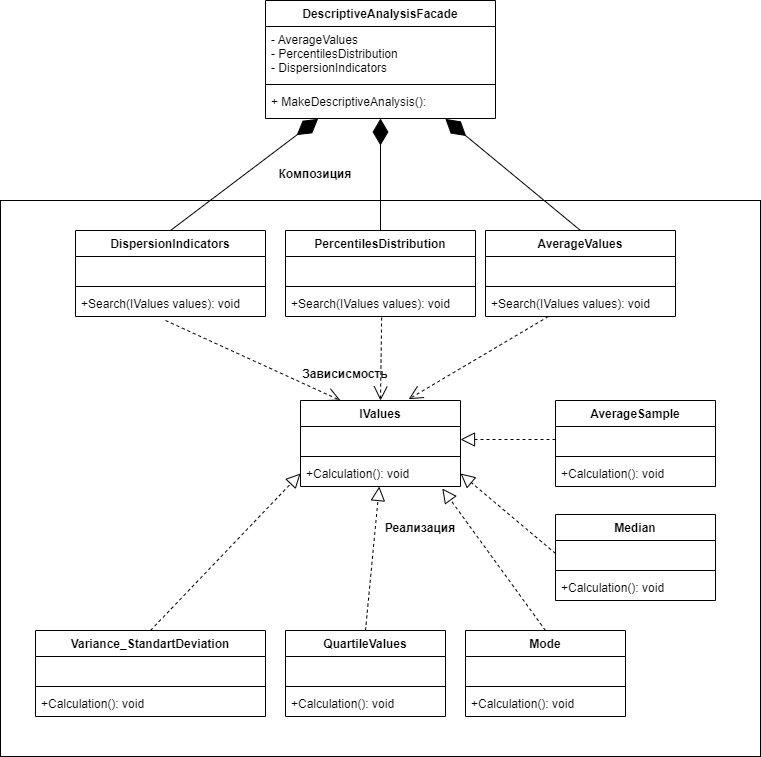


Рисунок 5 – UML диаграмма классов

**4 Результаты работы программы**

При запуске программа должна вызывать метод класса фасада модуля и выводить названия операций, которые были произведены. Вычислений вышеприведенных характеристик производить не требуется, выводятся только названия операций.

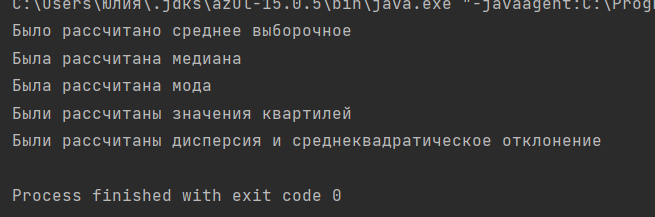


Рисунок 6 – Результат работы программы

**5 Исходный код**

# Литература

* 1. Эрик Фримен, Элизабет Робсон Head First Паттерны проектирования обновленное юбилейное издание, 2020. 656 с. ISBN: 978-5-4461-1034-6 Серия: Head First O’Reilly (дата обращения: 17.02.2022)
  2. Репозиторий с исходным кодом проекта // GitHub URL: https://github.com/KomogortsevaYulia/TRPK/tree/main/Pattern (дата обращения: 12.02.2022)
  3. Рефакторинг и паттерны проектирования. Фасад // Refactoring Guru URL: https://refactoring.guru/ru/design-patterns/facade (дата обращения: 17.02.2022)