

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА №3
по дисциплине
“Низкоуровневое программирование”

Вариант № 1
Транспортный формат XML

Студент:

Степанов Михаил
Андреевич

Группа Р33312

Преподаватель:

Кореньков Юрий Дмитриевич



Санкт-Петербург, 2023

Задание:

На базе данного транспортного формата описать схему протокола обмена информацией и воспользоваться существующей библиотекой по выбору для реализации модуля, обеспечивающего его функционирование.

Протокол должен включать представление информации о командах создания, выборки, модификации и удаления данных в соответствии с данной формой, и результатах их выполнения.

Используя созданные в результате выполнения заданий модули, разработать в виде консольного приложения две программы: клиентскую и серверную части. Серверная часть – получающая по сети запросы и операции описанного формата и последовательно выполняющая их над файлом данных с помощью модуля из первого задания. Имя файла данных для работы получать с аргументами командной строки, создавать новый в случае его отсутствия. Клиентская часть – в цикле получающая на стандартный ввод текст команд, извлекающая из него информацию о запрашиваемой операции с помощью модуля из второго задания и пересылающая её на сервер с помощью модуля для обмена информацией, получающая ответ и выводящая его в человеко- понятном виде в стандартный вывод.

Порядок выполнения:

1. Изучить выбранную библиотеку
 - a. Библиотека должна обеспечивать сериализацию и десериализацию с валидацией в соответствии со схемой
 - b. Предпочтителен выбор библиотек, поддерживающих кодогенерацию на основе схемы
 - c. Библиотека может поддерживать передачу данных посредством TSP соединения. Иначе, использовать сетевые сокеты посредством API ОС
 - d. Библиотека может обеспечивать диспетчеризацию удалённых вызовов. Иначе, реализовать диспетчеризацию вызовов на основе информации о виде команды
2. На основе существующей библиотеки реализовать модуль, обеспечивающий взаимодействие
 - a. Описать схему протокола в поддерживаемом библиотекой формате

Описание должно включать информацию о командах, их аргументах и результатах. Схема может включать дополнительные сущности (например, для итератора)

b. Подключить библиотеку к проекту и сформировать публичный интерфейс модуля с использованием встроенных или сгенерированных структур данных используемой библиотеки. Поддерживать установление соединения, отправку команд и получение их результатов. Поддерживать приём входящих соединений, приём команд и отправку их результатов.

c. Реализовать публичный интерфейс посредством библиотеки в соответствии с п1

3. Реализовать серверную часть в виде консольного приложения

a. В качестве аргументов командной строки приложение принимает: Адрес локальной конечной точки для прослушивания входящих соединений. Имя файла данных, который необходимо открыть, если он существует, иначе создать.

b. Работает с файлом данных посредством модуля из задания 1

c. Принимает входящие соединения и взаимодействует с клиентами посредством модуля из п2

d. Поступающая информация о запрашиваемых операциях преобразуется из структур данных модуля взаимодействия к структурам данных модуля управления данными и наоборот.

4. Реализовать клиентскую часть в виде консольного приложения

a. В качестве аргументов командной строки приложение принимает адрес конечной точки для подключения.

b. Подключается к серверу и взаимодействует с ним посредством модуля из п2

c. Читает со стандартного ввода текст команд и анализирует их посредством модуля из задания 2

d. Преобразует результат разбора команды к структурам данных модуля из п2, передаёт их для обработки на сервер, возвращаемые результаты выводит в стандартный поток вывода.

5. Результаты тестирования представить в виде отчёта, в который включить:
- d. В части 3 привести пример сеанса работы разработанных программ
 - e. В части 4 описать решение, реализованное в соответствии с пп.2-4
 - f. В часть 5 включить составленную схему п.2а

Аспекты реализации:

Lab1 представляет модуль работы с базой данных. Lab2 представляет модуль обработки клиентских запросов.

request_schema.xsd и response_schema.xsd - файлы, в которых описана схема запроса и ответа, через которые общаются клиент и сервер.

При выполнении лабораторной использовалась библиотека, поддерживающая кодогенерацию на основе схемы - *XSD Code Synthesis*.
Сетевое взаимодействие реализовано при помощи сокетов API ОС.

Результаты:

Пример №1:

```
Car(insert: {
    {
        node_name: Volkswagen,
        props {
            cyl : 123,
            privet : "privet"
        }
    },
    {
        node_name: Audi,
        props{
            capacity: 5,
            model: "A6"
        }
    }
})
```

```
        relations{
          cyl: "Volkswagen"
        }
      }
    }){
      id
    };
  };
```

-----RESPONSE-----

Status: 200

Message: Insert done.

Is finished: 1

Body:

Node (Name: Volkswagen Class: Car)

Field (Name: id Value: 1)

Node (Name: Audi Class: Car)

Field (Name: id Value: 2)

Пример №2:

```
Car(select: true, node_name: "Audi"){
  id,
  relations.cyl
};
```

-----RESPONSE-----

Status: 200

Message: Select done.

Is finished: 1

Body:

Node (Name: Audi Class: Car)

Field (Name: id Value: 2)

Relation:

Node (Name: Volkswagen Class: Car)

Field (Name: cyl Value: 123)

Field (Name: privet Value: "privet")

Вывод:

В ходе выполнения лабораторной работы были разработаны два модуля, которые в совокупности представляют клиент-серверное приложение. Также был разработан протокол сообщения между модулями при помощи xml-схемы.