

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития  
Кафедра инфокоммуникаций

**ОТЧЕТ**  
**ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 3**  
**дисциплины «Анализ данных»**

Выполнил:  
Степанов Леонид Викторович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение  
средств вычислительной  
техники и автоматизирование  
систем», очная форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Р.А., канд. техн. наук,  
доцент, доцент кафедры  
инфокоммуникаций

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

Тема: Разработка приложений с интерфейсом командной строки (CLI) в Python3

Цель: приобретение построения приложений с интерфейсом командной строки с помощью языка программирования Python версии 3.x.

Ход работы:

1. В файле primer1.py отработал 1 пример практической работы, на рис.1 отображены страницы руководства приложения.

```
(poetry) D:\Python-labs\lab3>python prog\primer.py --help
usage: workers [-h] [--version] {add,display,select} ...

positional arguments:
  {add,display,select}
    add                Add a new worker
    display            Display all workers
    select             Select the workers

optional arguments:
  -h, --help            show this help message and exit
  --version             show program's version number and exit

(poetry) D:\Python-labs\lab3>python prog\primer.py add --help
usage: workers add [-h] -n NAME [-p POST] -y YEAR filename

positional arguments:
  filename              The data file name

optional arguments:
  -h, --help            show this help message and exit
  -n NAME, --name NAME  The worker's name
  -p POST, --post POST  The worker's post
  -y YEAR, --year YEAR  The year of hiring

(poetry) D:\Python-labs\lab3>
```

Рисунок 1 – Страницы руководства

2. В файле ind.py написал программу по индивидуальному заданию в Лабораторной работе 2.16, реализовав интерфейс командной строки.

```

(poetry) D:\Python-labs\lab3>python prog\ind.py --help
usage: products [-h] [--version] {add,display,info} ...

positional arguments:
  {add,display,info}
    add                Add a new product
    display            Display all products
    info              Select the products

optional arguments:
  -h, --help          show this help message and exit
  --version           show program's version number and exit

(poetry) D:\Python-labs\lab3>python prog\ind.py add --help
usage: products add [-h] -n NAME [-m MARKET] -c COUNT filename

positional arguments:
  filename            The data file name

optional arguments:
  -h, --help          show this help message and exit
  -n NAME, --name NAME The product's name
  -m MARKET, --market MARKET
                     The market's name
  -c COUNT, --count COUNT
                     The count

(poetry) D:\Python-labs\lab3>python prog\ind.py info --help
usage: products info [-h] -p NAME_PRODUCT filename

positional arguments:
  filename            The data file name

optional arguments:
  -h, --help          show this help message and exit
  -p NAME_PRODUCT, --name_product NAME_PRODUCT
                     The required name of product

(poetry) D:\Python-labs\lab3>

```

Рисунок 2 – Страницы руководства индивидуального задания

3. В файле high.py написал программу по индивидуальному заданию в Лабораторной работе 2.16, реализовав интерфейс командной строки при помощи модуля click.

```

(poetry) D:\Python-labs\lab3>python prog\high.py add --help
Usage: high.py add [OPTIONS] FILENAME

    Добавить данные о работнике.

Options:
  --product_name TEXT  Name of product
  --market_name TEXT   Name of market
  --value INTEGER      Value of product
  --help               Show this message and exit.

(poetry) D:\Python-labs\lab3>python prog\high.py --help
Usage: high.py [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  add      Добавить данные о работнике.
  display  Отобразить список работников.
  select   Выбрать продукт с заданным именем.

(poetry) D:\Python-labs\lab3>python prog\high.py select --help
Usage: high.py select [OPTIONS] FILENAME TWO_FILENAME NAME

    Выбрать продукт с заданным именем.

Options:
  --help  Show this message and exit.

(poetry) D:\Python-labs\lab3>

```

Рисунок 3 – Страницы руководства задания повышенного уровня

Вывод: в ходе лабораторной работы были приобретены знания о построение приложения с интерфейсом командной строки с помощью языка программирования Python версии 3.x.

Контрольные вопросы:

1. Отличие между терминалом и консолью

Терминал и консоль – это термины, связанные с работой в командной строке операционной системы.

Терминал – это физическое устройство, которое позволяет пользователю взаимодействовать с компьютером посредством текстового интерфейса. Консоль – это программное обеспечение, предоставляющее пользователю доступ к командной строке операционной системы.

## 2. Консольное приложение и его определение

Консольное приложение – это программа, которая работает в командной строке операционной системы. Она взаимодействует с пользователем через текстовый интерфейс, принимая команды и предоставляя результаты выполнения.

## 3. Средства языка программирования Python для построения приложений командной строки

Для построения приложений командной строки на языке программирования Python существуют несколько средств:

`sys.argv` - это список аргументов командной строки, передаваемых при запуске скрипта на Python.

`getopt` – модуль Python для парсинга аргументов командной строки.

`argparse` – модуль Python для создания гибких командных интерфейсов.

## 4. Особенности построения CLI с использованием модуля `sys`

Модуль `sys` в Python предоставляет доступ к некоторым переменным и функциям, связанным с интерпретатором Python и его окружением. Он позволяет работать с аргументами командной строки и другими системными параметрами.

## 5. Особенности построения CLI с использованием модуля `getopt`

Модуль `getopt` в Python предоставляет средства для парсинга аргументов командной строки. Он позволяет обрабатывать опции и аргументы командной строки, упрощая разработку приложений командной строки.

## 6. Особенности построения CLI с использованием модуля `argparse`

Модуль `argparse` в Python предоставляет более гибкие средства для создания командных интерфейсов. Он позволяет определять аргументы, опции и подкоманды, а также автоматически генерировать справку для пользователей.