

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ ФГАОУ ВО  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Институт цифрового развития  
Кафедра инфокоммуникаций**

**Дисциплина «Языки программирования»  
Отчет по практической работе № 2.18  
Работа с переменными окружения в Python3**

**Выполнил: студент группы ИТС-б-о-21-1  
Крамаренко Илья Витальевич**

---

**(подпись)**

**Проверил: к.т.н., доцент  
Кафедры инфокоммуникаций  
Воронкин Р.А.**

---

**(подпись)**

**Ставрополь, 2022**

**Цель работы:** приобретение навыков по работе с переменными окружения с помощью языка программирования Python версии 3.x.

Ссылка на репозиторий <https://github.com/danilusikov0913/YPIr8>

**Ход работы:**

## 1. Пример

Для хранения имени файла данных будем использовать переменную окружения `WORKERS_DATA`.

При этом сохраним возможность передавать имя файла данных через именной параметр `--data`.

Иными словами, если при запуске программы в командной строке не задан параметр `--data`, то

имя файла данных должно быть взято из переменной окружения `WORKERS_DATA`

После успешного составления программы, нужно добавить переменную среду

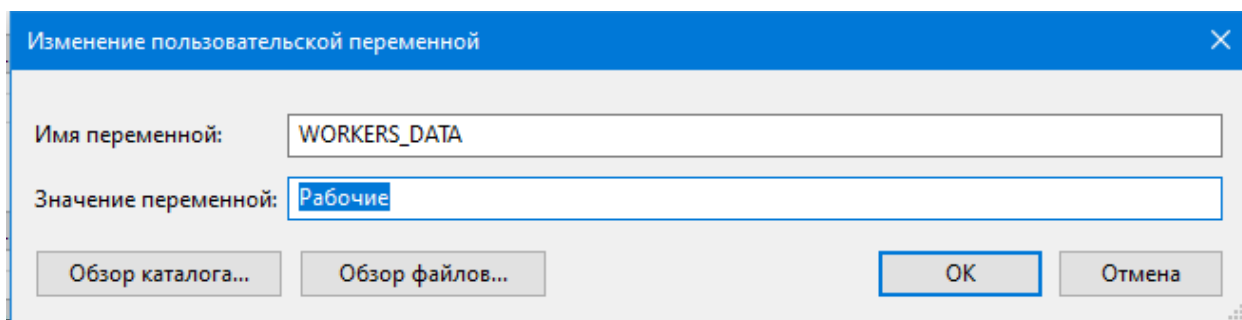


Рисунок 1 – Добавление переменной среды

```
C:\MonstR\8>python lr8prim.py add --name="Сидоров Сидор" --post="Главный инженер" --year=2012
C:\MonstR\8>python lr8prim.py display
+-----+-----+-----+-----+
| No | Ф.И.О. | Должность | Год |
+-----+-----+-----+-----+
| 1 | Сидоров Сидор | Главный инженер | 2012 |
+-----+-----+-----+-----+

C:\MonstR\8>python lr8prim.py select --period=10
+-----+-----+-----+-----+
| No | Ф.И.О. | Должность | Год |
+-----+-----+-----+-----+
| 1 | Сидоров Сидор | Главный инженер | 2012 |
+-----+-----+-----+-----+

C:\MonstR\8>
```

Рисунок 2 – Результат работы

## 2. Индивидуальное задание 1

### Задание 1

Для своего варианта лабораторной работы 2.17 добавьте возможность получения имени файла данных, используя соответствующую переменную окружения.

Рисунок 4 – Код индивидуального задания

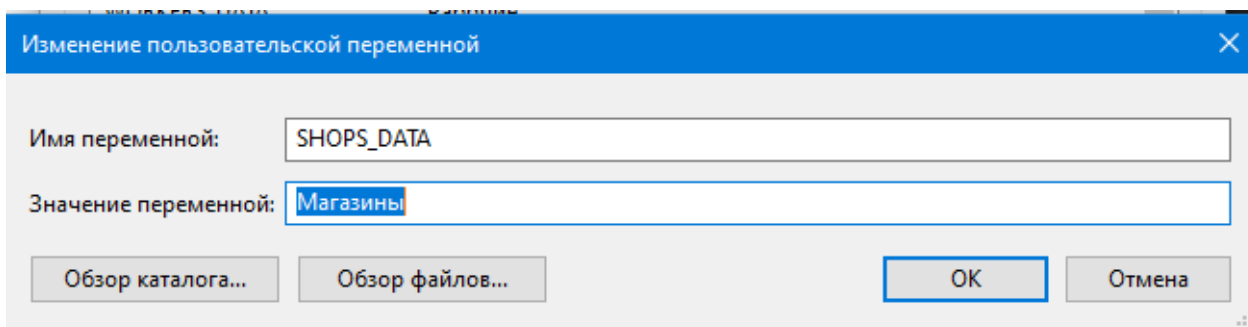


Рисунок 5 – другая переменная среда

```
C:\MonstR\8>python lr8iz.py add -ns="Магнит" -pr="Молоко" -ps=72
C:\MonstR\8>python lr8iz.py display
+-----+-----+-----+-----+
| No | Название магазина | Название продукта | Стоимость товара |
+-----+-----+-----+-----+
| 1 | Магнит | Молоко | 72 |
+-----+-----+-----+-----+
C:\MonstR\8>python lr8iz.py select -SS="Молоко"
+-----+-----+-----+-----+
| No | Название магазина | Название продукта | Стоимость товара |
+-----+-----+-----+-----+
| 1 | Магнит | Молоко | 72 |
+-----+-----+-----+-----+
C:\MonstR\8>python lr8iz.py select -SS="Сыр"
Список магазинов пуст.
C:\MonstR\8>
```

Рисунок 6 – Результат работы

### Контрольные вопросы:

1. Каково назначение переменных окружения?

Переменная среды (переменная окружения) – это короткая ссылка на какой-либо объект в системе. С помощью таких сокращений, например, можно

создавать универсальные пути для приложений, которые будут работать на любых ПК, независимо от имен пользователей и других параметров.

## 2. Какая информация может храниться в переменных окружения?

Переменная окружения может хранить информацию о путях к исполняемым файлам, заданном по умолчанию текстовом редакторе, браузере, языковых параметрах (локали) системы или настройках раскладки клавиатуры.

## 3. Как получить доступ к переменным окружения в ОС Windows?

Компьютер, свойства, дополнительные параметры и среды, дополнительно, переменные среды

## 4. Каково назначение переменных PATH и PATHEXT?

«PATH» позволяет запускать исполняемые файлы и скрипты, «лежащие» в определенных

PATHEXT, в свою очередь, дает возможность не указывать даже расширение файла, если оно прописано в ее значениях каталогов, без указания их точного местоположения.

## 5. Как создать или изменить переменную окружения в Windows?

Компьютер, свойства, дополнительные параметры и среды, дополнительно, переменные среды, создать или изменить

## 6. Что представляют собой переменные окружения в ОС Linux?

Переменные окружения в Linux представляют собой набор именованных значений, используемых другими приложениями.

## 7. В чем отличие переменных окружения от переменных оболочки?

Переменные окружения и оболочки всегда присутствуют в сеансах оболочки и могут быть очень полезны. Они позволяют родительским процессам устанавливать детали конфигурации для своих дочерних процессов и являются способом установки определенных параметров без использования отдельных файлов.

## 8. Как вывести значение переменной окружения в Linux?

## 9. Какие переменные окружения Linux Вам известны?

10. Какие переменные оболочки Linux Вам известны?
11. Как установить переменные оболочки в Linux?
12. Как установить переменные окружения в Linux?
13. Для чего необходимо делать переменные окружения Linux постоянными?

14. Для чего используется переменная окружения PYTHONHOME ?

Переменная среды PYTHONHOME изменяет расположение стандартных библиотек Python. По умолчанию библиотеки ищутся в `prefix/lib/pythonversion` и `exec_prefix/lib/pythonversion`, где `prefix` и `exec_prefix` - это каталоги, зависящие от установки, оба каталога по умолчанию - `/usr/local`.

Когда для PYTHONHOME задан один каталог, его значение заменяет `prefix` и `exec_prefix`. Чтобы указать для них разные значения, установите для PYTHONHOME значение `prefix:exec_prefix`

15. Для чего используется переменная окружения PYTHONPATH ?

Переменная среды PYTHONPATH изменяет путь поиска по умолчанию для файлов модуля.

Формат такой же, как для оболочки PATH : один или несколько путей к каталогам, разделенных `os.pathsep` (например, двоеточие в Unix или точка с запятой в Windows). Несуществующие каталоги игнорируются. `$ unset NEW_VAR`

Помимо обычных каталогов, отдельные записи PYTHONPATH могут относиться к zip-файлам, содержащим чистые модули Python в исходной или скомпилированной форме. Модули расширения нельзя импортировать из zip-файлов.

Путь поиска по умолчанию зависит от установки Python, но обычно начинается с префикса `/lib/pythonversion`. Он всегда добавляется к PYTHONPATH.

16. Какие еще переменные окружения используются для управления работой интерпретатора Python?

PYTHONSTARTUP PYTHONOPTIMIZE PYTHONBREAKPOINT PYTHONDEBUG  
PYTHONINSPECT PYTHONUNBUFFERED  
PYTHONVERBOSE PYTHONCASEOK PYTHONDONTWRITEBYTECODE  
PYTHONPYCACHEPREFIX PYTHONHASHSEED PYTHONIOENCODING PYTHONNOUSERSITE  
PYTHONUSERBASE PYTHONWARNINGSPYTHONFAULTHANDLER PYTHONTRACEMALLOC  
PYTHONPROFILEIMPORTTIME PYTHONASYNCIODEBUG  
PYTHONMALLOC PYTHONMALLOCSTATS  
PYTHONLEGACYWINDOWSFSENCODING PYTHONLEGACYWINDOWSSSTDIO  
PYTHONCOERCECLOCALE

17. Как осуществляется чтение переменных окружения в программах на языке программирования Python?

Для начала потребуется импортировать модуль `os`, чтобы считывать переменные. Для доступа к переменным среды в Python используется объект `os.environ`. С его помощью программист может получить и изменить значения всех переменных среды. Далее мы рассмотрим различные способы чтения, проверки и присвоения значения переменной среды.

18. Как проверить, установлено или нет значение переменной окружения в программах на языке программирования Python?

19. Как присвоить значение переменной окружения в программах на языке программирования Python?

Для начала потребуется импортировать модуль `os`, чтобы считывать переменные. Для доступа к переменным среды в Python используется объект `os.environ`. С его помощью программист может получить и изменить значения всех переменных среды. Далее мы рассмотрим различные способы чтения, проверки и присвоения значения переменной среды.

**Вывод:** в ходе лабораторной работы приобретены навыки построения приложений с интерфейсом командной строки с помощью языка программирования Python версии 3.x.