Смартфон, как и любой другой современный гаджет, работающий на ОС, может работать некорректно, или выходить из строя из-за различных программных ошибок. Может возникнуть зависание, самопроизвольная перезагрузка и он станет работать медленнее.

Перечисленные проблемы могут свидетельствовать об неполадках или программных сбоях. Например, Android является открытой системой, доступ к файлам которой сможет получить любой продвинутый пользователь. При повреждении системных файлов устройство может начать работать некорректно.

Для установления причин неисправностей рекомендуют обращаться в специализированные сервисные центры. Опытные мастера продиагностируют телефон, определят причины возникших неисправностей, установят новую прошивку или произведут замену вышедших из строя компонентов.

Теперь можно поговорить о самом главном это цена самой системы. На самом деле операционные системы имеют разные цены они зависят от качества сборки и материалов из которых их собрали. Если сравнивать две операционных систем Android и Ios, то это совершенно разные системы, которые имеют разную ценовую категорию. Ios система намного дороже Android, но в этом есть свои плюсы, она некогда не стареет и не выходит из строя в этом Android уступает Ios.

После рассмотрения всех видов мобильных операционных систем устройств, можно сделать вывод, что смартфоны в нашем мире – явление стремительно новое. Современные смартфоны являются признаком коммерчески занятого человека, которому необходимо устройство, или фаната новинок электронной техники.

Предшественниками современного смартфона были карманные коммуникаторы, которые тоже использовали в своем устройстве ОС. Однако ОС, которые использовали коммуникаторы в 1990-х годах, значительно отличались от современных ОС и имели в своей структуре намного меньше функций. С этих ОС и начинается история мобильных устройств.

Теперь можно поговорить о самом главном это цена самой системы. На самом деле операционные системы имеют разные цены они зависят от качества сборки и материалов из которых их собрали. Если сравнивать две операционных систем Android и Ios то это совершенно разные системы, которые имеют разную ценовую категорию. Ios система намного дороже Android, но в этом есть свои плюсы, она некогда не стареет и не выходит из строя в этом Android уступает Ios.

***

1. История Android: эволюция самой популярной операционной системы в мире [Электронный ресурс]. -URL: <https://trashbox.ru/link/history-of-android> (дата обращения 10.11.21) ;
2. Эволюция мобильных ОС: истории провалов и успеха Об этом сообщает "Рамблер"[Электронный ресурс].-URL: https://news.rambler.ru/gadgets/43712769/?utm_content=news_media&utm_medium=read_more&utm_source=copylink (дата обращения 10.11.21) ;
3. Развитие операционных систем мобильных устройств [Электронный ресурс]. -URL: <https://scienceforum.ru/2017/article/2017040137> (дата обращения 10.11.21).

Хаджиев А.В., Магамедова Д.М.

Возможности встроенного модуля os при работе с файловой системой в языке Python

*Чеченский государственный педагогический университет
(Россия, Грозный)*

doi: 10.18411/trnio-01-2022-53

Аннотация

В статье описываются способы работы базовых функций встроенного модуля os языка программирования Python при взаимодействии с установленной операционной системой и ее иерархией файлов.

Ключевые слова: язык программирования python, возможности модуля os, функции модуля, файловая система, интерпретатор.

Abstract

This article describes how the basic functions of the Python built-in os module work when interacting with the installed operating system and its file hierarchy.

Keywords: python programming language, os module capabilities, module functions, file system, interpreter.

Встроенные функции в языке программирования Python, позволяющие выполнять различного рода итерации, доступны для разработчика без установки дополнительных библиотек. При необходимости решения математической проблемы, разработчик прописывает в интерпретаторе блок команд для его разрешения с базовыми операторами, чтобы манипулировать со значениями целочисленных и/или вещественных чисел (операндов) переменных. Но чтобы сэкономить время при разработке программы, не переписывая сценарии, разработчиками задаются собственные или же используются иные функции из различных импортируемых модулей. Об одном из таких модулей посвящена данная статья.

Модуль os предназначен для работы с установленной операционной системой и, в частности, с её файлами. Он содержит в себе массу методов для манипуляции над файлами (как пример, возможности обрабатывать, создавать, переименовывать, перемещать, удалять файлы и папки; возможность выводить списки файлов и каталоги). Помимо этого, данный модуль включает в себя методы и для более глубокого рода операций, таких как обработка переменных среды, управление системными процессами, возможности задавать аргументы командной строки и так далее. Кроме того, os имеет в себе расширенные атрибуты файлов, которые есть в операционной системе Linux.

Процедура подключения модуля проходит базовым способом: чтобы извлечь модуль как единое целое, импортеру необходимо прописать команду `import` и, как правило, приписывается она в начале сценария. Чтобы извлечь отдельные функции из модуля, задается команда `from os import название_функции`. Рассмотрим некоторые возможности модуля на примере использования его методов.

Метод `os.name` используется для выявления имени операционной системы и, в зависимости от установленной платформы, функция возвращает имя системы в строковом представлении. Код программы выглядит следующим образом:

```
[In:1] import os
[In:2] print(os.name)
[Out:2]nt
```

Результатом является выведенная строка `nt`, потому что программа была запущена на платформе Windows 10.

Метод `os.environ` позволяет выводить сведения о конфигурации компьютера в виде большого словаря с переменными окружения ОС, который можно изменять:

```
[In:1] import os
[In:2] print(os.environ)
[Out:2]environ({'ALLUSERSPROFILE': 'C:\\ProgramData', ...})
```

С помощью метода `os.getcwd` можно получить абсолютный путь до текущего файла, как пример, интерпретатора:

```
[In:1] import os
[In:2] print(os.getcwd())
[Out:2]C:\\Users\\Имя\\AppData\\Local\\Programs\\Python\\Python310
```

Для создания папки или каталога существуют две разновидности команд - `os.mkdir` и `os.makedirs`. Возможности этих двух функций кроются в их названиях, так, `mkdir` предназначена для создания одного каталога и при повторном запуске программа выдаст

ошибку, потому что папка с указанным именем уже создана, а `makedirs` способна создавать бесконечную вложенную структуру:

```
[In:1] import os
[In:2] os.mkdir("папка")
[In:3] os.makedirs("папка")
```

Перемещение файла или папки осуществляется функцией `os.replace`, для переименования файла или папки используется функция `os.rename` более того, файл с новым именем может располагаться в другой папке.

Функция `os.listdir` выводит список файлов и директорий в указанной папке:

```
[In:2] print(os.listdir(path))
[Out:2] ['файл.txt', 'snd_dir', 'проект.py', 'fst_dir']
```

Для осуществления удаления папки или целого каталога используются две функции - `os.rmdir` и `os.removedirs`. Первая функция `rmdir` удаляет указанный пользователем каталог, но в том случае, если указанный каталог не содержит файлов, то есть, если каталог пустой. Вторая функция `removedirs` удаляет всю цепочку вложенных каталогов. Пользователю стоит быть очень внимательным используя приведенные выше функции удаления, чтобы случайно не задеть нужные каталоги.

Метод `os.startfile` позволяет запускать файлы и папки напрямую из программы. Осуществляется это следующим образом:

```
[In:1] import os
[In:2] os.startfile(r"D:\test.txt")
```

Метод `walk`, можно получить доступ к названиям и путям всех подпапок и файлов, относящихся к заданному каталогу:

```
[In:1] import os
[In:2] for root, directories, files in os.walk(r"D:\1PMI(3)"):
[In:3]     print(root)
[In:4]     for directory in directories:
[In:5]         print(directory)
[In:6]     for file in files:
[In:7]         print(file)
```

В статье были приведены и рассмотрены базовые функции модуля `os` языка Python. За счет множества методов по управлению операционной системой модуль `os` предоставляет не только возможность получать сведения о платформе и ее прилегающих файлах, но и работать с этими же файлами, хранящиеся в дисках, создавать новые файлы и директории, переименовывать и удалять их.

***

1. Лутц, М. Изучаем Python / М.Л. Лутц. – г. Чехов: ООО “Диалектика”, 2019. – 819 с.
2. Хеллман, Д. Стандартная библиотека Python 3: справочник с примерами/ Д. Хеллман.– г. Чехов: ООО “Диалектика”, 2019. – 1361 с.
3. Рамальо, Л. Python. К вершинам мастерства / Л. Рамальо. – г. Москва: ДМК-Пресс, 2016. – 768 с.

Mitrofanov V.S.

MFS Risk Monitoring

*Reshetnev Siberian State University of Science and Technology
(Russia, Krasnoyarsk)*

doi: 10.18411/trnio-01-2022-54

Abstract

Everything we do while managing a software development project should be aimed at dealing with the risks of not meeting deadlines, overspending resources, developing the wrong product that is required.