

DataOps.onETL



Unit#7

Объект скачивания данных FileDownloader



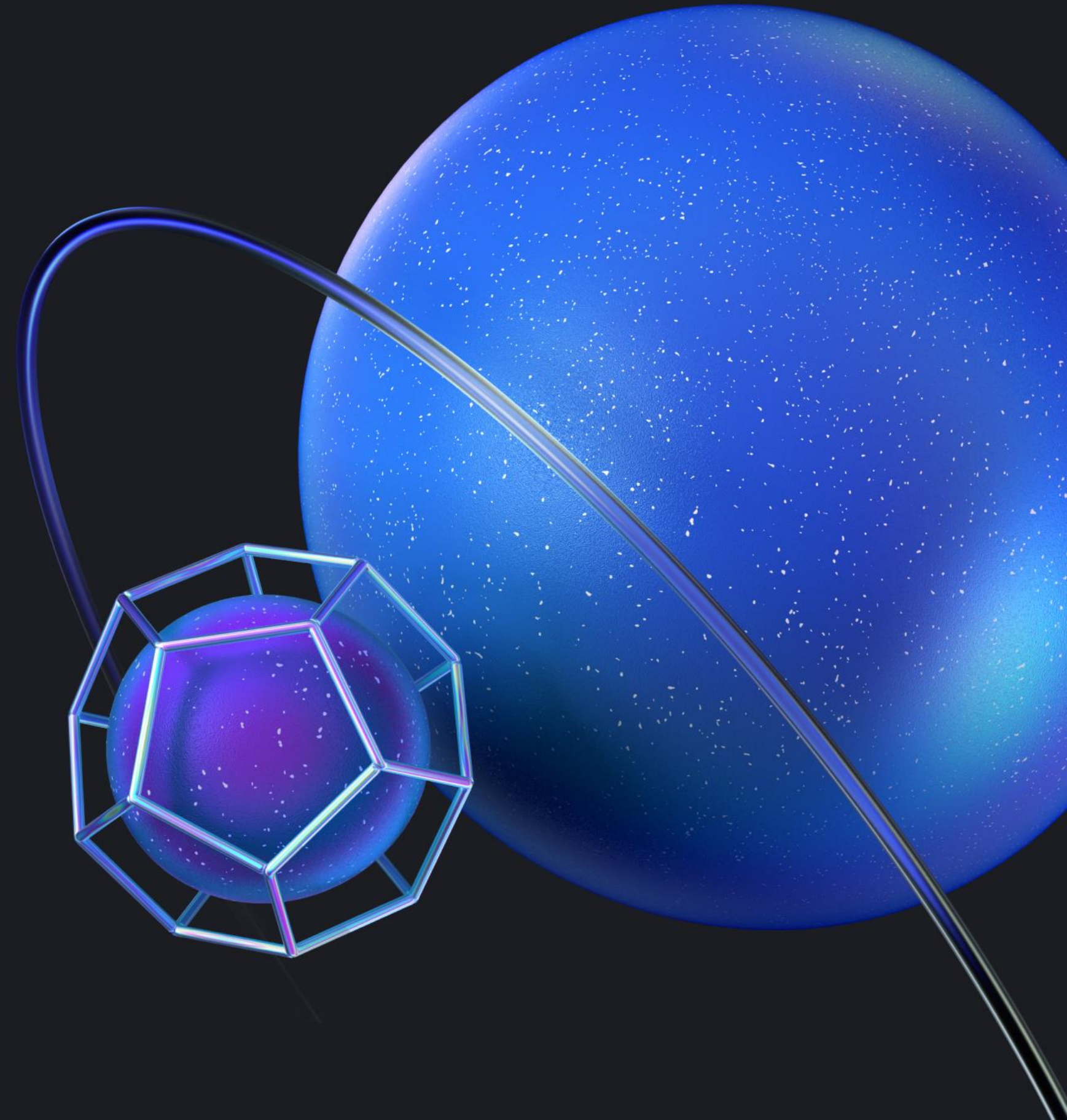
Саттар Гюльмамедов

РО команды ETL

МТС Тета

х

DataOps Platform



onETL урок # 7



- Объект манипуляции данными FileDownloader
- Опции конструктора
- Методы объекта
- FileDownloaderResult
- Методы, атрибуты и свойства FileDownloaderResult

Объекты манипуляции данными

→ FileDownloader

→ FileUploader

→ FileMover

→ FileFilters

→ FileLimits

FileDownloader



- FileDownloader()
- File Downloader Options
- File Downloader Result



MTC Tera

x

DataOps Platform

FileDownloader()



```
from onetl.file import FileDownloader
```

```
download = FileDownloader(  
    connection=connection,  
    source_path="/path/to/remote/source",  
    local_path="/path/to/local",  
)
```

```
download.run()
```

Python

Параметр	Объяснение
connection	Коннектор к файловой системе, где лежат исходные данные
local_path	Путь на локальной файловой системе, куда будут скачиваться файлы из источника
source_path	Путь на файловой системе-источнике откуда будут скачиваться файлы
temp_path	Путь для временного хранения файла, если необходимо
filter	Фильтр для отбора файлов и папок
limit	Ограничение на число скачиваемых файлов
hwm	Метка инкрементального процесса
options	Дополнительные параметры

FileDownloader Options



Опция	Объяснение	Значения или их тип
if_exists	Что делать, если в назначении есть такие же файлы	error, ignore, replace_file или replace_entire_directory
delete_source	Необходимость удаления файлов в источнике	True False
workers	Количество параллельных потоков обрабатывающих файлы	int



Методы filedownloader

→ view_files()

→ run()

.view_files()

```
from onetl.file import FileDownloader

download = FileDownloader(source_path="/remote", ... )

download.view_files()

...
[
    RemoteFile("/remote/file1.txt"),
    RemoteFile("/remote/file3.txt"),
    RemoteFile("/remote/nested/file3.txt"),
]
```

Python

.run()

```
from onetl.file import FileDownloader

download = FileDownloader(local_path="/local", ...)

downloaded_files = download.run(
    [
        "/remote/file1.txt",
        "/any/nested/path/file2.txt",
    ]
)

downloaded_files.successful = {
    LocalPath("/local/file1.txt"),
    LocalPath("/local/file2.txt")
}
```

Python

FileDownloader Result



→ Атрибуты и свойства

→ Методы



MTC Teta

x

DataOps Platform

FileDownloader Result

категории свойств



- общие
- failed
- missing
- skipped
- successful

FileDownloader Result

общие свойства



Свойство	Объяснение
details	Детальное описание результата
summary	Краткое описание результата
is_empty	Возвращает True, если successful, failed и skipped не содержат файлов
total_count	Общее число файлов в результате
total_size	Общий размер файлов в результате

МТ
С

FileDownloader Result свойства по статусам файлов

Свойство	Объяснение
failed	
failed_count	Количество файлов, которые не удалось скачать из-за ошибок
failed_size	Размер файлов, которые не удалось скачать из-за ошибок
missing	
missing_count	Количество отсутствовавших файлов
skipped	
skipped_count	Количество пропущенных файлов
skipped_size	Размер пропущенных файлов
successful	
successful_count	Количество скачанных файлов
successful_size	Размер скачанных файлов

Методы FileDownloader Result



Метод	Объяснение
dict()	Возвращает словарь, представляющий объект результат, с возможностью отбора полей
json()	Возвращает json, представляющий объект результат, с возможностью отбора полей
raise_if_contains_zero_size()	Инициирует исключение, если среди скачанных содержится файл с нулевым размером
raise_if_empty()	Инициирует исключение, если нет файлов ни в одном из атрибутов successful, failed или skipped
raise_if_failed()	Инициирует исключение если из-за ошибок не был скачан хотя бы один файл
raise_if_missing()	Инициирует исключение, если отсутствовал хотя бы один файл
raise_if_skipped()	Инициирует исключение, если был пропущен хотя бы один файл

Спасибо!



onetools@mts.ru
<https://t.me/c/1511728757/5>

MTC Тета

x

DataOps Platform

