

# PySpark

## Инициализация SparkContext

```
In from pyspark import SparkContext

# sc – от англ. spark context
# appName – от англ. application name, название приложения
sc = SparkContext(appName="IntroToSpark")
```

## Перевод списка в RDD

```
In pyspark_entry = sc.parallelize(['2009-01-01', 0, 0, 24])

# Извлечение содержимого RDD
print(pyspark_entry.take(n_elements))
```

## Создание объекта SparkSession

```
In from pyspark.sql import SparkSession

APP_NAME = 'sampleApp'

# builder – англ. конструктор сессии
spark = SparkSession.builder.appName(APP_NAME).getOrCreate()

# не отображать контекст при выполнении операций
spark = SparkSession.builder.appName(APP_NAME) \
    .config('spark.ui.showConsoleProgress', 'false').getOrCreate()
```

## Создание датафрейма PySpark

```
In from pyspark.sql import SparkSession

APP_NAME = "DataFrames"
SPARK_URL = "local[*]"

spark = SparkSession.builder.appName(APP_NAME).getOrCreate()

df = pd.read_csv('data.csv')
spark_df = spark.createDataFrame(df)
```

## Чтение csv-файла в PySpark Dataframe

```
In # format='csv' – укажите формат файла
# header='true' – укажите, что в файле есть заголовок (имена столбцов)
# inferSchema='true' – англ. выводить схему,
# укажите, что типы данных должны быть выведены
taxi = spark.read.load('data.csv', format='csv', header='true', inferSchema='true')
```

#### Отображение информации о столбцах датафрейма

In `print(spark_df.describe().show())`

#### Отображение подробной информации датафрейма

In `print(spark_df.summary().show())`

#### Регистрация временной таблицы

In `spark_df.registerTempTable("spark_df")`

#### Выполнение SQL-запроса с агрегатными функциями

In `print(spark_df.groupBy("column_name").mean().select("column_1", "column_2").show())`

#### Выполнение SQL-запроса с агрегатными функциями и сортировкой

```
print(spark_df.groupBy("column_name").mean().select("column_1", "column_2")) \
.sort("column_2", ascending=False).show())
```

#### Удаление пропущенных значений

In `spark_df = spark_df.dropna()`

#### Замена пропусков значением value

In `spark_df = spark_df.fillna(value)`

#### Выполнение SQL-запроса

In `# query - sql-запрос  
print(spark.sql(query).show())`

## Словарь

#### Распределённые системы

компьютерные системы, которые хранят файлы с данными на нескольких компьютерах и предоставляют доступ к ним. Файл делится на фрагменты, причём каждый фрагмент может быть сохранён несколько раз на разных компьютерах.

#### Узел (node)

отдельный компьютер с ресурсами вычисления и хранения данных

#### Кластер (cluster)

набор связанных узлов

#### Мастер-узел, или ведущий узел (Name Node)

узел, который распределяет файлы между компьютерами в кластере

#### Узлы данных (Data Nodes)

узлы, в которых содержатся и обрабатываются данные. Чтобы избежать потери информации, каждый файл дублируется в нескольких узлах данных.