**Bekerja dengan data di Azure Databricks**

Untuk bekerja dengan data di Azure Databricks, Anda dapat menggunakan objek dataframe.

**Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan dapat:

* Memahami kerangka data
* Kerangka data kueri
* Memvisualisasikan data

1. **Pengantar**

Di Azure Databricks, ilmuwan data menggunakan DataFrames untuk menyusun data mereka. **DataFrame (** <https://learn.microsoft.com/id-id/azure/databricks/getting-started/dataframes-python> **)** setara dengan tabel relasional di Spark SQL. Dalam modul ini, Anda akan mempelajari apa itu DataFrames dan bagaimana menggunakannya.

1. **Tujuan pembelajaran**

Setelah menyelesaikan modul ini, Anda akan mampu:

* Menjelaskan DataFrame.
* Kueri DataFrames.
* Visualisasikan data dengan Spark.

1. **Memahami kerangka data**

Spark menggunakan 3 API berbeda: RDD, DataFrames, dan DataSet. Fondasi arsitekturnya adalah dataset terdistribusi yang tangguh (RDD). DataFrame API dirilis sebagai abstraksi di atas RDD, diikuti kemudian oleh Dataset API. Kami hanya akan menggunakan DataFrames dalam contoh buku catatan kami.

DataFrames adalah kumpulan data yang terdistribusi, diatur ke dalam baris dan kolom. Setiap kolom dalam DataFrame memiliki nama dan tipe terkait.

Spark DataFrames dapat dibuat dari berbagai sumber, seperti file CSV, JSON, file Parket, tabel Apache Hive, tabel log, dan database eksternal.

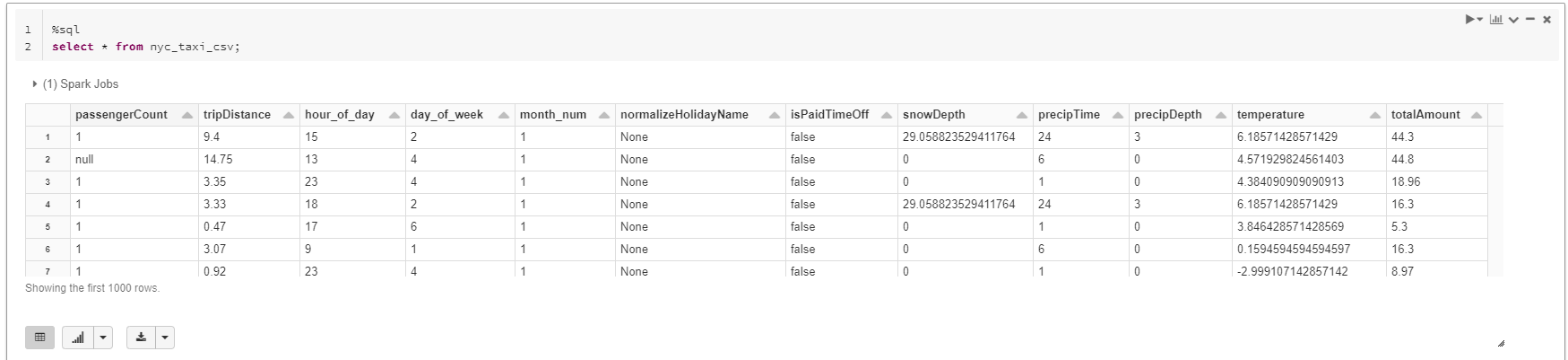
**Catatan**

**Informasi selengkapnya:** untuk informasi selengkapnya tentang struktur data Spark, lihat DataFrames dalam dokumentasi Azure Databricks.

**Link** : <https://learn.microsoft.com/id-id/azure/databricks/getting-started/dataframes-python>

1. **Menggunakan Spark untuk memuat data tabel**

Dengan asumsi kita memiliki data ini yang tersedia dalam tabel:



Kita dapat menggunakan Spark untuk memuat data tabel dengan menggunakan metode **sql:**



1. **Menggunakan Spark untuk memuat file/data DBFS**

Kami juga dapat membaca data dari file asli yang telah kami unggah; atau memang dari file lain yang tersedia di DBFS. Kodenya sama terlepas dari apakah file itu lokal atau di penyimpanan jarak jauh yang diinstal, berkat titik pemasangan DBFS.



Spark mendukung banyak format data yang berbeda, seperti CSV, JSON, XML, Parket, Avro, ORC, dan lainnya.

1. **Ukuran bingkai data**

Untuk mendapatkan jumlah baris yang tersedia dalam DataFrame, kita dapat menggunakan metode **count().**

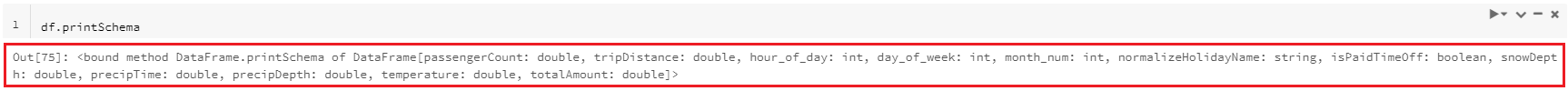


1. **Struktur DataFrame**

Untuk mendapatkan metadata skema untuk DataFrame yang diberikan, kita dapat menggunakan metode **printSchema().**

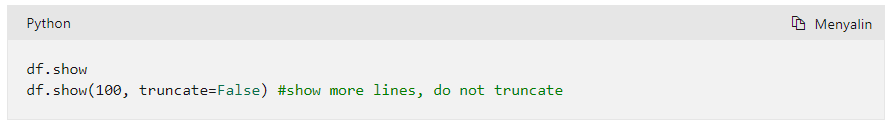
Setiap kolom dalam DataFrame yang diberikan memiliki nama, tipe, dan flag nullable.





1. **Isi DataFrame**

Spark memiliki fungsi bawaan yang memungkinkan untuk mencetak baris di dalam DataFrame: **show()**



Secara default itu hanya akan menampilkan 20 baris pertama di DataFrame Anda dan itu akan memotong kolom panjang. Parameter tambahan tersedia untuk mengganti pengaturan ini.

1. **Kerangka data kueri**

DataFrames memungkinkan pemrosesan data dalam jumlah besar. Spark menggunakan komputer pengoptimalan untuk menghasilkan kueri logis. Data didistribusikan ke kluster Anda dan Anda mendapatkan kinerja yang luar biasa untuk sejumlah besar data.

Spark SQL adalah komponen yang memperkenalkan DataFrames, yang menyediakan dukungan untuk data terstruktur dan semi-terstruktur.

Spark memiliki banyak antarmuka (API) untuk menangani DataFrames:

* Kami telah melihat metode **.sql()**, yang memungkinkan untuk menjalankan kueri SQL arbitrer pada data tabel.
* Opsi lainnya adalah menggunakan bahasa khusus domain Spark untuk manipulasi data terstruktur, tersedia di Scala, Java, Python, dan R.

1. **DataFrame API**

Apache Spark DataFrame API menyediakan serangkaian fungsi yang kaya (pilih kolom, filter, gabung, agregat, dan sebagainya) yang memungkinkan Anda memecahkan masalah analisis data umum secara efisien.

Operasi kompleks di mana tabel digabungkan, difilter, dan direstrukturisasi mudah ditulis, mudah dipahami, diketik dengan aman, terasa alami bagi orang-orang dengan pengalaman sql sebelumnya, dan dilengkapi dengan kecepatan tambahan pemrosesan paralel yang diberikan oleh komputer Spark.

Untuk memuat atau menyimpan data, gunakan **read** dan **write**:



Untuk mendapatkan data yang tersedia dalam DataFrame gunakan **select**:



Untuk mengekstrak baris pertama, gunakan **take**:



Untuk mengurutkan data, gunakan metode **sort**:



Untuk menggabungkan baris dalam beberapa DataFrames gunakan **union**:



Operasi ini setara dengan **UNION ALL** dalam SQL. Untuk melakukan penyatuan set gaya SQL (yang melakukan deduplikasi elemen), gunakan fungsi ini diikuti oleh **distinct().**

Kerangka data harus memiliki struktur/skema yang sama.

Untuk menambah atau memperbarui kolom, gunakan **withColumn** atau **withColumnRenamed**:



Untuk menggunakan **alias** untuk seluruh DataFrame atau kolom tertentu:



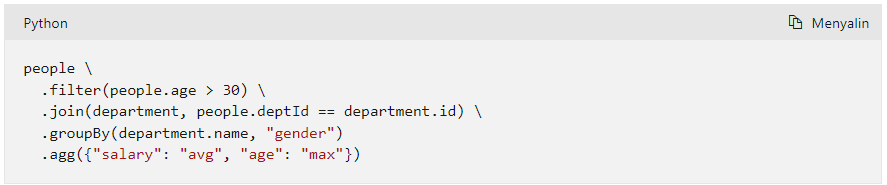
Untuk membuat tampilan sementara:



Untuk menggabungkan seluruh DataFrame tanpa grup, gunakan **agg**:



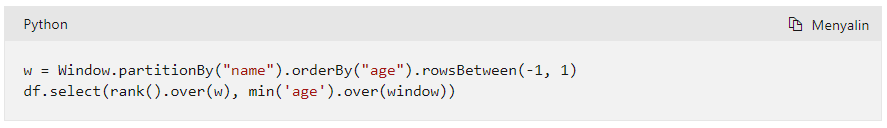
Untuk melakukan kueri yang lebih kompleks, gunakan **filter**, **groupBy**, dan **join**:



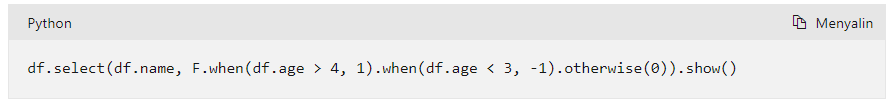
Jenis gabungan ini didukung: inner, cross, outer, full, full\_outer, left, left\_outer, right, right\_outer, left\_semi, dan left\_anti.

Perhatikan bahwa **filter** adalah alias untuk **where**.

Untuk menggunakan agregasi kolom menggunakan windows:



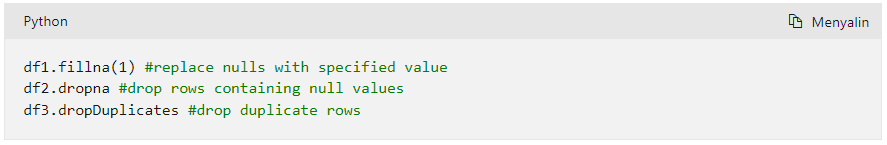
Untuk menggunakan daftar kondisi kolom dan mengembalikan ekspresi, gunakan **when**:



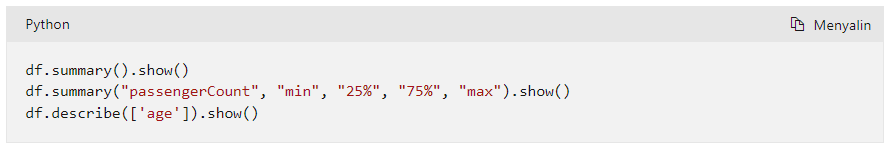
Untuk memeriksa keberadaan data, gunakan **isNull** atau **isNotNull**:



Untuk membersihkan data, gunakan **dropna**, **fillna** atau **dropDuplicates**:



Untuk mendapatkan statistik tentang DataFrame gunakan **summary** atau **describe**:



Statistik yang tersedia adalah:

* Menghitung
* Rerata
* Stddev
* Min
* Maks
* Persentil perkiraan arbitrer yang ditentukan sebagai persentase (misalnya, 75%).

Untuk menemukan korelasi antar kolom tertentu, gunakan **corr**. Operasi ini saat ini hanya mendukung Koefisien Korelasi Pearson:



**Catatan**

**Informasi selengkapnya:** untuk informasi selengkapnya tentang API Spark, lihat

**DataFrame API** : <https://spark.apache.org/docs/2.4.0/api/python/pyspark.sql.html#pyspark.sql.DataFrame?azure-portal=true>

dan

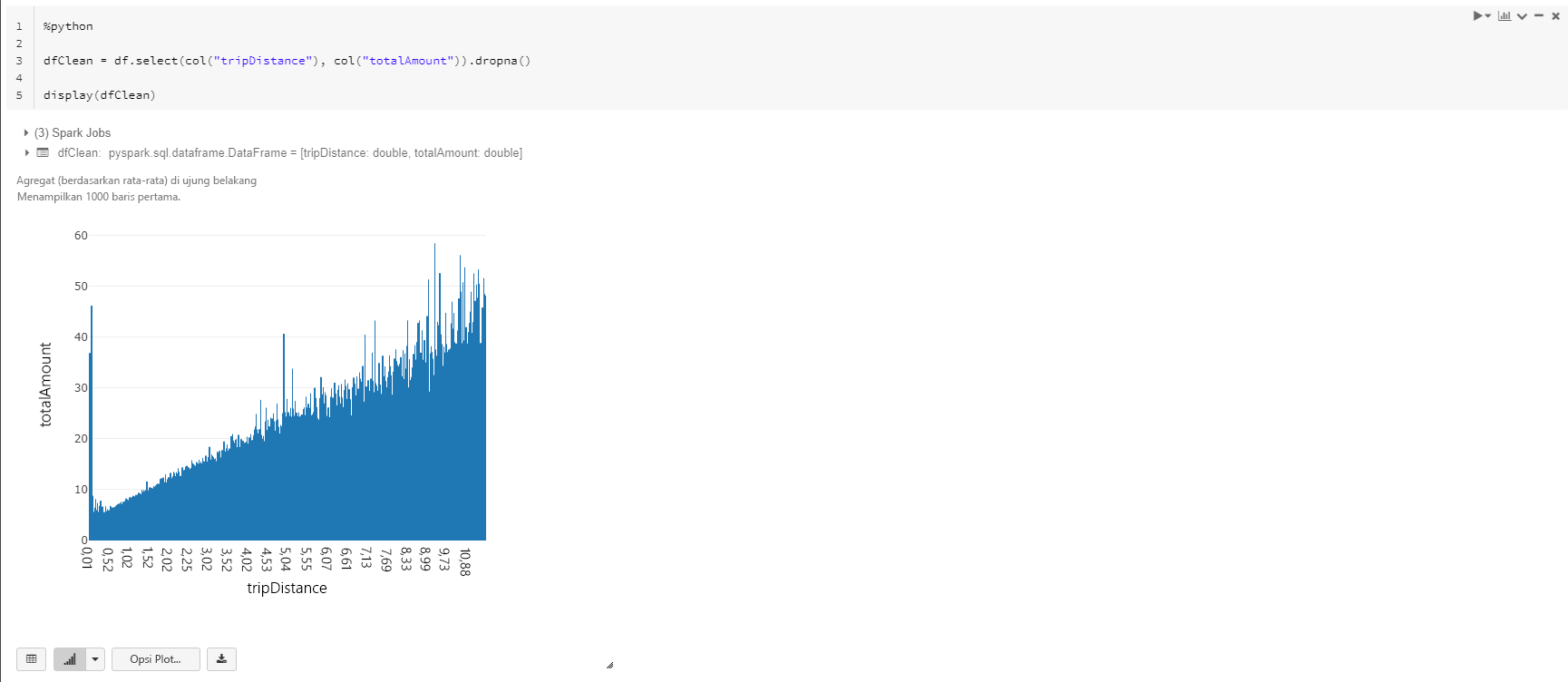
**API Kolom** dalam dokumentasi Spark. : <https://spark.apache.org/docs/2.4.0/api/python/pyspark.sql.html#pyspark.sql.Column?azure-portal=true>

1. **Memvisualisasikan data**

Spark memiliki fungsi **show** bawaan, yang memungkinkan untuk mencetak baris dalam DataFrame.

Azure Databricks menambahkan kemampuan tampilannya sendiri dan menambahkan berbagai jenis visualisasi lain yang siap pakai menggunakan fungsi **display** dan **displayHTML**.

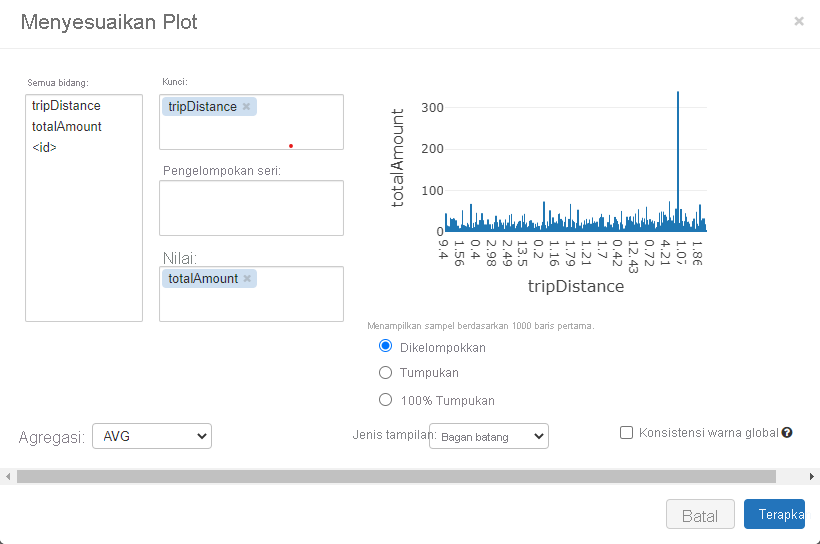
Data yang sama yang telah kita lihat sebelumnya sebagai tabel dapat ditampilkan sebagai diagram batang, pai, histogram, atau grafik lainnya. Bahkan peta atau gambar dapat ditampilkan:



1. **Opsi plot**

Opsi tampilan berikut tersedia:

* Kita dapat memilih kolom DataFrame untuk digunakan sebagai sumbu (kunci, nilai).
* Kita dapat memilih untuk mengelompokkan rangkaian data kita.
* Kita dapat memilih agregasi yang akan digunakan dengan data yang dikelompokkan (rata-rata, jumlah, hitungan, min, maks).



**Catatan**

**Informasi selengkapnya:** untuk informasi selengkapnya tentang visualisasi yang tersedia, lihat **Visualisasi** dalam dokumentasi Azure Databricks.

**Link** : <https://learn.microsoft.com/id-id/azure/databricks/notebooks/visualizations/>

1. **Latihan - Bekerja dengan data di Azure Databricks**

Sekarang adalah kesempatan Anda untuk merasakan sendiri Azure Databricks dengan memuat data, memanipulasinya, dan memvisualisasikan hasilnya.

Dalam latihan ini, Anda akan:

* Muat data ke dalam DataFrame.
* Kueri kerangka data.
* Transformasi data menggunakan DataFrames.
* Visualisasikan data.

1. **Instruksi**

Ikuti petunjuk berikut untuk menyelesaikan latihan:

1. Buka petunjuk latihan di <https://aka.ms/mslearn-dp090.> \ <https://microsoftlearning.github.io/dp-090-databricks-ml/>
2. Selesaikan latihan **Bekerja dengan Data di Azure Databricks.**
3. **Ringkasan**

Dalam modul ini, Anda telah mempelajari cara bekerja dengan data di Azure Databricks.

Sekarang setelah Anda menyelesaikan modul ini, Anda dapat:

* Menjelaskan DataFrame.
* Kueri DataFrames.
* Visualisasikan data dengan Spark.