Progetto di Analisi dei dati con SQL

Luigi Mennella

Descrizione del progetto

Scopo: Dato un laboratorio, analizzare per ogni operatore come è variato il valore ottenuto degli esperimenti, prima e dopo la data di cambio di macchinario (1 maggio 2020).

Dati di partenza*: Sono interessate dal macchinario le molecole:

- Il cui nome inizia AB e finisce con D;
- Il cui nome inizia con F e NON finisce per P.
- I dati sono disponibili nei seguenti file: *Esperimenti_1.csv*, *Esperimenti_2.csv*.

Analisi dei file di input

L'input è rappresentato da due file **csv** con :

- Delimitatore *punto e virgola (;)*
- Presenza di intestazione
- Presenza di 5 colonne
- Data in formato DD/MM/AAAA
- Separatore decimale *virgola (,)*.

```
IdEsperimento;Data;Operatore;Valore;Molecola
113;22/04/2020;1;2,622719463;FFDAP
114;23/04/2020;1;1,852159855;BAPEF
115;24/04/2020;1;7,38344943;ABRID
116;25/04/2020;1;2,138829897;ABRID
117;26/04/2020;1;4,659331211;ABCCD
118;27/04/2020;1;0,441903197;TBWA
119;28/04/2020;1;0,889873355;ACBBE
120;29/04/2020;1;2,182529694;ABCDE
121;30/04/2020;1;1,203937297;FFDAP
122;01/05/2020;1;0,800957324;BAPEF
123;02/05/2020;1;1,957619777;ABRID
124;03/05/2020;1;5,942261361;ABRID
125;04/05/2020;1;0,967289284;FFDAG
126;05/05/2020;2;1,940118029;FFDAG
127;06/05/2020;2;4,865734562;ABCCD
```

Importazione dei dati

Carichiamo inizialmente i dati in una **tabella di staging** senza vincoli.

```
CREATE TABLE dbo.StagingEsperimento(
    IdEsperimento varchar(255),
    Data  varchar(255),
    Operatore varchar(255),
    Valore varchar(255),
    Molecola varchar(255),
    )
GO
```

```
BULK INSERT dbo.StagingEsperimento
FROM '...\Progetto_Esperimenti.csv'
WITH
(
    FIRSTROW = 2,
    FIELDTERMINATOR = ';',
    ROWTERMINATOR = '\n',
    TABLOCK
)
```

Va ricordato che la procedura di importazione dati viene eseguita due volte, una per ognuno dei file da importare.

Importazione dei dati - 2

Successivamente trasferiamo i dati nella **tabella target**, con relativa ***chiave primaria*** e ***vincoli non nullità.***

```
CREATE TABLE dbo.Esperimento(
    IdEsperimento INT PRIMARY KEY NOT NULL
    Data Date NOT NULL,
    Operatore varchar(255) NOT NULL,
    Valore decimal (18,10) NOT NULL,
    Molecola varchar(255) NOT NULL);
```

```
INSERT INTO dbo.Esperimento
    (IdEsperimento, Data, Operatore,
     Valore, Molecola)
SELECT CAST(IdEsperimento AS INT) AS IdEsperimento,
    CAST(CONCAT(RIGHT(Data,4),'-',substring(Data,4,2),'-',LEFT(Data,2)) AS DATE) AS Data,
    Operatore,
    CAST(REPLACE(Valore,',','.') as DECIMAL(18,10)) AS Valore,
    Molecola
FROM dbo.StagingEsperimento;

SELECT * FROM DBO.Esperimento
```

Per modificare i formati del campi data e valore, vengono usati i comandi: **CAST**, **CONCAT**, **REPLACE** e **SUBRSTRING**.

Scrittura della query in SQL

Dopo aver studiato diverse alternative, optiamo per una query, con:

- **CASE WHEN** per differenziare la media in funzione della data
- **WHERE** per selezionare le molecole
- Differenza tra le medie (**Diff**)
- Uso di una **CTE** per una migliore leggibilità

```
WITH FILTRO AS (
        SELECT Operatore,
               CONVERT(DECIMAL (18,2),AVG(CASE WHEN Data < '20200501'
               THEN Valore ELSE NULL END))  AS MediaP,
               CONVERT(DECIMAL (18,2),AVG(CASE WHEN Data >= '20200501'
               THEN Valore ELSE NULL END))  AS MediaD
            FROM dbo.Esperimento
            WHERE LEFT(Molecola,2) = 'AB' AND RIGHT(Molecola,1) = 'D'
            OR LEFT(Molecola,1) = 'F' AND RIGHT(Molecola,1) <> 'P'
        GROUP BY Operatore)
SELECT Operatore,
        MediaP,
        MediaD,
        MediaD-MediaP as Diff,
        CONVERT(DECIMAL (18,2),CASE WHEN MediaP=0 THEN NULL
        ELSE (MediaD-MediaP)/MediaP*100 END) as DiffPer
FROM FILTRO
```

Analisi dei risultati

I risultati mostrano una variazione importante delle misurazioni per tutti gli operatori, con un valore massimo per l'operatore n.1 e minimo per i n.2.

	Operatore	MediaP	MediaD	Diff
1	1	2.11	3.09	0.98
2	2	2.71	2.99	0.28
3	3	2.03	2.89	0.86
4	Totale	2.24	2.97	0.73

Si evidenzia inoltre che per ottenere la tabella di riepilogo dei risultati è stato necessario utilizzare le seguenti funzioni:

- **Tabelle temporanee** per stimare separatamente per singolo operatore e quello totale
- **UNION ALL per unire unire i risultati delle tabelle temporanee**
- **ORDER BY** per garantire l'ordine corretto.