

در این بخش فانکشن دیگری از توابع تحلیلی به نام CUME_DIST را بررسی می‌نماییم.

CUME_DIST

بوسیله تابع CUME_DIST می‌توان ارزیابی نمود، در یک گروه، چه درصد از مقادیر، مساوی یا کوچکتر از مقدار سطر جاری می‌باشند، به این تابع cumulative distribution نیز گفته می‌شود.

Syntax تابع CUME_DIST به صورت زیر است:

```
CUME_DIST( )
OVER ( [ partition_by_clause ] order_by_clause )
```

شرح Syntax:

1- Partition By Clause : بوسیله پارامتر فوق می‌توانید، نتیجه پرس جو (Query)، خود را دسته بندی نمایید.

2- order by clause : همانطور که از نامش مشخص است، جهت مرتب نمودن خروجی Query می‌باشد.

معمولا شرح عملکرد توابع تحلیلی، کمی مشکل است. بنابراین برای درک، عملکرد تابع CUME_DIST چند مثال را بررسی می‌کنیم.

در ابتدا بوسیله Script زیر یک جدول ایجاد و 10 رکورد در آن درج می‌کنیم:

```
Create Table TestCUME_DIST
(SalesOrderID int not null,
OrderQty smallint not null,
ProductID int not null
);
GO
Insert Into TestCUME_DIST
Values (43663,1,760),(43667,3,710),(43667,1,773),
(43667,1,775),(43667,1,778),(43669,1,747),
(43670,1,709),(43670,2,710),(43670,2,773),(43670,1,776)
```

مثال اول: Script زیر را اجرا می‌کنیم، سپس خروجی آن را بررسی می‌نماییم:

```
SELECT SalesOrderID, OrderQty,
       CUME_DIST() OVER(ORDER BY SalesOrderID) AS [CUME_DIST]
FROM TestCUME_DIST ORDER BY [CUME_DIST] DESC
```

پس از اجرا خروجی بصورت زیر خواهد بود:

	SalesOrderID	OrderQty	CUME_DIST
1	43670	1	1
2	43670	2	1
3	43670	2	1
4	43670	1	1
5	43669	1	0.6
6	43667	3	0.5
7	43667	1	0.5
8	43667	1	0.5
9	43667	1	0.5
10	43663	1	0.1

در ادامه اجازه دهید، مقادیری که در فیلد CUME_DIST بدست آمده است را بصورت تصویری بررسی کنیم.
مقادیر سطر اول تا چهارم:

	SalesOrderID	OrderQty	CUME_DIST
1	43670	1	1
2	43670	2	1
3	43670	2	1
4	43670	1	1
5	43669	1	0.6
6	43667	3	0.5
7	43667	1	0.5
8	43667	1	0.5
9	43667	1	0.5
10	43663	1	0.1

سوال اول: چند سطر، SalesOrderID آن برابر ۴۳۶۷۰ می باشد؟

جواب: چهار سطر (Row)، فرض می کنیم $C1=4$

سوال دوم: چند سطر (Row)، زیر SalesOrderID=43670 قرار دارد؟

جواب: شش سطر (Row)، فرض می کنیم $C2=6$

سوال سوم: تعداد کل سطرها (Rows)، چند عدد می باشد؟

جواب: ۱۰ سطر (Rows)، فرض می کنیم $C3=10$

برای بدست آوردن CUME_DIST چهار سطر اول از فرمول زیر پیروی می نماییم:

$$\text{Rows} = (C1+C2)/C3 = (4+6)/10 = 1$$

*** برای بدست آوردن CUME_DIST سطر پنجم نیز خواهیم داشت:

$$\text{Rows} = (c1+c2)/c3 \text{ بنابراین خواهیم داشت: } \text{Rows} = (1+5)/10 = 6/10$$

مثال دوم : ابتدا Script زیر را اجرا نمایید:

```
SELECT SalesOrderID, OrderQty, ProductID,
       CUME_DIST() OVER(PARTITION BY SalesOrderID ORDER BY ProductID ) AS [CUME_DIST]
FROM TestCUME_DIST
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY SalesOrderID DESC, [CUME_DIST] DESC
```

خروجی :

	SalesOrderID	OrderQty	ProductID	CUME_DIST
1	43670	1	776	1
2	43670	2	773	0.75
3	43670	2	710	0.5
4	43670	1	709	0.25
5	43669	1	747	1
6	43667	1	778	1
7	43667	1	775	0.75
8	43667	1	773	0.5
9	43667	3	710	0.25
10	43663	1	760	1

همانگونه که ملاحظه می‌کنید، در این مثال، خروجی، براساس SalesOrderID به چهار گروه تقسیم می‌شود و عملیات مرتب سازی روی فیلد ProductID انجام می‌گیرد، بنابراین CUME_DIST، روی هر گروه بر روی فیلد ProductID محاسبه می‌شود.

گروه اول : نحوه محاسبه Cume_DIST سطر اول:

سوال: چه تعداد از مقادیر ProductID آن برابر 776 می‌باشد؟

جواب: فقط مقدار سطر اول، بنابراین خواهیم داشت $C1=1$

سوال: چه تعداد از مقادیر کوچکتر از ProductID=776 می‌باشد؟

جواب: مقدار سه سطر، در واقع مقادیر سطر دوم، سوم و چهارم کوچکتر از مقدار سطر اول می‌باشند، $C2=3$

سوال: تعداد کل سطرهای گروه اول چه مقدار می‌باشد؟

جواب: 4 سطر

بنابراین برای بدست آوردن CUME_DIST سطر اول خواهیم داشت:

$$\text{Rows} = (1+3)/4 = 1$$

محاسبه سطر دوم از گروه اول بدون شرح:

$$\text{Rows} = (1+2)/4 = 0.75$$

امیدوارم مفید واقع شده باشد.