SQL Date & Time.

Agenda: Different components of Date & Time.
:

CTE.

Extract:
Given a date, time.
- Day, Month, Year.
- Hr, Min, Sec.
- Week, Quarter.

Day	Market opened	with 30mins.
1	8.AM.	8:30AM.
2	9.40 AM.	10:10 AM.
3	4:PM.	4:430PM.

All trans.
between this
time

String
| 8:00 AM. | → Datetime.

↑ hour   ↑ min   ↑ AM(PM).

Where is hr
Where is min
Where is sec.
if it is 24 hr/ 12 hr.
↓
AM/PM

Customer Purchases

Datetime — Info

DT DT

S.T   ET

↓
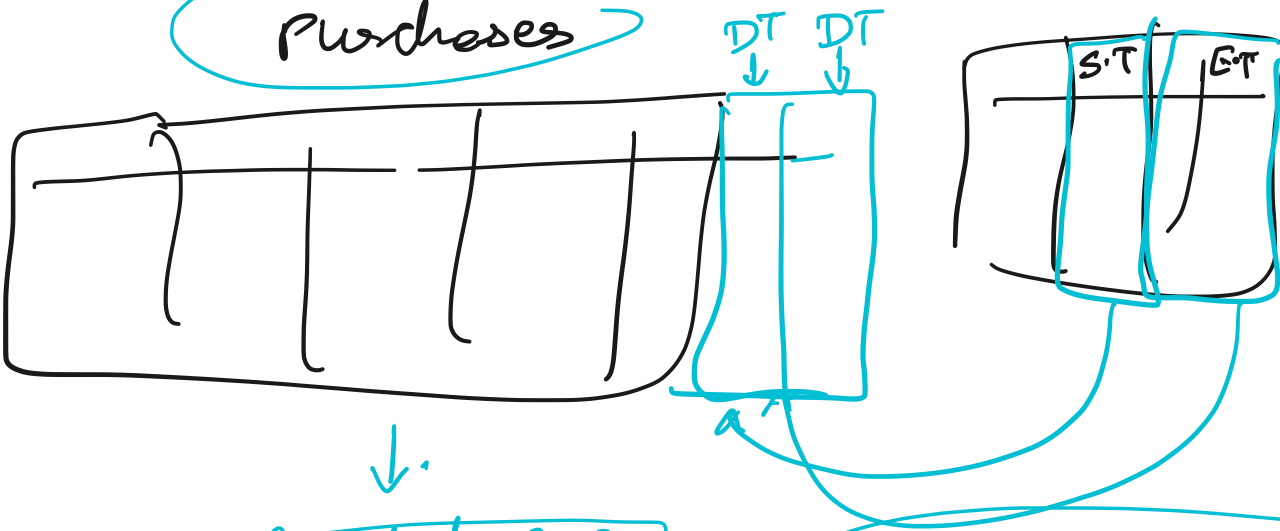
Created as a separate table. → customer-purchases-date

Cus.Id	Purchase date	
1	Jan 1	2 days
	Jan 3	7 days
	Jan 10	19
	Jan 29	2.
	Jan 31	
2	Feb 1	2
	Feb 3	1 day.
	Feb 4.	

Jan
Feb.
Marl
⋮
May 1
⋮
May 15
⋮
May 31
2019.

>31
>31
31 days.
15
cur date
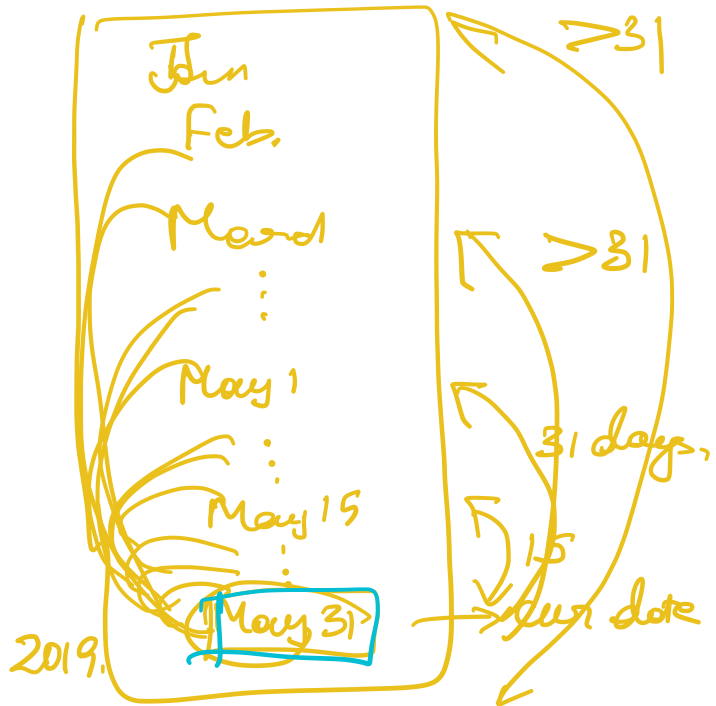
May 3	~	May 30	≠ 1
31	~	May 20	= 11
May 3	~	May 1	= 30
May 31	~	Apr 15	= 45.

✗

CTE ⟹ common table expression.

• CTE are useful to breakdown complex queries

into chunks of simpler queries.

- CTEs helps to reuse the queries in the subsequent queries.
- CTEs are useful and helps in easy readability when there are multiple joins, multiple groupby which otherwise will look complicated with many sub-queries.
- The temp output of each query in CTEs are deleted once the query execution completes, unlike using temp tables were the data is retained for the entire session.

Syntax:

```
With  <query_name1> as ( Your query 1 ),

        <query_name 2> as (your query 2 i.e.
                            Select * from query_name1
                            where ..... )
        <query_name3> as (your query 3 i.e.
                            Select col1, col 2....
                            for query_name2)
                  ⋮
            <query_name10> as (your query 10 )


    Select * from query_name 10.
```

- When using CTEs, the queries have access to the previous queries but not queries after it
- i.e. query 2 can access query_1, but not query 3....10. Similarly query 6 can access query 1, 2, 3, 4, 5 but not 7, 8, 9, 10.