ASSIGNMENT
DB BUILT IN SQL FUNCTION

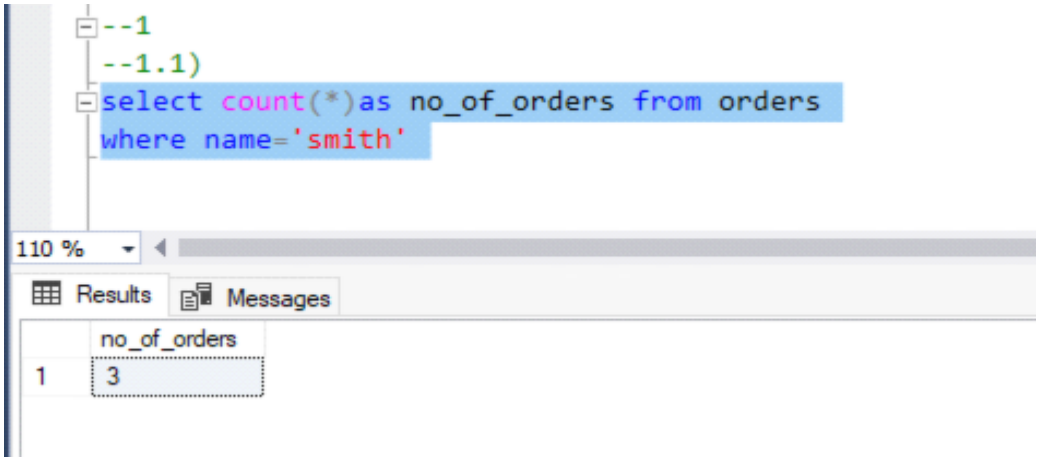
QUESTIONS

ORDER TABLE

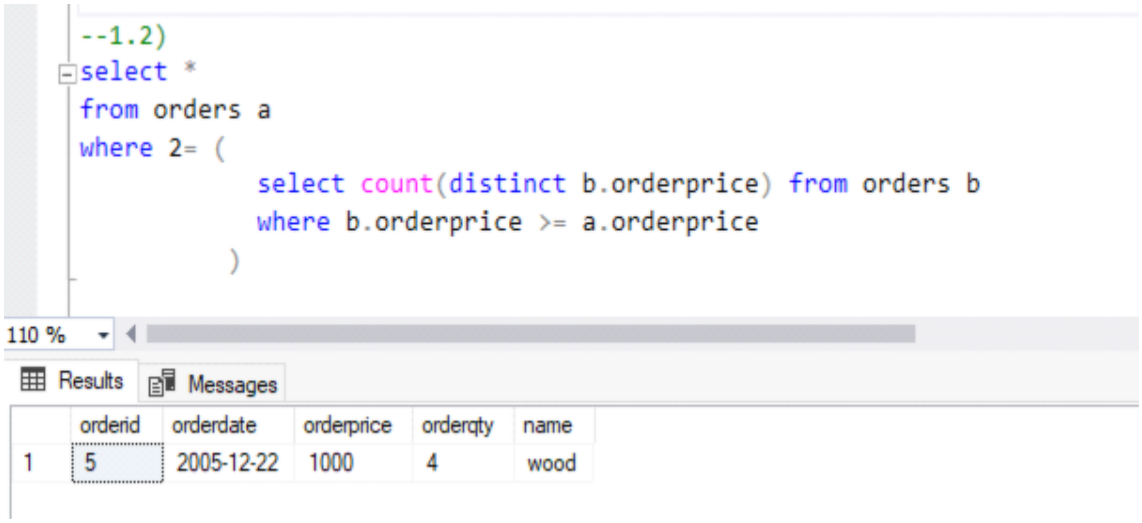
```
--CREATE TABLE
create table orders
(
orderid int identity(1,1) primary key ,
orderdate date ,
orderprice int,
orderqty int,
name varchar(10)
)

--INSERT INTO TABLE
insert into orders values
('2005/12/22',160,2,'smith'),
('2005/08/10',190,2,'johnson'),
('2005/07/13',500,5,'baldwin'),
('2005/07/15',420,2,'smith'),
('2005/12/22',1000,4,'wood'),
('2005/10/02',820,4,'smith'),
('2005/11/03',2000,2,'baldwin')

select * from orders
```

	orderid	orderdate	orderprice	orderqty	name
1	1	2005-12-22	160	2	smith
2	2	2005-08-10	190	2	johnson
3	3	2005-07-13	500	5	baldwin
4	4	2005-07-15	420	2	smith
5	5	2005-12-22	1000	4	wood
6	6	2005-10-02	820	4	smith
7	7	2005-11-03	2000	2	baldwin

1. **Count the number of orders made by a customer with CustName Smith**



2. **What is the 2nd largest order price**



3. **What if we are interested in the total value of all the orders?**

4. **what is the average number of items per order?**

```
--1.4)
select orderid,avg(orderqty) as average_per_order
from orders
group by orderid
```

110 %

Results | Messages

	orderid	average_per_order
1	1	2
2	2	2
3	3	5
4	4	2
5	5	4
6	6	4
7	7	2

5. **what if I need average order price when the price is more than 200.**

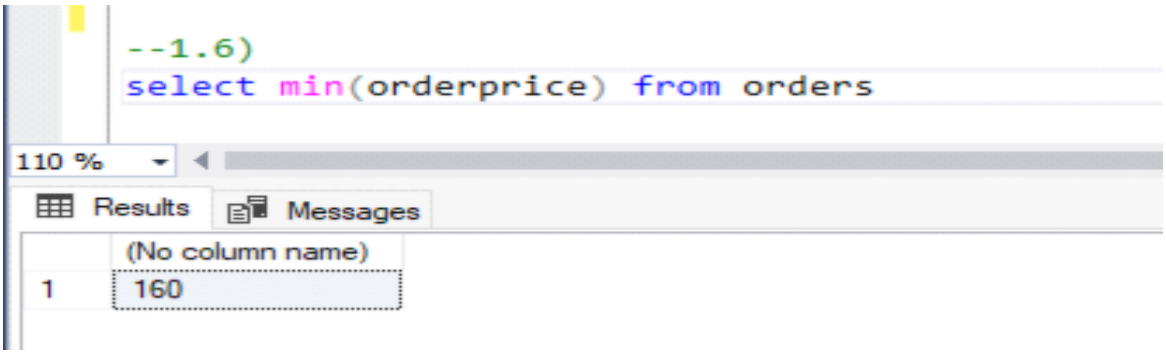
```
--1.5)
select avg(orderprice) as average_price
from orders
where orderprice >200
```

110 %

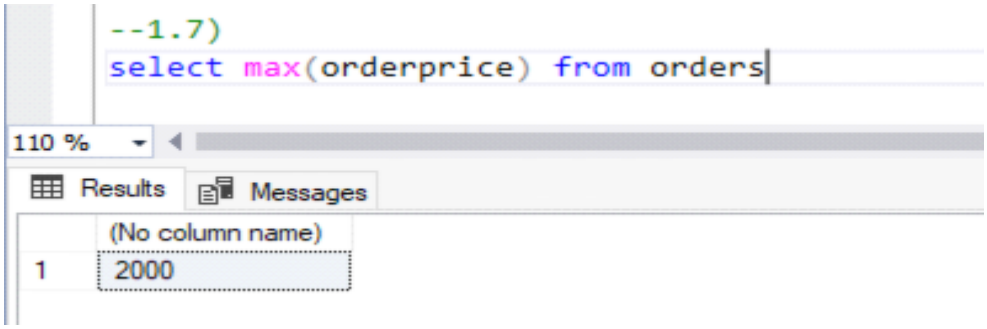
Results | Messages

	average_price
1	948

6. **minimum price paid for any of the orders.**



7. **maximum price paid for any of the orders**



EMPLOYEE TABLE

```
--2)
create table emp
(
id int identity(1,1) primary key,
name  varchar(10) not null  ,
joining_date date,
leaving_date date
)

insert into emp values
('vishal','2024/02/02',NULL),
('sahil','2019/03/01','2023/11/01'),
('shrey','2018/01/02','2023/11/01'),
('geetansh','2020/11/01','2022/11/01'),
('saksham','2020/02/01','2021/11/01')

select * from emp
```

110 %

Results | Messages

	id	name	joining_date	leaving_date
1	1	VIshaL	2024-02-02	NULL
2	2	SAhiL	2019-03-01	2023-11-01
3	3	SHreY	2018-01-02	2023-11-01
4	4	GEetansH	2020-11-01	2022-11-01
5	5	SAkshaM	2020-02-01	2021-11-01

1. **write a sql query to modify the emp_name in 'employee' table as the first 2 letters should be in upper case and last 1'letter in upper case and rest of the letters in lower case. eg emp_name is Gulrez then query should return the result as GUlreZ.**

```
--2)
UPDATE EMP
SET NAME =CONCAT (UPPER(LEFT(name,2)),LOWER(SUBSTRING ( NAME, 3, LEN(NAME)-3 )),UPPER(RIGHT(name,1)))
from emp


--3.1)
```

110 %

Results | Messages

	id	name	joining_date	leaving_date
1	1	VIshaL	2024-02-02	NULL
2	2	SAhiL	2019-03-01	2023-11-01
3	3	SHreY	2018-01-02	2023-11-01
4	4	GEetansH	2020-11-01	2022-11-01
5	5	SAkshaM	2020-02-01	2021-11-01

2. **List all the employees from 'employee' table those have left the 3. organization and find the number time spent in the company in the format x Years, x Months, xDays. eg, if the joining date of a employee is 31/1/2017 and leaving date if 15/03/2019 then the query result should be 1year, 1month and 16 days.**

```
--3.1)
SELECT * FROM emp
WHERE leaving_date is not NULL
```

110 %

Results | Messages

	id	name	joining_date	leaving_date
1	2	SAhiL	2019-03-01	2023-11-01
2	3	SHreY	2018-01-02	2023-11-01
3	4	GEetansH	2020-11-01	2022-11-01
4	5	SAkshaM	2020-02-01	2021-11-01

```sql
--3.2)
SELECT name ,
(CASE
    WHEN leaving_date IS NULL  THEN (
    CONCAT (
            FLOOR(DATEDIFF(YEAR,CAST(joining_date AS DATE),CAST(GETDATE() AS DATE))), ' years ' ,
            DATEDIFF(MONTH,CAST(joining_date AS DATE),CAST(GETDATE() AS DATE))%12, ' months ',
            FLOOR((DATEDIFF(DAY,CAST(joining_date AS DATE),CAST(GETDATE() AS DATE))%365.25)%12), ' days '
            ) )
    ELSE  (
    CONCAT (
            FLOOR(DATEDIFF(YEAR,CAST(joining_date AS DATE),CAST(leaving_date AS DATE))), ' years ' ,
            DATEDIFF(MONTH,CAST(joining_date AS DATE),CAST(leaving_date AS DATE))%12, ' months ',
            FLOOR((DATEDIFF(DAY,CAST(joining_date AS DATE),CAST(leaving_date AS DATE))%365.25)%12), ' days '
            ) )
END  ) AS duration
FROM emp

--4)
SELECT * ,
```

110 %

Results | Messages

	name	duration
1	VIshaL	0 years 0 months 8 days
2	SAhiL	4 years 8 months 5 days
3	SHreY	5 years 10 months 2 days
4	GEetansH	2 years 0 months 4 days
5	SAkshaM	1 years 9 months 9 days

3. Find the numbers of days spent by a employee in the company with current date.

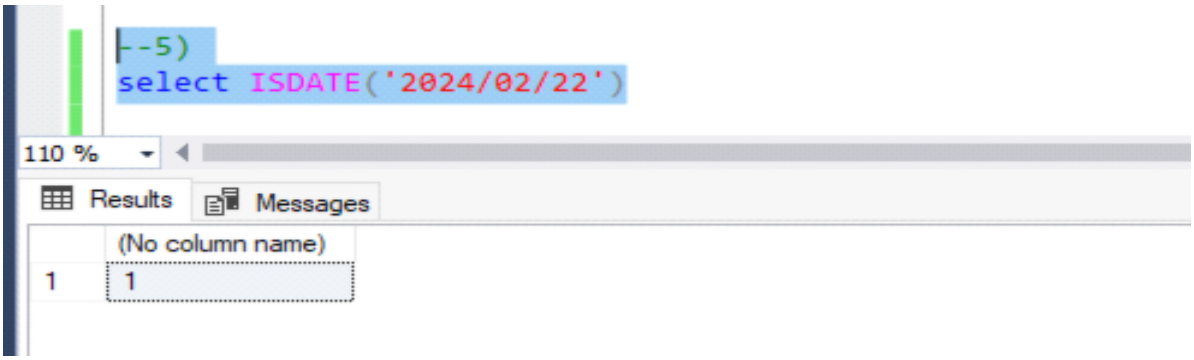
```sql
--4)
SELECT * ,
(CASE
    WHEN leaving_date IS NULL  THEN (DATEDIFF(DAY,CAST(joining_date AS DATE),CAST(GETDATE() AS DATE)) )
    ELSE  (DATEDIFF(DAY,CAST(joining_date AS DATE),CAST(leaving_date AS DATE)))
END) AS duration
FROM EMP
```

110 %

Results | Messages

	id	name	joining_date	leaving_date	duration
1	1	VIshaL	2024-02-02	NULL	20
2	2	SAhiL	2019-03-01	2023-11-01	1706
3	3	SHreY	2018-01-02	2023-11-01	2129
4	4	GEetansH	2020-11-01	2022-11-01	730
5	5	SAkshaM	2020-02-01	2021-11-01	639

4. **Which function checks whether the expression is a valid date or not?**



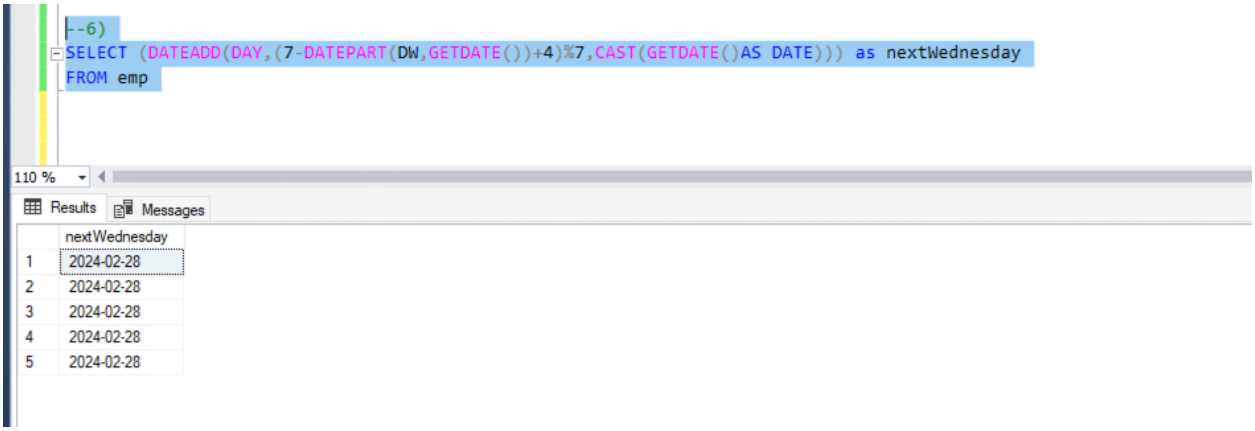
```
--5)
select ISDATE('2024/02/22')
```

110 %

Results | Messages

	(No column name)
1	1

5. **Find next Wednesday for the given date?**



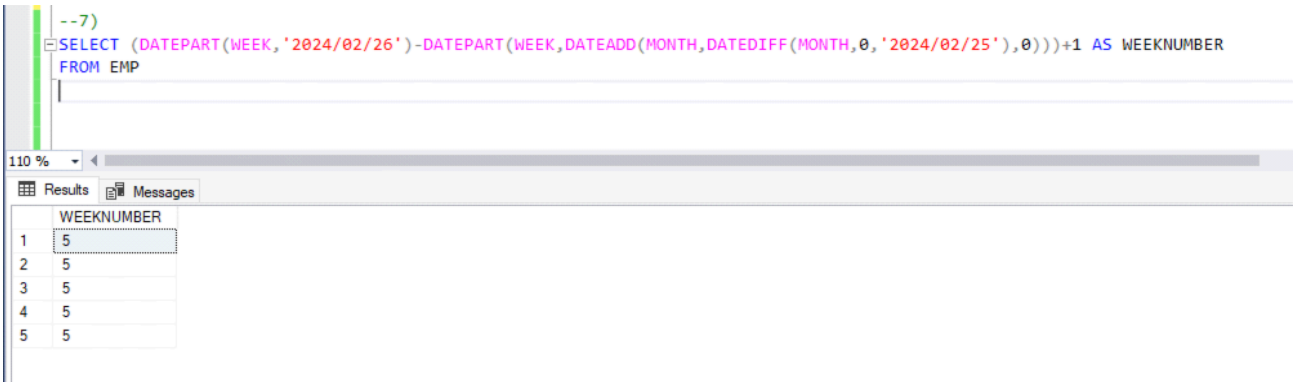
```
--6)
SELECT (DATEADD(DAY,(7-DATEPART(DW,GETDATE())+4)%7,CAST(GETDATE()AS DATE))) as nextWednesday
FROM emp
```

110 %

Results | Messages

	nextWednesday
1	2024-02-28
2	2024-02-28
3	2024-02-28
4	2024-02-28
5	2024-02-28

6. **Find week number of the month from the given date?**



```
--7)
SELECT (DATEPART(WEEK,'2024/02/26')-DATEPART(WEEK,DATEADD(MONTH,DATEDIFF(MONTH,0,'2024/02/25'),0)))+1 AS WEEKNUMBER
FROM EMP
```

110 %

Results | Messages

	WEEKNUMBER
1	5
2	5
3	5
4	5
5	5

7. Find the week's Monday date from the given date?

```
--8)
SELECT DATEADD(DAY ,2-DATEPART(DW,GETDATE()),CAST(GETDATE() AS DATE))
FROM EMP
```

110 %

Results | Messages

	(No column name)
1	2024-02-19
2	2024-02-19
3	2024-02-19
4	2024-02-19
5	2024-02-19