

AVERAGE, MAXIMUM, MINIMUM, JOINS

(1) With the example relations of your choice, demonstrate the following Aggregate Functions:

- Average: avg
- Minimum: Min
- Maximum: Max
- Total: sum
- Count: count

INPUT:-

```
CREATE TABLE Details (  
    Person_code VARCHAR(50),  
    Name VARCHAR(50),  
    Age INT,  
    Address VARCHAR(50),  
    Occupation VARCHAR(50),  
    Salary INT,  
    Family_Member INT  
);  
INSERT INTO Details VALUES ('P01', 'RIA', '21', 'DELHI', 'STUDENT', '00', '4')  
INSERT INTO Details VALUES ('P02', 'RAMA', '43', 'NOIDA', 'TEACHER', '25000', '5')  
INSERT INTO Details VALUES ('P03', 'RAJESH', '54', 'GHAZIABAD', 'BUSINESS', '50000', '3')  
INSERT INTO Details VALUES ('P04', 'SEEMA', '44', 'JAMMU', 'SALES', '20000', '4')  
INSERT INTO Details VALUES ('P05', 'NEEMA', '46', 'HARYANA', 'HR', '45000', '3')  
INSERT INTO Details VALUES ('P06', 'ASHA', '57', 'PUNJAB', 'TELECALLER', '20000', '3')  
INSERT INTO Details VALUES ('P07', 'VIVEK', '40', 'INDRAPURAM', 'DIRECTOR', '50000', '6')  
INSERT INTO Details VALUES ('P08', 'SAMARTH', '17', 'VAISHALI', 'STUDENT', '00', '3')  
INSERT INTO Details VALUES ('P09', 'NEHA', '25', 'USA', 'HOUSEWIFE', '00', '4')  
INSERT INTO Details VALUES ('P10', 'AJITESH', '24', 'KUNDALI', 'BUSINESS', '55000', '3')  
SELECT * FROM Details  
  
SELECT AVG(Age) AS AverageAge FROM Details;  
SELECT MIN(Salary) AS MinimumSalary FROM Details;  
SELECT MAX(Family_Member) AS MaximumMember FROM Details;  
SELECT SUM(Salary) AS TotalSalary FROM Details;  
SELECT COUNT(Person_code) AS NumberOfPerson FROM Details;
```

OUTPUT:-

	Person_code	Name	Age	Address	Occupation	Salary	Family_Member
1	P01	RIA	21	DELHI	STUDENT	0	4
2	P02	RAMA	43	NOIDA	TEACHER	25000	5
3	P03	RAJESH	54	GHAZIABAD	BUSINESS	50000	3
4	P04	SEEMA	44	JAMMU	SALES	20000	4
5	P05	NEEMA	46	HARYANA	HR	45000	3
6	P06	ASHA	57	PUNJAB	TELECALLER	20000	3
7	P07	VIVEK	40	INDRAPURAM	DIRECTOR	50000	6
8	P08	SAMARTH	17	VAISHALI	STUDENT	0	3
9	P09	NEHA	25	USA	HOUSEWIFE	0	4
10	P10	AJITESH	24	KUNDALI	BUSINESS	55000	3

	AverageAge
1	37

	MinimunSalary
1	0

	MaximumMember
1	6

	TotalSalary
1	265000

	NumberOfPerson
1	10

(2) With an example scenario of your choice, demonstrate the following Join expressions:

LEFT JOIN: This join returns all the rows of the table on the left side of the join and matching rows for the table on the right side of join. The rows for which there is no matching row on right side, the result-set will contain *null*. LEFT JOIN is also known as LEFT OUTER JOIN.

INPUT:-

```

CREATE TABLE MARKSHEET1(
    Roll_no INT,
    Name VARCHAR(50),
    Maths_Marks INT,
    English_Marks INT,
    Science_Marks INT,
    Hindi_Marks INT
);
INSERT INTO MARKSHEET1 VALUES('01','RAMA','80','76','78','90')
INSERT INTO MARKSHEET1 VALUES('03','RAJESH','85','98','89','98')
INSERT INTO MARKSHEET1 VALUES('04','JAANVI','90','90','98','76')
INSERT INTO MARKSHEET1 VALUES('02','RIA','87','75','90','89')
SELECT * FROM MARKSHEET1;

CREATE TABLE DETAIL1(
    Roll_no INT,
    Course_ID INT,
);
INSERT INTO DETAIL1 VALUES('02','01')
INSERT INTO DETAIL1 VALUES('01','02')
INSERT INTO DETAIL1 VALUES('03','03')
INSERT INTO DETAIL1 VALUES('04','04')
SELECT * FROM DETAIL1;

SELECT MARKSHEET1.Name,DETAIL1.Course_ID
FROM MARKSHEET1
LEFT JOIN DETAIL1
ON MARKSHEET1.Roll_no = DETAIL1.Roll_no;

```

OUTPUT:-

	Roll_no	Name	Maths_Marks	English_Marks	Science_Marks	Hindi_Marks
1	1	RAMA	80	76	78	90
2	3	RAJESH	85	98	89	98
3	4	JAANVI	90	90	98	76
4	2	RIA	87	75	90	89

	Roll_no	Course_ID
1	2	1
2	1	2
3	3	3
4	4	4

	Name	Course_ID
1	RAMA	2
2	RAJESH	3
3	JAANVI	4
4	RIA	1

RIGHT JOIN: RIGHT JOIN is similar to LEFT JOIN. This join returns all the rows of the table on the right side of the join and matching rows for the table on the left side of join. The rows for which there is no matching row on left side, the result-set will contain *null*. RIGHT JOIN is also known as RIGHT OUTER

INPUT:-

```

CREATE TABLE MARKSHEET1(
    Roll_no INT,
    Name VARCHAR(50),
    Maths_Marks INT,
    English_Marks INT,
    Science_Marks INT,
    Hindi_Marks INT
);
INSERT INTO MARKSHEET1 VALUES('01','RAMA','80','76','78','90')
INSERT INTO MARKSHEET1 VALUES('03','RAJESH','85','98','89','98')
INSERT INTO MARKSHEET1 VALUES('04','JAANVI','90','90','98','76')
INSERT INTO MARKSHEET1 VALUES('02','RIA','87','75','90','89')
SELECT * FROM MARKSHEET1;

CREATE TABLE DETAIL1(
    Roll_no INT,
    Course_ID INT,
);
INSERT INTO DETAIL1 VALUES('02','01')
INSERT INTO DETAIL1 VALUES('01','02')
INSERT INTO DETAIL1 VALUES('03','03')
INSERT INTO DETAIL1 VALUES('04','04')
INSERT INTO DETAIL1 VALUES('05','09')
INSERT INTO DETAIL1 VALUES('06','10')
INSERT INTO DETAIL1 VALUES('07','12')
INSERT INTO DETAIL1 VALUES('08','14')
SELECT * FROM DETAIL1;

SELECT MARKSHEET1.Name,DETAIL1.Course_ID
FROM MARKSHEET1
RIGHT JOIN DETAIL1
ON MARKSHEET1.Roll_no = DETAIL1.Roll_no;

```

OUTPUT:-

	Roll_no	Name	Maths_Marks	English_Marks	Science_Marks	Hindi_Marks
1	1	RAMA	80	76	78	90
2	3	RAJESH	85	98	89	98
3	4	JAANVI	90	90	98	76
4	2	RIA	87	75	90	89

	Roll_no	Course_ID
1	2	1
2	1	2
3	3	3
4	4	4
5	5	9
6	6	10
7	7	12
8	8	14

	Name	Course_ID
1	RIA	1
2	RAMA	2
3	RAJESH	3
4	JAANVI	4
5	NULL	9
6	NULL	10
7	NULL	12
8	NULL	14

FULL JOIN: FULL JOIN creates the result-set by combining result of both LEFT JOIN and RIGHT JOIN. The result-set will contain all the rows from both the tables. The rows for which there is no matching, the result-set will contain

INPUT:-

```

CREATE TABLE MARKSHEET1(
    Roll_no INT,
    Name VARCHAR(50),
    Maths_Marks INT,
    English_Marks INT,
    Science_Marks INT,
    Hindi_Marks INT
);
INSERT INTO MARKSHEET1 VALUES('01','RAMA','80','76','78','90')
INSERT INTO MARKSHEET1 VALUES('03','RAJESH','85','98','89','98')
INSERT INTO MARKSHEET1 VALUES('04','JAANVI','90','90','98','76')
INSERT INTO MARKSHEET1 VALUES('02','RIA','87','75','90','89')
SELECT * FROM MARKSHEET1;

CREATE TABLE DETAIL1(
    Roll_no INT,
    Course_ID INT,
);
INSERT INTO DETAIL1 VALUES('02','01')
INSERT INTO DETAIL1 VALUES('01','02')
INSERT INTO DETAIL1 VALUES('03','03')
INSERT INTO DETAIL1 VALUES('04','04')
INSERT INTO DETAIL1 VALUES('05','09')
INSERT INTO DETAIL1 VALUES('06','10')
INSERT INTO DETAIL1 VALUES('07','12')
INSERT INTO DETAIL1 VALUES('08','14')
SELECT * FROM DETAIL1;

SELECT MARKSHEET1.Name,DETAIL1.Course_ID
FROM MARKSHEET1
FULL JOIN DETAIL1
ON MARKSHEET1.Roll_no = DETAIL1.Roll_no;

```

OUTPUT:-

	Roll_no	Name	Maths_Marks	English_Marks	Science_Marks	Hindi_Marks
1	1	RAMA	80	76	78	90
2	3	RAJESH	85	98	89	98
3	4	JAANVI	90	90	98	76
4	2	RIA	87	75	90	89

	Roll_no	Course_ID
1	2	1
2	1	2
3	3	3
4	4	4
5	5	9
6	6	10
7	7	12
8	8	14

	Name	Course_ID
1	RAMA	2
2	RAJESH	3
3	JAANVI	4
4	RIA	1
5	NULL	9
6	NULL	10
7	NULL	12
8	NULL	14

INNER JOIN: The INNER JOIN keyword selects all rows from both the tables as long as the condition satisfies. This keyword will create the result-set by combining all rows from both the tables where the condition satisfies i.e value of the common field will be same.

INPUT:-

```

CREATE TABLE MARKSHEET1(
    Roll_no INT,
    Name VARCHAR(50),
    Maths_Marks INT,
    English_Marks INT,
    Science_Marks INT,
    Hindi_Marks INT
);
INSERT INTO MARKSHEET1 VALUES('01','RAMA','80','76','78','90')
INSERT INTO MARKSHEET1 VALUES('03','RAJESH','85','98','89','98')
INSERT INTO MARKSHEET1 VALUES('04','JAANVI','90','90','98','76')
INSERT INTO MARKSHEET1 VALUES('02','RIA','87','75','90','89')
SELECT * FROM MARKSHEET1;
CREATE TABLE DETAIL1(
    Roll_no INT,
    Course_ID INT,
);
INSERT INTO DETAIL1 VALUES('02','01')
INSERT INTO DETAIL1 VALUES('01','02')
INSERT INTO DETAIL1 VALUES('03','03')
INSERT INTO DETAIL1 VALUES('04','04')
INSERT INTO DETAIL1 VALUES('05','09')
INSERT INTO DETAIL1 VALUES('06','10')
INSERT INTO DETAIL1 VALUES('07','12')
INSERT INTO DETAIL1 VALUES('08','14')
SELECT * FROM DETAIL1;
SELECT MARKSHEET1.Name,DETAIL1.Course_ID
FROM MARKSHEET1
INNER JOIN DETAIL1
ON MARKSHEET1.Roll_no = DETAIL1.Roll_no;

```

OUTPUT:-

	Roll_no	Name	Maths_Marks	English_Marks	Science_Marks	Hindi_Marks
1	1	RAMA	80	76	78	90
2	3	RAJ...	85	98	89	98
3	4	JAA...	90	90	98	76
4	2	RIA	87	75	90	89

	Roll_no	Course_ID
1	2	1
2	1	2
3	3	3
4	4	4
5	5	9
6	6	10
7	7	12
8	8	14

	Name	Course_ID
1	RIA	1
2	RAMA	2
3	RAJ...	3
4	JAA...	4