SQL ASSIGNMENT SS

--T N B K Yesasvi

Insert more than 10 records in each and every table created.

Snap of the all the tables once the insertion is completed.

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity
John	S	1	Residence in Telangana	M	19	Hyderbad
Jim	Kim	2	Residence in Andhrapradesh	M	20	Chittor
Roman	R	3	Residence in Telangana, India	M	19	Khammam
Bailey	Roy	4	Residence in Telangana	F	21	Secunderbad
Becky	L	5	Residence in Andhrapradesh	F	20	Nellore
Ramu	Singh	6	Residence in Telangana	M	19	Hyderbad
Kiran	Kumar	7	Residence in Andhrapradesh	M	20	Nellore
Sony	Yadav	8	Residence in Mumbai	F	19	Mumbai
Sakshi	Singh	9	Residence in Uttar Pradesh	F	21	Lucknow
Indra	Kiran	10	Residence in Andhrapradesh	M	21	Nellore
Vikram	R	11	Residence in Andhrapradesh	M	20	Guntur
Sam	Jim	13	Residence in Andhra Pradesh	F	21	Gudur
NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
1    SELECT * FROM StudentAdmissionPaymentDetails;
2
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	AmountPaid	AmountBalance	PaymentMode	PaymentType	BankName	PaymentStatus
1	5000	5000	Online	UPI	ANDB	Success
2	7000	3000	Online	UPI	SBI	Success
3	6000	4000	Offline	Cash	ANDB	Success
4	5000	5000	Online	Debit Card	CB	Fail
5	4000	6000	Offline	Check	SBI	Success
6	5500	4500	Online	UPI	UN	Success
7	9000	1000	Online	Credit Card	HDFC	Success
8	2000	8000	Offline	DD	Axis	Success
9	8000	2000	Online	UPI	HDFC	Fail
10	9500	500	Offline	Check	Axis	Success
11	7500	2500	Online	Net Banking	HDFC	Success

```
1 •  SELECT * FROM student.studentsubjectinformation;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SubjectOpted	StudentRollNo	SubjectTotalMarks	SubjectObtainedMarks	StudentMarksPercentage	StudentGrade
Science	1	50	45	NULL	A
Maths	2	50	43	NULL	A
Economics	3	50	47	NULL	A+
Science	4	50	41	NULL	A
Biology	5	50	47	NULL	A+
English	6	50	48	NULL	O
Computers	7	50	50	NULL	O
Maths	8	50	40	NULL	B
Science	9	50	45	NULL	A
Science	10	50	44	NULL	A
English	11	50	41	NULL	A
History	2	50	44	NULL	A
Biology	7	50	49	NULL	O
Zoology	13	50	50	NULL	O

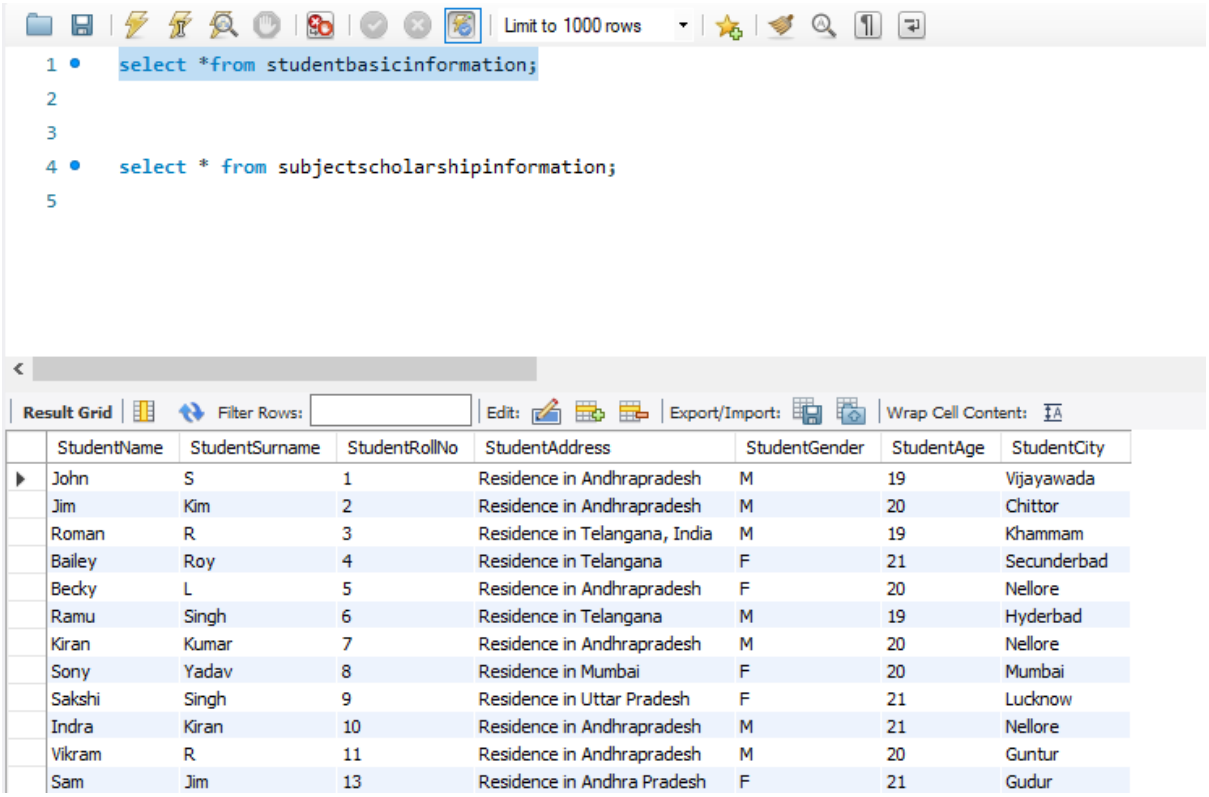
```
1 •  select * from subjectscholarshipinformation
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipOpted	ScholarshipStatus
1	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
2	Off-campus	Student Scholarship	5000	Merit	Yes	Active
3	On-campus	Toppers Scholarship	1000	Merit	Yes	Inactive
4	On-campus	Charity Scholarship	10000	Normal	Yes	Active
5	Off-campus	Toppers Scholarship	10000	Merit	Yes	Active
6	On-campus	Charity Scholarship	8000	Normal	Yes	Inactive
7	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
8	On-campus	Charity Scholarship	6000	Merit	Yes	Active
9	On-campus	Special Scholarship	3000	Normal	Yes	Active
10	On-campus	Toppers Scholarship	9000	Merit	Yes	Inactive
11	Off-campus	Toppers Scholarship	10000	Normal	Yes	Active
13	On-campus	Toppers Scholarship	8000	Merit	Yes	Active

Update any 5 records of your choice in any table like update the StudentAddress with some other address content and likewise so on with any records of any table of your choice

6. Snap of the all the tables post updation



StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity
John	S	1	Residence in Andhrapradesh	M	19	Vijayawada
Jim	Kim	2	Residence in Andhrapradesh	M	20	Chittor
Roman	R	3	Residence in Telangana, India	M	19	Khammam
Bailey	Roy	4	Residence in Telangana	F	21	Secunderbad
Becky	L	5	Residence in Andhrapradesh	F	20	Nellore
Ramu	Singh	6	Residence in Telangana	M	19	Hyderbad
Kiran	Kumar	7	Residence in Andhrapradesh	M	20	Nellore
Sony	Yadav	8	Residence in Mumbai	F	20	Mumbai
Sakshi	Singh	9	Residence in Uttar Pradesh	F	21	Lucknow
Indra	Kiran	10	Residence in Andhrapradesh	M	21	Nellore
Vikram	R	11	Residence in Andhrapradesh	M	20	Guntur
Sam	Jim	13	Residence in Andhra Pradesh	F	21	Gudur

```
3
4 •  select * from subjectscholarshipinformation;
5
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipOpted	ScholarshipStatus
1	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
2	Off-campus	Student Scholarship	5000	Merit	Yes	Active
3	On-campus	Toppers Scholarship	1000	Merit	Yes	Inactive
4	On-campus	Charity Scholarship	2000	Normal	Yes	Active
5	Off-campus	Toppers Scholarship	4000	Merit	Yes	Active
6	On-campus	Charity Scholarship	8000	Normal	Yes	Inactive
7	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
8	On-campus	Charity Scholarship	6000	Merit	Yes	Active
9	On-campus	Special Scholarship	3000	Normal	Yes	Active
10	On-campus	Toppers Scholarship	9000	Merit	Yes	Inactive
11	On-campus	Special Scholarship	10000	Normal	Yes	Active
13	On-campus	Toppers Scholarship	8000	Merit	Yes	Active

7.Select the student details records who has received the scholarship more than 5000Rs/-

```
1 •  SELECT * FROM subjectscholarshipinformation WHERE ScholarshipAmount>5000;
```

sult Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipOpted	ScholarshipStatus
1	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
6	On-campus	Charity Scholarship	8000	Normal	Yes	Inactive
7	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
8	On-campus	Charity Scholarship	6000	Merit	Yes	Active
10	On-campus	Toppers Scholarship	9000	Merit	Yes	Inactive
11	On-campus	Special Scholarship	10000	Normal	Yes	Active
13	On-campus	Toppers Scholarship	8000	Merit	Yes	Active

Select the students who opted for scholarship but has not got the scholarship

```
SELECT * FROM subjectscholarshipinformation WHERE ScholarshipOpted = 'Yes'
AND ScholarshipStatus = 'Inactive';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipOpted	ScholarshipStatus
3	On-campus	Toppers Scholarship	1000	Merit	Yes	Inactive
6	On-campus	Charity Scholarship	8000	Normal	Yes	Inactive
10	On-campus	Toppers Scholarship	9000	Merit	Yes	Inactive

Fill in data for the percentage column i.e. StudentMarksPercentage in the table StudentSubjectInformation by creating and using the stored procedure created

```
SELECT * FROM student.studentsubjectinformation;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SubjectOpted	StudentRollNo	SubjectTotalMarks	SubjectObtainedMarks	StudentMarksPercentage	StudentGrade
Science	1	50	45	90	A
Maths	2	50	43	86	A
Economics	3	50	47	94	A+
Science	4	50	41	82	A
Biology	5	50	47	94	A+
English	6	50	48	96	O
Computers	7	50	50	100	O
Maths	8	50	40	80	B
Science	9	50	45	90	A
Science	10	50	44	88	A
English	11	50	41	82	A
History	2	50	44	88	A
Biology	7	50	49	98	O
Zoology	13	50	50	100	O

Decide the category of the scholarship depending upon the marks/percentage obtained by the student and likewise update the ScholarshipCategory column, create a stored procedure in order to handle this operation

```
SELECT * FROM SubjectScholarshipInformation;
SELECT * FROM StudentSubjectInformation;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipOpted	ScholarshipStatus
1	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
2	Off-campus	Student Scholarship	5000	NonMerit	Yes	Active
3	On-campus	Toppers Scholarship	1000	Merit	Yes	Inactive
4	On-campus	Charity Scholarship	2000	NonMerit	Yes	Active
5	Off-campus	Toppers Scholarship	4000	Merit	Yes	Active
6	On-campus	Charity Scholarship	8000	Merit	Yes	Inactive
7	On-campus	Toppers Scholarship	10000	Merit	Yes	Active
8	On-campus	Charity Scholarship	6000	NonMerit	Yes	Active
9	On-campus	Special Scholarship	3000	Merit	Yes	Active
10	On-campus	Toppers Scholarship	9000	NonMerit	Yes	Inactive
11	On-campus	Special Scholarship	10000	NonMerit	Yes	Active
13	On-campus	Toppers Scholarship	8000	Merit	Yes	Active

Create the View which shows balance amount to be paid by the student along with the student detailed information (use join)

```
    ON
    studentbasicinformation.StudentRollNo = studentadmissionpaymentdetails.StudentRollNo;

SELECT * FROM Balance_View;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

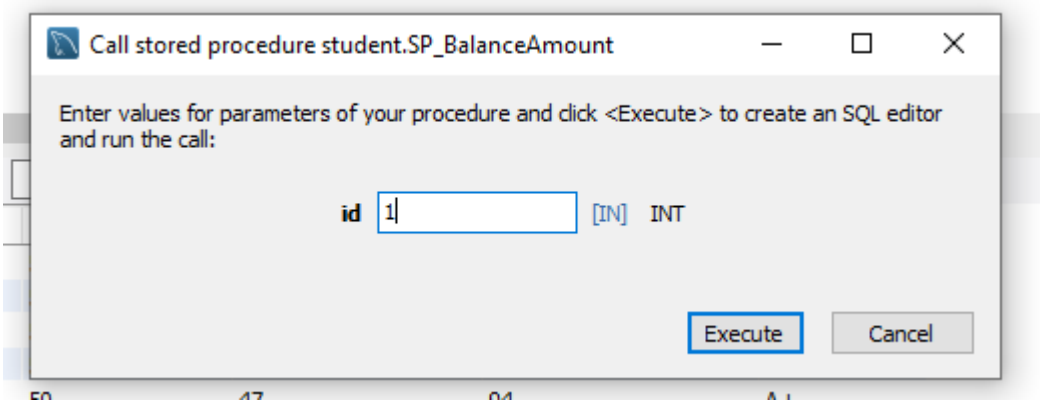
StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity	AmountBalance
John	S	1	Residence in Andhrapradesh	M	19	Vijayawada	5000
Jim	Kim	2	Residence in Andhrapradesh	M	20	Chittor	3000
Roman	R	3	Residence in Telangana, India	M	19	Khammam	4000
Bailey	Roy	4	Residence in Telangana	F	21	Secunderbad	5000
Becky	L	5	Residence in Andhrapradesh	F	20	Nellore	6000
Ramu	Singh	6	Residence in Telangana	M	19	Hyderbad	4500
Kiran	Kumar	7	Residence in Andhrapradesh	M	20	Nellore	1000
Sony	Yadav	8	Residence in Mumbai	F	20	Mumbai	8000
Sakshi	Singh	9	Residence in Uttar Pradesh	F	21	Lucknow	2000
Indra	Kiran	10	Residence in Andhrapradesh	M	21	Nellore	500
Vikram	R	11	Residence in Andhrapradesh	M	20	Guntur	2500

Get the details of the students who haven't got any scholarship (use joins/subqueries)

```
    studentbasicinformation.StudentSurname,
    studentbasicinformation.StudentRollNo,
    studentbasicinformation.StudentAddress,
    studentbasicinformation.StudentGender,
    studentbasicinformation.StudentAge,
    studentbasicinformation.StudentCity
    FROM
    studentbasicinformation
    INNER JOIN
    subjectscholarshipinformation
    ON
    studentbasicinformation.StudentRollNo =subjectscholarshipinformation.StudentRollNo
    AND
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity
Roman	R	3	Residence in Telangana, India	M	19	Khammam
Ramu	Singh	6	Residence in Telangana	M	19	Hyderbad
Indra	Kiran	10	Residence in Andhrapradesh	M	21	Nellore



```
call student.SP_BalanceAmount(1);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity	AmountBalance
John	S	1	Residence in Andhrapradesh	M	19	Vijayawada	5000

Limit to 1000 rows

```
SELECT
*
FROM
studentsubjectinformation
ORDER BY
StudentMarksPercentage
DESC LIMIT 0,5;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

SubjectOpted	StudentRollNo	SubjectTotalMarks	SubjectObtainedMarks	StudentMarksPercentage	StudentGrade
Computers	7	50	50	100	O
Zoology	13	50	50	100	O
Biology	7	50	49	98	O
English	6	50	48	96	O
Economics	3	50	47	94	A+

ctinformation1 ×

Try to use all the three types of join learned today in a relevant way, and explain the same why you thought of using that particular join for your selected scenarios (try to cover relevant and real time scenarios for all the three studied joins)

```
11
12    /*Inner Join: Here Inner join is used to retrive the data of students whose grades are assigned.*/
13    SELECT
14    studentbasicinformation.StudentName,
15    studentbasicinformation.StudentSurname,
16    studentbasicinformation.StudentRollNo,
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentGrade
Jim	Kim	2	A
Roman	R	3	A+
Bailey	Roy	4	A
Becky	L	5	A+
Ramu	Singh	6	O
Kiran	Kumar	7	O
Kiran	Kumar	7	O
Sony	Yadav	8	B
Sakshi	Singh	9	A
Indra	Kiran	10	A
Indra	Kiran	10	NULL
Vikram	R	11	A
Vikram	R	11	NULL
Sam	Jim	13	O

Limit to 1000 rows

```
25
26    /*Left Join: Here Left Join is used to retrive the data of all the students in Studentbasicinformation Table and Matched fields
27    Studentsubjectinformation Table*/
28
29    SELECT
30    studentbasicinformation.StudentName,
31    studentbasicinformation.StudentSurname,
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentGrade
John	S	1	A
Jim	Kim	2	A
Jim	Kim	2	A
Roman	R	3	A+
Bailey	Roy	4	A
Becky	L	5	A+
Ramu	Singh	6	O
Kiran	Kumar	7	O
Kiran	Kumar	7	O
Sony	Yadav	8	B
Sakshi	Singh	9	A
Indra	Kiran	10	A
Indra	Kiran	10	NULL
Vikram	R	11	A
Vikram	R	11	NULL
Sam	Jim	13	O
Rino	NULL	15	NULL

Result Grid | Form Editor | Field Types | Query Stats | Execution Plan

Result 4 | Read Only

```
33    studentsubjectinformation.StudentGrade
34    FROM
35    studentbasicinformation
36    LEFT JOIN
37    studentsubjectinformation
38    ON
39    studentbasicinformation.StudentRollNo =studentsubjectinformation.StudentRollNo;
40
41
42    /*Right Join: Here Right Join is used to retrive the data of all grades in Studentsubjectinformation table and matched details o
43    Studentbasicinformation table.*/
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentGrade
John	S	1	A
Jim	Kim	2	A
Roman	R	3	A+
Bailey	Roy	4	A
Becky	L	5	A+
Ramu	Singh	6	O
Kiran	Kumar	7	O
Sony	Yadav	8	B
Sakshi	Singh	9	A
Indra	Kiran	10	A
Vikram	R	11	A
Jim	Kim	2	A
Kiran	Kumar	7	O
Sam	Jim	13	O
Vikram	R	11	NULL

Get the count of the Scholarship category which is highly been availed by the students, i.e. get the count of the total number of students corresponding to the each scholarships category

```
4     AS
5     COUNT
6     FROM
7     subjectscholarshipinformation
8     WHERE
9     ScholarshipCategory
10    IS NOT NULL
11    GROUP BY
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ScholarshipCategory	COUNT
Merit	7
NonMerit	5

Along with the assignment no. 17 try to retrieve the maximum used scholarship category

```
SELECT
ScholarshipCategory, COUNT(*)
AS
'MaxCount'
FROM
subjectscholarshipinformation
WHERE
ScholarshipCategory
IS NOT NULL
GROUP BY
ScholarshipCategory
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

ScholarshipCategory	MaxCount
Merit	7

Retrieve the percentage of the students along with students detailed information who has scored the highest percentage along with availing the maximum scholarship amount

```
INNER JOIN
studentsubjectinformation
INNER JOIN
subjectscholarshipinformation
ON
studentbasicinformation.StudentRollNo = studentsubjectinformation.StudentRollNo
AND
studentbasicinformation.StudentRollNo = subjectscholarshipinformation.StudentRollNo
WHERE
StudentMarksPercentage = (SELECT MAX(StudentMarksPercentage)
FROM
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentGender	StudentAge	StudentCity	StudentMarksPercentage	ScholarshipAmount
Kiran	Kumar	7	Residence in Andhrapradesh	M	20	Nellore	100	10000