

SQL CONCEPTS AND FUNDAMENTALS

Solution 4:

```
mysql> select * from studentbasicinformation;
```

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentFatherName	StudentMotherName	StudentClass	optedForScholarship
Andrew	Buchanan	453461	28959 Sonya Drive South Ricardo, HI 63087	Martin Sims	Amanda Moore	12	no
Jordan	Graven	503480	0496 William Tunnel Apt. 238 Port Paulberg, PA 63031	Edmund Scarberry	Rosa Lynch	4	no
Sara	Baumbach	1152857	95019 Cantrell Road Suite 997 Brendafort, NJ 37636	Christopher Schleider	Arlene Gilmore	10	yes
Maria	Johnston	1964390	3138 Davis Mountain Apt. 931 South Stevenchester, KY 51253	Luis Andreas	Marilyn Young	11	yes
Hubert	Harvey	3370518	20980 James Courts East Barbara, OR 36776	Alan Snipes	Arlene Charles	12	no
Jason	Garcia	4316976	0784 Jay Mill Johnstonon, NY 83218	Rick Palmer	Ilana Perez	13	yes
Linda	Sanders	5535741	57719 Payne Square Apt. 808 Wrightville, OR 75477	Roy Keith	Luana Lord	8	yes
Alan	Grollimund	6439006	4178 Christopher Views Suite 293 Lake Nathaniel, IN 49936	Gerard Reyna	Ruby Scallon	2	no
Richard	Mcconkey	6617786	0706 Natalie Crossroad Suite 970 Port Christinefurt, VT 60236	Jesse Bond	Maryann Dixon	1	no
Renee	Sanchez	6784727	02503 Scott Courts Apt. 750 West Dawntown, WY 64087	Chad Keenan	Amber Long	2	yes
Mary	Redfield	8209187	979 Shelly Hollow East Laurieview, TX 78435	Bruce Soliz	Hazel Fulmer	5	yes
Robert	Rinaldi	8600206	31233 Brown Flat Suite 773 Harrisonhaven, MA 98811	Robert Hall	Barbara Lesueur	3	no
Hugo	Barre	8702173	1304 Andrew Hill South Natasha, WI 17701	Daniel Cook	Cathy Mix	4	no
Melinda	Wright	9444289	953 Ashlee Row Apt. 619 Jonesmouth, VA 20178	James Ebling	Kathryn Marrs	3	yes

```
14 rows in set (0.00 sec)
```

```
mysql>
```

MySQL 8.0 Command Line Client

```
mysql> select * from StudentSubjectInformation;
```

SubjectOpted	SubjectObtainedMarks	StudentMarksPercentage	IsPassed	StudentRollNo	SubjectTotalMa
Astronomy,Biology,Biomedical,Sciences,Chemistry,Earth Sciences	280	46.67	YES	57884	
Food Science and Technology	41	41.13	YES	1181076	
Earth Sciences,Environmental Sciences	118	59.13	YES	1857843	
Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences	490	70.13	YES	3839847	
Biomedical,Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology	398	66.33	YES	5195773	
Earth Sciences	81	81.13	YES	5939517	
Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences	236	39.33	YES	6166948	
Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences	129	32.25	NO	6463208	
Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography	6	1.13	NO	7280114	
Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences	497	99.40	YES	7647913	
Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences	125	25.13	NO	7653687	
Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences	322	53.67	YES	7764433	
Life Sciences,Materials Sciences	67	33.50	YES	8087388	
Life Sciences,Materials Sciences	20	10.13	NO	8741329	
Biomedical,Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography	418	38.13	YES	9878512	1

```
15 rows in set (0.00 sec)
```

```
mysql>
```

```
mysql> select * from StudentAdmissionPaymentDetails;
```

StudentRollNo	AmountPaid	AmountBalance	Fines	MethodOfPayment	Hosteller	LateAdmission
673689	190792	92318	3880	Debit Card	no	no
1734814	197300	115831	5727	Cheque	no	no
1847020	107791	5678	8944	Net Banking	yes	no
1919195	151710	89191	8378	Debit Card	no	no
2765565	71083	119767	8264	Debit Card	no	yes
3895324	17269	98397	9331	UPI	no	yes
4074423	139373	174240	6935	Debit Card	no	yes
4135393	149257	91773	8465	UPI	no	yes
4739818	27154	117197	2644	Cash	yes	yes
7789414	131743	184931	710	UPI	no	no
8034935	42509	70051	1358	Demand Draft	no	yes
8451379	84621	56651	2300	Credit Card	no	yes
8862816	101089	79902	6390	UPI	no	no
8867862	57978	188461	1727	Cheque	no	yes
9341595	54035	6123	3851	UPI	yes	no

```
15 rows in set (0.03 sec)
```

```
mysql>
mysql> select * from SubjectScholarshipInformation;
```

StudentRollNo	ScholarshipName	ScholarshipDescription	ScholarshipAmount	ScholarshipCategory	ScholarshipTenure	ScholarshipCurrency
436721	Raman Kant Munjal Scholarship	DESCRIPTION - 7	730000	Private	3	INR
663255	L'Oréal India For Young Women in Science Scholarships	DESCRIPTION - 4	25000	Private	2	INR
1434786	CLP India Scholarship Scheme	DESCRIPTION - 1	1800	Private	4	USD
1702726	L'Oréal India For Young Women in Science Scholarships	DESCRIPTION - 13	25000	Private	2	INR
1998446	The Anant Fellowship	DESCRIPTION - 2	100000	Government	4	INR
2942796	Sitaram Jindal Foundation Scholarship Scheme	DESCRIPTION - 6	12000	Semi-Government	4	INR
3048303	Raman Kant Munjal Scholarship	DESCRIPTION - 15	730000	Private	3	INR
3608365	Sitaram Jindal Foundation Scholarship Scheme	DESCRIPTION - 12	12000	Semi-Government	4	INR
4103209	L'Oréal India For Young Women in Science Scholarships	DESCRIPTION - 9	25000	Private	2	INR
4247655	CLP India Scholarship Scheme	DESCRIPTION - 10	1800	Private	4	USD
4984015	Sitaram Jindal Foundation Scholarship Scheme	DESCRIPTION - 5	12000	Semi-Government	4	INR
5696799	The Anant Fellowship	DESCRIPTION - 14	100000	Government	4	INR
5885182	HERO INDIA	DESCRIPTION - 3	2400	Private	2	INR
8549137	Japanese Government (MEXT) Scholarships	DESCRIPTION - 8	117000	Foreign	4	YEN
9300900	NIU Scholarship cum Admission Test (NSAT)	DESCRIPTION - 11	8500	Private	3	INR

```
15 rows in set (0.00 sec)
```

```
mysql>
```

Solution 5 & 6:

```
MySQL 8.0 Command Line Client

mysql> update StudentBasicInformation set StudentAddress='96968 Anderson Centers Suite 901 Williamsport, MI 90376' where StudentRollNo=4316976;
Query OK, 1 row affected (0.61 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentBasicInformation set StudentAddress='265 Cameron Ferry South Sheila, AL 85300' where StudentRollNo=370518;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0 Changed: 0 Warnings: 0

mysql> update StudentBasicInformation set StudentAddress='886 Teresa Course South Dennisland, RI 22917' where StudentRollNo=4316976;
Query OK, 1 row affected (5.09 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentBasicInformation set StudentAddress='0331 Jon Route Robertfort, MI 84482' where StudentRollNo=8209187;
Query OK, 1 row affected (1.48 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentBasicInformation set StudentAddress='0506 Maria Via Apt. 031 Saratown, SC 20954' where StudentRollNo=9444289;
Query OK, 1 row affected (1.80 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql>
```

```
MySQL 8.0 Command Line Client

mysql> update StudentAdmissionPaymentDetails set MethodOfPayment='UPI' where StudentRollNo=3895324;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1 Changed: 0 Warnings: 0

mysql> update StudentAdmissionPaymentDetails set MethodOfPayment='UPI' where StudentRollNo=4074423;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentAdmissionPaymentDetails set MethodOfPayment='Cheque' where StudentRollNo=673689;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentAdmissionPaymentDetails set MethodOfPayment='Cheque' where StudentRollNo=8862816;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentAdmissionPaymentDetails set MethodOfPayment='Cash' where StudentRollNo=8867862;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql>
```

```
MySQL 8.0 Command Line Client

mysql> update StudentSubjectInformation set SubjectOpted='Materials Sciences' where StudentRollNo=5939517;
Query OK, 1 row affected (0.73 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentSubjectInformation set SubjectOpted='Materials Sciences,Mathematics,Physical Geography' where StudentRollNo=1181076;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentSubjectInformation set SubjectOpted='Life Sciences,Materials Sciences' where StudentRollNo=8087388;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1 Changed: 0 Warnings: 0

mysql> update StudentSubjectInformation set SubjectOpted='Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography,Physics' where StudentRollNo=6463208;
Query OK, 1 row affected (0.05 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update StudentSubjectInformation set SubjectOpted='Life Sciences' where StudentRollNo=7764433;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql>
```


MySQL 8.0 Command Line Client

```
mysql> update SubjectScholarshipInformation set ScholarshipDescription='Description -> 20' where StudentRollNo=1702726;
Query OK, 1 row affected (1.45 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update SubjectScholarshipInformation set ScholarshipDescription='Description -> 26' where StudentRollNo=5696799;
Query OK, 1 row affected (0.15 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update SubjectScholarshipInformation set ScholarshipDescription='Description -> 10' where StudentRollNo=436721;
Query OK, 1 row affected (0.28 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update SubjectScholarshipInformation set ScholarshipDescription='Description -> 23' where StudentRollNo=5885182;
Query OK, 1 row affected (0.35 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update SubjectScholarshipInformation set ScholarshipDescription='Description -> 23' where StudentRollNo=1998446;
Query OK, 1 row affected (0.08 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql>
```

MySQL 8.0 Command Line Client

```
mysql> select * from studentbasicinformation;
```

StudentName	StudentSurname	StudentRollNo	StudentAddress	StudentFatherName	StudentMotherName	StudentClass
Andrew	Buchanan	453461	28959 Sonya Drive South Ricardo, MI 63887	Martin Sims	Amanda Moore	12
Jordan	Graven	593488	8496 William Tunnel Apt. 238 Port Paulberg, PA 63031	Edmund Scarberry	Rosa Lynch	4
Sara	Baumbach	1152857	95019 Centrell Road Suite 097 Wendatort, NJ 37636	Christopher Schleider	Arlene Gilaore	10
Maria	Johnston	1964390	3138 Davis Mountain Apt. 931 South Stevenchester, KY 51253	Luis Andreas	Marilyn Young	11
Hubert	Harvey	3370518	20980 James Courts East Barbara, OR 36776	Alan Snipes	Arline Charles	12
Jason	Garcia	4316976	886 Teresa Course South Dennisland, RI 22017	Rick Palmer	Ilano Perez	13
Linda	Sanders	5535741	57719 Payne Square Apt. 888 Wrightville, OR 75477	Roy Keith	Luna Lord	8
Alan	Grollmund	6439006	4178 Christopher Views Suite 293 Lake Nathaniel, IN 49936	Gerard Reyna	Ruby Scallon	2
Richard	Mcconkey	6617786	0786 Natalie Crossroad Suite 970 Port Christinefurt, VT 60236	Jesse Bond	Maryann Dixon	1
Renee	Sanchez	6784727	02503 Scott Courts Apt. 750 West Downtown, NY 64087	Chad Keenan	Amber Long	2
Robert	Owen	7107272	213 Theresa Island Suite 452 Kariside, NH 43094	Manuel Cleveland	Maria Huffman	7
Mary	Redfield	8209187	0331 Jon Route Robertfort, MI 84482	Bruce Soliz	Hazel Fulmer	5
Robert	Rinaldi	8600206	31233 Brown Flat Suite 773 Harrisonhaven, MA 98811	Robert Hall	Barbara Lesueur	3
Hugo	Barre	8702173	1304 Andrew Hill South Natasha, MI 17701	Daniel Cook	Cathy Mix	4
Melinda	Wright	9444289	0506 Maria Via Apt. 031 Saratown, SC 20954	James Ebling	Kathryn Marrs	3

```
15 rows in set (0.00 sec)

mysql>
```

MySQL 8.0 Command Line Client

```
mysql> select * from StudentAdmissionPaymentDetails;
```

StudentRollNo	AmountPaid	AmountBalance	Fines	MethodOfPayment	Hosteller	LateAdmission
673689	190792	92318	3880	Cheque	no	no
1734814	197300	115831	5727	Cheque	no	no
1847020	107791	5678	8944	Net Banking	yes	no
1919195	151710	89191	8378	Debit Card	no	no
2765565	71083	119767	8264	Debit Card	no	yes
3895324	17269	98397	9331	UPI	no	yes
4074423	139373	174240	6935	UPI	no	yes
4135393	149257	91773	8465	UPI	no	yes
4739818	27154	117197	2644	Cash	yes	yes
7789414	131743	184931	710	UPI	no	no
8034935	42509	70051	1358	Demand Draft	no	yes
8451379	84621	56651	2300	Credit Card	no	yes
8862816	101089	79902	6390	Cheque	no	no
8867862	57978	188461	1727	Cash	no	yes
9341595	54035	6123	3851	UPI	yes	no

```
15 rows in set (0.00 sec)

mysql>
```

MySQL 8.0 Command Line Client

```
mysql> select * from StudentSubjectInformation;
+-----+-----+-----+-----+-----+-----+-----+
| SubjectOpted | SubjectTotalMarks | SubjectObtainedMarks | StudentMarksPercentage | IsPassed | StudentRollNo |
+-----+-----+-----+-----+-----+-----+-----+
| Astronomy,Biology,Biomedical,Sciences,Chemistry,Earth Sciences | 600 | 280 | 46.67 | YES | 11 |
| Materials Sciences,Mathematics,Physical Geography | 400 | 164 | 41.00 | YES | 18 |
| Earth Sciences,Environmental Sciences | 200 | 118 | 59.00 | YES | 38 |
| Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences | 700 | 490 | 70.00 | YES | 51 |
| Biomedical,Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology | 600 | 398 | 66.33 | YES | 59 |
| Materials Sciences | 100 | 81 | 81.00 | YES | 61 |
| Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences | 600 | 236 | 39.33 | YES | 64 |
| Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography,Physics | 400 | 129 | 32.25 | NO | 72 |
| Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography | 600 | 6 | 1.00 | NO | 76 |
| Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences | 500 | 497 | 99.40 | YES | 76 |
| Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences | 500 | 125 | 25.00 | NO | 77 |
| Life Sciences | 600 | 322 | 53.67 | YES | 80 |
| Life Sciences,Materials Sciences | 200 | 67 | 33.50 | YES | 87 |
| Life Sciences,Materials Sciences | 200 | 20 | 10.00 | NO | 98 |
| Biomedical,Sciences,Chemistry,Earth Sciences,Environmental Sciences,Food Science and Technology,General Sciences,Life Sciences,Materials Sciences,Mathematics,Physical Geography | 1100 | 418 | 38.00 | YES |
+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.00 sec)

mysql>
```

MySQL 8.0 Command Line Client

```
mysql> select * from SubjectScholarshipInformation;
+-----+-----+-----+-----+-----+-----+-----+
| StudentRollNo | ScholarshipName | ScholarshipDescription | ScholarshipAmount | ScholarshipCategory | ScholarshipTenure | ScholarshipCurrency |
+-----+-----+-----+-----+-----+-----+-----+
| 436721 | Raman Kant Munjal Scholarship | Description -> 10 | 730000 | Private | 3 | INR |
| 663255 | L'Oréal India For Young Women in Science Scholarships | DESCRIPTION - 4 | 25000 | Private | 2 | INR |
| 1434786 | CLP India Scholarship Scheme | DESCRIPTION - 1 | 1800 | Private | 4 | USD |
| 1702726 | L'Oréal India For Young Women in Science Scholarships | Description -> 20 | 25000 | Private | 2 | INR |
| 1998446 | The Anant Fellowship | Description -> 23 | 100000 | Government | 4 | INR |
| 2942796 | Sitaram Jindal Foundation Scholarship Scheme | DESCRIPTION - 6 | 12000 | Semi-Government | 4 | INR |
| 3048303 | Raman Kant Munjal Scholarship | DESCRIPTION - 15 | 730000 | Private | 3 | INR |
| 3608365 | Sitaram Jindal Foundation Scholarship Scheme | DESCRIPTION - 12 | 12000 | Semi-Government | 4 | INR |
| 4103209 | L'Oréal India For Young Women in Science Scholarships | DESCRIPTION - 9 | 25000 | Private | 2 | INR |
| 4247655 | CLP India Scholarship Scheme | DESCRIPTION - 10 | 1800 | Private | 4 | USD |
| 4984015 | Sitaram Jindal Foundation Scholarship Scheme | DESCRIPTION - 5 | 12000 | Semi-Government | 4 | INR |
| 5696799 | The Anant Fellowship | Description -> 26 | 100000 | Government | 4 | INR |
| 5885182 | HERO INDIA | Description -> 23 | 2400 | Private | 2 | INR |
| 8549137 | Japanese Government (MEXT) Scholarships | DESCRIPTION - 8 | 117000 | Foreign | 4 | YEN |
| 9300900 | NIU Scholarship cum Admission Test (NSAT) | DESCRIPTION - 11 | 8500 | Private | 3 | INR |
+-----+-----+-----+-----+-----+-----+-----+
15 rows in set (0.05 sec)

mysql>
```

Solution 7:

 MySQL 8.0 Command Line Client

```
mysql> select t1.studentname, t1.studentrollno, t4.scholarshipamount  
-> from studentbasicinformation t1, subjectscholarshipinformation t4  
-> where t1.studentrollno=t4.studentrollno and t4.scholarshipamount>5000;
```

studentname	studentrollno	scholarshipamount
Andrew	453461	730000
Jordan	503480	25000
Maria	1964390	25000
Hubert	3370518	100000
Jason	4316976	12000
Linda	5535741	730000
Alan	6439006	12000
Richard	6617786	25000
Robert	7107272	12000
Mary	8209187	100000
Hugo	8702173	117000
Melinda	9444289	8500

12 rows in set (0.07 sec)

```
mysql> _
```

Solution 8:

MySQL 8.0 Command Line Client

```
mysql> select distinct t1.studentname,t1.studentrollno
  -> from studentbasicinformation t1, subjectscholarshipinformation t4
  -> where t1.studentrollno<>t4.studentrollno and t1.optedForScholarship='yes';
+-----+-----+
| studentname | studentrollno |
+-----+-----+
| Melinda    | 9444289       |
| Mary       | 8209187       |
| Renee      | 6784727       |
| Linda      | 5535741       |
| Jason      | 4316976       |
| Maria      | 1964390       |
| Sara       | 1152857       |
+-----+-----+
7 rows in set (0.00 sec)

mysql> _
```

Solution 9:

MySQL 8.0 Command Line Client

```
mysql>
mysql> DELIMITER //
mysql> CREATE PROCEDURE updatePer(IN targetRoll INT)
  -> BEGIN
  -> update studentsubjectinformation
  -> set studentmarkspercentage=(subjectobtainedmarks*100)/subjecttotalmarks
  -> where studentrollno=rollno;
  -> END//
Query OK, 0 rows affected (0.06 sec)
```


Solution 10:

```
mysql> delimiter //
mysql> create procedure updateScholar(IN RollNo INT)
  -> BEGIN
  -> update SubjectScholarshipInformation
  -> set StudentRollNo=RollNo and ScholarshipName=
  -> case when rollno in (select studentrollno from StudentSubjectInformation where StudentSubjectInformation.StudentMarksPercentage^C
mysql> create procedure updateScholar(IN RollNo INT)
  -> BEGIN
  -> update SubjectScholarshipInformation
  -> set StudentRollNo=RollNo and Scholarshipcategory=
  -> case when rollno IN (select studentrollno from StudentSubjectInformation where StudentSubjectInformation.studentmarkspercentage between 70 and 100) THEN "MERIT"
  -> else "NON-MERIT"
  -> end;
  -> END//
Query OK, 0 rows affected (0.09 sec)

mysql> _
```

Solution 11:

```
MySQL 8.0 Command Line Client
mysql>
mysql> create view BalAmn as
  -> select studentname,studentrollno,amountbalance
  -> from studentbasicinformation join StudentAdmissionPaymentDetails
  -> where student^C
mysql> create view BalAmn as
  -> select t1.studentname,t1.studentrollno,t2.amountbalance
  -> from studentbasicinformation t1 join StudentAdmissionPaymentDetails t2
  -> where t1.studentrollno=t2.studentrollno;
Query OK, 0 rows affected (0.14 sec)

mysql> select * from balamn;
+-----+-----+-----+
| studentname | studentrollno | amountbalance |
+-----+-----+-----+
| Andrew      | 453461        | 92318         |
| Jordan      | 503480        | 115831        |
| Sara        | 1152857       | 5678          |
| Maria       | 1964390       | 89191         |
| Hubert      | 3370518       | 119767        |
| Jason       | 4316976       | 98397         |
| Linda       | 5535741       | 174240        |
| Alan        | 6439006       | 91773         |
| Richard     | 6617786       | 117197        |
| Renee       | 6784727       | 184931        |
| Mary        | 8209187       | 56651         |
| Robert      | 8600206       | 79902         |
| Hugo        | 8702173       | 188461        |
| Melinda     | 9444289       | 6123          |
+-----+-----+-----+
14 rows in set (0.03 sec)

mysql> _
```


Solution 12:

MySQL 8.0 Command Line Client

```
mysql> select DISTINCT t1.studentname,t1.studentsurname,t1.studentrollno
-> from studentbasicinformation t1,SubjectScholarshipInformation t2
-> where t1.optedforscholarship="yes" and t1.studentrollno NOT IN (select t2.studentrollno from SubjectScholarshipInformation);
```

studentname	studentsurname	studentrollno
Melinda	Wright	9444289
Mary	Redfield	8209187
Renee	Sanchez	6784727
Linda	Sanders	5535741
Jason	Garcia	4316976
Maria	Johnston	1964390
Sara	Baumbach	1152857

7 rows in set (0.00 sec)

mysql>

Solution 13:

MySQL 8.0 Command Line Client

```
mysql> delimiter //
mysql> create procedure findBalAmn(IN RollNo INT, OUT Bal INT)
-> BEGIN
-> select AmountBalance from StudentAdmissionPaymentDetails
-> where StudentRollNo=RollNo;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> delimiter ;
mysql> call findBalAmn(9444289,@Bal);
```

AmountBalance
6123

1 row in set (0.02 sec)

Query OK, 0 rows affected (0.02 sec)

mysql>

Solution 14:

```
mysql> select t1.studentName,t1.studentsurname  
-> from StudentBasicInformation t1, StudentSubjectInformation t2  
-> order by t2.studentmarkspercentage DESC  
-> LIMIT 5;
```

studentName	studentsurname
Maria	Johnston
Sara	Baumbach
Jordan	Graven
Hubert	Harvey
Andrew	Buchanan

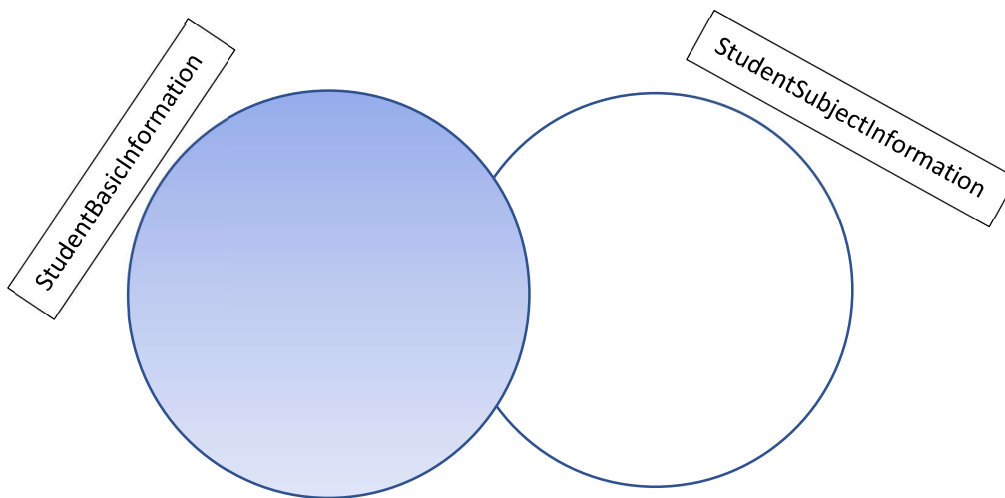
```
5 rows in set (0.04 sec)
```

Question 15: 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)

Solution:

- Left Join

Consider a scenario where we want to retrieve all the information of a student who has failed in the examination. Then we retrieve all information from left table (StudentBasicInformation) and only a columns' information (IsPassed) from Right table (StudentSubjectInformation).



```
SELECT StudentName,StudentSurname
FROM StudentBasicInformation
LEFT JOIN StudentSubjectInformation
ON StudentBasicInformation.StudentRollNo= StudentSubjectInformation.StudentRollNo
where StudentSubjectInformation.isPassed='no';
```

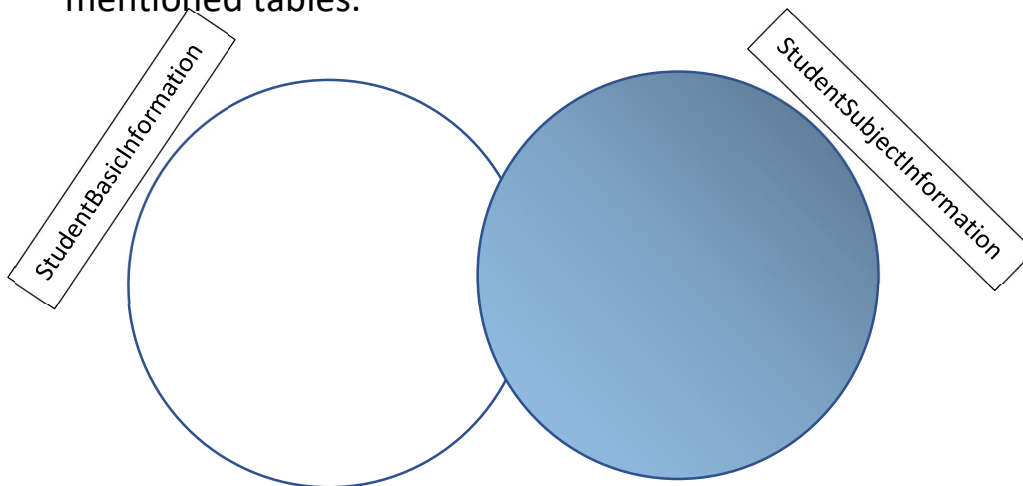
Output:

```
MySQL 8.0 Command Line Client
mysql> SELECT StudentName,StudentSurname
-> FROM StudentBasicInformation
-> LEFT JOIN StudentSubjectInformation
-> ON StudentBasicInformation.StudentRollNo= StudentSubjectInformation.StudentRollNo
-> where StudentSubjectInformation.isPassed='no';
+-----+-----+
| StudentName | StudentSurname |
+-----+-----+
| Alan        | Grollimund     |
| Richard     | Mcconkey       |
| Hugo        | Barre          |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

- Right Join

Consider a scenario where we have to find the average percentage of the class of all the students who opted for scholarship. Then we perform right join on the two above mentioned tables.

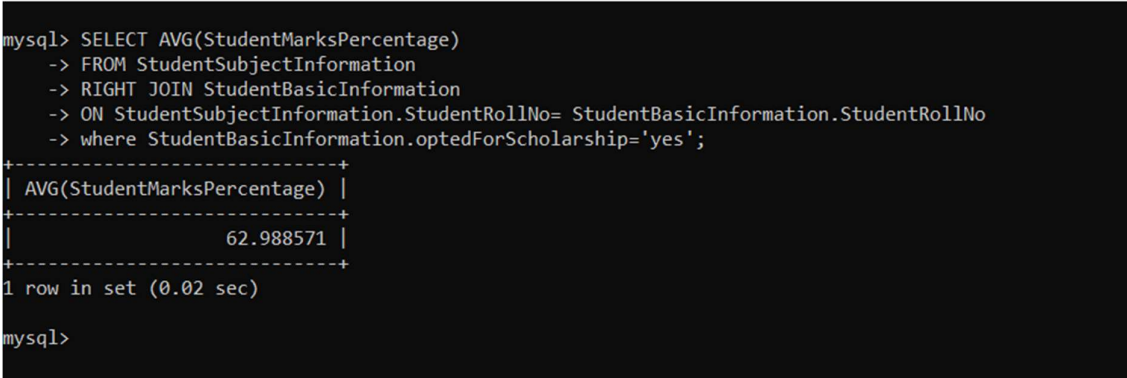



```

SELECT AVG(StudentMarksPercentage)
FROM StudentSubjectInformation
RIGHT JOIN StudentBasicInformation
ON StudentSubjectInformation.StudentRollNo= StudentBasicInformation.StudentRollNo
where StudentBasicInformation.optedForScholarship='yes';

```

OUTPUT:



```

MySQL 8.0 Command Line Client

mysql> SELECT AVG(StudentMarksPercentage)
-> FROM StudentSubjectInformation
-> RIGHT JOIN StudentBasicInformation
-> ON StudentSubjectInformation.StudentRollNo= StudentBasicInformation.StudentRollNo
-> where StudentBasicInformation.optedForScholarship='yes';
+-----+
| AVG(StudentMarksPercentage) |
+-----+
|                62.988571    |
+-----+
1 row in set (0.02 sec)

mysql>

```

Question 16: Mention the differences between the delete, drop and truncate commands.

Solution:

Delete	Drop	Truncate
It comes under DML.	It comes under DDL.	It comes under DDL.
Used to delete one or more tuples of a table.	It is used to delete the entire structure of the table.	It is used to delete all the tuples of a table.
Where clause can be used.	Where clause can't be used.	Where clause can't be used.

Question 17: 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 each scholarship's category.

Solution:

Select scholarshipcategory,count(scholarshipcategory) from
SubjectScholarshipInformation group by scholarshipcategory;

```
MySQL 8.0 Command Line Client

mysql> select scholarshipcategory,count(scholarshipcategory) from SubjectScholarshipInformation
-> group by scholarshipcategory;
+-----+-----+
| scholarshipcategory | count(scholarshipcategory) |
+-----+-----+
| Private             | 9                           |
| Government          | 2                           |
| Semi-Government     | 3                           |
| Foreign             | 1                           |
+-----+-----+
4 rows in set (0.00 sec)

mysql> _
```

Question 18: Along with the assignment no. 17 try to retrieve the maximum used scholarship category.

Solution:

Select scholarshipcategory,count(scholarshipcategory) from
SubjectScholarshipInformation group by scholarshipcategory LIMIT 1;

```
mysql> Select scholarshipcategory,count(scholarshipcategory) from SubjectScholarshipInformation group by scholarshipcategory limit 1;
+-----+-----+
| scholarshipcategory | count(scholarshipcategory) |
+-----+-----+
| Private             | 9                           |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Question 19: Retrieve the percentage of the students along with students detailed information who has scored the highest percentage along with availing the maximum scholarship amount

Solution:

```
select      StudentName,      StudentSurname,      T1.StudentRollNo,
StudentMarksPercentage, ScholarshipAmount

from StudentBasicInformation T1 join StudentSubjectInformation T3
on T1.StudentRollNo=T3.StudentRollNo

left join SubjectScholarshipInformation T4
on T1.StudentRollNo=T4.StudentRollNo

order by StudentMarksPercentage desc,ScholarshipAmount desc

limit 1;
```

```
MySQL 8.0 Command Line Client

mysql> select StudentName, StudentSurname, T1.StudentRollNo, StudentMarksPercentage, ScholarshipAmount
-> from StudentBasicInformation T1 join StudentSubjectInformation T3
-> on T1.StudentRollNo=T3.StudentRollNo
-> left join SubjectScholarshipInformation T4
-> on T1.StudentRollNo=T4.StudentRollNo
-> order by StudentMarksPercentage desc,ScholarshipAmount desc
-> limit 1;
+-----+-----+-----+-----+-----+
| StudentName | StudentSurname | StudentRollNo | StudentMarksPercentage | ScholarshipAmount |
+-----+-----+-----+-----+-----+
| Renee       | Sanchez        | 6784727       | 99.40                  | 1800              |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Question 20: Difference between the Triggers, Stored Procedures, Views and Functions.

Solution:

Views: A view is actually a composition of a table in the form of a predefined SQL query. A view can contain all rows of a table or select rows from a table. A MySQL view can be created from one or many tables which depend on the written MySQL query to create a view.

Basis of Diff.	Stored Procedure	Function	Trigger
Executability	Cannot be executed automatically.	Cannot be executed automatically.	Can be executed automatically.
Calling	Can't be called from functions but can be called from triggers.	Can be called from stored procedures or triggers.	Can't be called from stored procedure or triggers.
Parameters	Can accept any type of parameter and also can have OUT parameter.	Can accept any type of parameter but don't have an OUT parameter.	Can't accept a parameter.
Return	May or may not a value on execution.	Function must return a value.	Can never return a value.



VISHNU SINGH SENGAR

INT 653
Dev Intern