

1.)

select STDNO, STDFIRSTNAME, STDLASTNAME

from univdb.student

where STDMAJOR not in 'FIN' and STDGPA > 3.4;

The screenshot shows a SQL query editor with a query window and a results window. The query window contains the following SQL code:

```
GROUP BY e.offerno  
  
--1.  
select STDNO, STDFIRSTNAME, STDLASTNAME  
from univdb.student  
where STDMAJOR not in 'FIN' and STDGPA > 3.4;  
  
--2.  
select offering.OFFERNO, cf.COURSENO, cf.CRSDESC  
from univdb.offering offering, univdb.course cf  
where offering.COURSENO = cf.COURSENO and  
offering.OFFTERM = 'SUMMER' and offering.OFFYEAR = '2017' and offering.facno is null;
```

The results window shows the output of the first query, displaying 4 rows of student data:

STDNO	STDFIRSTNAME	STDLASTNAME
1 234-56-7890	CANDY	KENDALL
2 567-89-0123	MARIAH	DODGE
3 876-54-3210	CRISTOPHER	COLAN
4 901-23-4567	WILLIAM	PILGRIM

2.)

select offering.OFFERNO, c.COURSENO, c.CRSDESC

from univdb.offering offering, univdb.course c

where offering.COURSENO = c.COURSENO and

offering.OFFTERM = 'SUMMER' and offering.OFFYEAR = '2017' and offering.facno is null;

The screenshot shows the SQL Developer interface. The Query Builder window displays the following SQL query:

```
GROUP BY e.offerno  
  
--1.  
select STDNO, STDFIRSTNAME, STDLASTNAME  
from univdb.student  
where STDMAJOR not in 'FIN' and STDGPA > 3.4;  
  
--2.  
select offering.OFFERNO, c.COURSENO, c.CRSDESC  
from univdb.offering offering, univdb.course c  
where offering.COURSENO = c.COURSENO and  
offering.OFFTERM = 'SUMMER' and offering.OFFYEAR = '2017' and offering.facno is null;
```

The Query Result window shows the results of the query:

OFFERNO	COURSENO	CRSDESC
1	1111 IS320	FUNDAMENTALS OF BUSINESS PROGRAMMING

3.)

select s.STDNO, s.STDFIRSTNAME, s.STDLASTNAME, s.STDMAJOR, s.STDGPA, enroll.ENRGRADE

from univdb.student s, univdb.enrollment enroll, univdb.offering offering

where s.STDNO = enroll.STDNO and offering.OFFERNO = enroll.OFFERNO and

offering.COURSENO = 'FIN300' and offering.OFFTERM = 'WINTER' and offering.OFFYEAR = '2017';

The screenshot shows the SQL Developer interface. The Query Builder window displays the following SQL query:

```
--2.  
select offering.OFFERNO, c.COURSENO, c.CRSDESC  
from univdb.offering offering, univdb.course c  
where offering.COURSENO = c.COURSENO and  
offering.OFFTERM = 'SUMMER' and offering.OFFYEAR = '2017' and offering.facno is null;  
  
-- 3.  
select s.STDNO, s.STDFIRSTNAME, s.STDLASTNAME, s.STDMAJOR, s.STDGPA, enroll.ENRGRADE  
from univdb.student s, univdb.enrollment enroll, univdb.offering offering  
where s.STDNO = enroll.STDNO and offering.OFFERNO = enroll.OFFERNO and  
offering.COURSENO = 'FIN300' and offering.OFFTERM = 'WINTER' and offering.OFFYEAR = '2017';  
  
--4.
```

The Query Result window shows the results of the query:

STDNO	STDFIRSTNAME	STDLASTNAME	STDMAJOR	STDGPA	ENRGRADE
1	123-45-6789	HOMER WELLS	IS	3	3.2
2	124-56-7890	BOB NORBERT	FIN	2.7	2.7

4.)

select offering.COURSENO, COUNT(offering.COURSENO)

from univdb.student s, univdb.enrollment enrollment, univdb.offering offering

where s.STDNO = enrollment.STDNO and enrollment.OFFERNO = offering.OFFERNO and

s.STDCLASS = 'SR' and (s.STDMAJOR = 'FIN' OR s.STDMAJOR = 'IS')

GROUP BY offering.COURSENO

ORDER BY COUNT(offering.COURSENO) DESC;

The screenshot shows a SQL query editor with a worksheet titled 'Query Builder'. The query is as follows:

```
--4.
select offering.COURSENO, COUNT(offering.COURSENO)
from univdb.student s, univdb.enrollment enrollment, univdb.offering offering
where s.STDNO = enroll.STDNO and offering.OFFERNO = enroll.OFFERNO and
offering.COURSENO = 'FIN300' and offering.OFFTERM = 'WINTER' and offering.OFFYEAR = '2017';

--5.
select ustudent.STDMAJOR, max(ustudent.STDGPA)
```

Below the query editor, the 'Query Result' tab is active, showing the results of the query. The results are displayed in a table with 2 columns: COURSENO and COUNT(OFFERING.COURSENO). The table contains 4 rows of data.

COURSENO	COUNT(OFFERING.COURSENO)
1 IS320	4
2 IS480	3
3 IS460	2
4 FIN480	2

5.)

select s.STDMAJOR, max(s.STDGPA)

from univdb.student s

where s.STDCLASS = 'JR'

GROUP BY s.STDMAJOR

HAVING avg(s.stdgpa) > 3.05;

The screenshot shows a SQL IDE interface with multiple tabs. The active tab is 'Aashish\_Assignment2.sql'. The 'Query Builder' pane displays the following SQL query:

```
from univdb.student s, univdb.enrollment enrollment, univdb.offering offering
where s.STDNO = enrollment.STDNO and enrollment.OFFERNO = offering.OFFERNO and
s.STDCLASS = 'SR' and (s.STDMAJOR = 'FIN' OR s.STDMAJOR = 'IS')
GROUP BY offering.COURSENO
ORDER BY COUNT(offering.COURSENO) DESC;

--5.
select s.STDMAJOR, max(s.STDGPA)
from univdb.student s
where s.STDCLASS = 'JR'
GROUP BY s.STDMAJOR
HAVING avg(s.stdgpa) > 3.05;

--6.
```

The 'Query Result' pane shows the results of the query, indicating that 2 rows were fetched in 0.055 seconds. The results are as follows:

	STDMAJOR	MAX(S.STDGPA)
1	IS	3.6
2	ACCT	3.5

6.)

```
SELECT f1.facfirstname, f1.faclastname, offering1.courseno, f2.facfirstname, f2.faclastname
FROM univdb.offering offering1, univdb.offering offering2, univdb.faculty f1, univdb.faculty f2
WHERE f1.facno = offering1.facno and
f1.facsupervisor = offering2.facno and f2.facno = f1.facsupervisor and
offering1.courseno = offering2.courseno and
offering1.OFFYEAR = 2017 and offering2.OFFYEAR = 2017;
```

The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
from univdb.student s
where s.STDCCLASS = 'JR'
GROUP BY s.STDMAJOR
HAVING avg(s.stdgpa) > 3.05;

--6.

SELECT f1.facfirstname, f1.faclastname, offering1.courseno, f2.facfirstname, f2.faclastname
FROM univdb.offering offering1, univdb.offering offering2, univdb.faculty f1, univdb.faculty f2
WHERE f1.facno = offering1.facno and
f1.facsupervisor = offering2.facno and f2.facno = f1.facsupervisor and
offering1.courseno = offering2.courseno and
offering1.OFFYEAR = 2017 and offering2.OFFYEAR = 2017;

select * from univdb.faculty;
select * from univdb.offering;
```

The results pane shows the output of the query, which is a list of faculty and offering records. The results are as follows:

FACFIRSTNAME	FACLASTNAME	COURSENO	FACFIRSTNAME_1	FACLASTNAME_1
1 LEONARD	FIBON	IS320	VICTORIA	EMMANUEL
2 LEONARD	VINCE	IS320	LEONARD	FIBON

7.)

select s.STDMAJOR, max(s.STDGPA)

from univdb.student s

where s.STDCLASS = 'JR'

GROUP BY s.STDMAJOR

HAVING avg(s.STDGPA) >= 3.3;

The screenshot shows a SQL query editor with a query window and a results window. The query window contains the following SQL code:

```
--7.
select s.STDMAJOR, max(s.STDGPA)
from univdb.student s
where s.STDCLASS = 'JR'
GROUP BY s.STDMAJOR
HAVING avg(s.STDGPA) >= 3.3;

--8.
--List the course number, offering number, number of students, and
--average enrollment grade of IS course offerings in which more than 5 students are enrolled
```

The results window shows the following data:

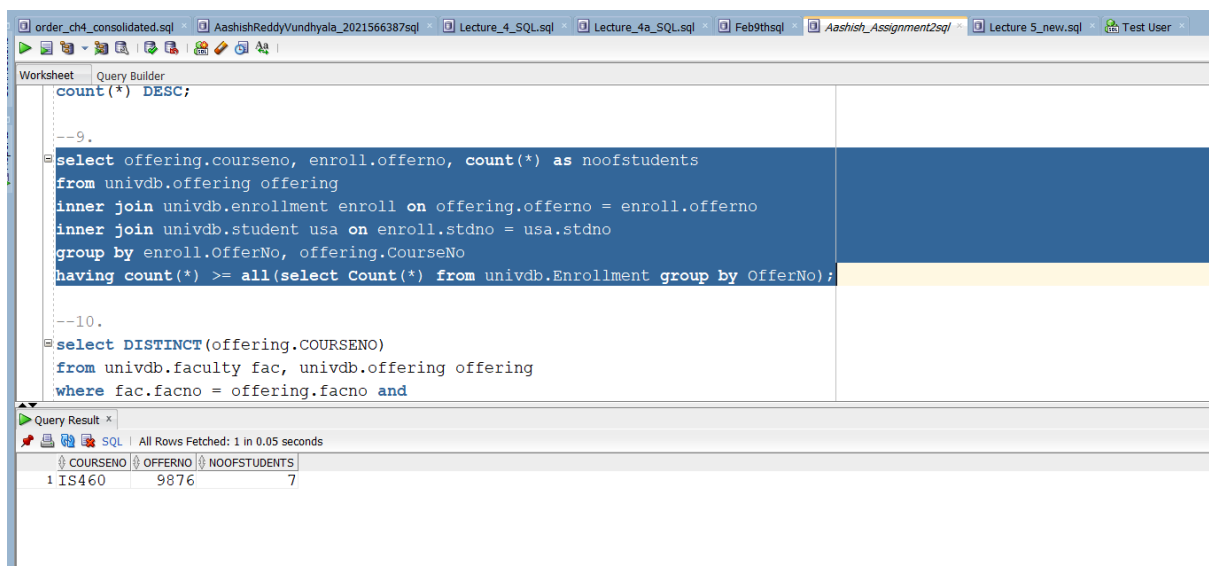
STDMAJOR	MAX(S.STDGPA)
1 IS	3.6
2 ACCT	3.5

```
select o.courseno, e.offerno, count(*) as noofstds, avg(e.enrgrade)
from univdb.offering o, univdb.enrollment e
where o.offerno = e.offerno
and o.CourseNo LIKE 'IS%'
GROUP BY o.CourseNo, e.OfferNo
HAVING count(*) > 5
ORDER BY o.CourseNo,
count(*) DESC;
```

[illegible]

9.)

```
select offering.courseno, enroll.offerNo, count(*) as noofstudents
from univdb.offering offering
inner join univdb.enrollment enroll on offering.offerNo = enroll.offerNo
inner join univdb.student usa on enroll.stdno = usa.stdno
group by enroll.OfferNo, offering.CourseNo
having count(*) >= all(select Count(*) from univdb.Enrollment group by OfferNo);
```



The screenshot shows a SQL query editor with a query window and a results window. The query window contains the following SQL code:

```
count(*) DESC;
--9.
select offering.courseno, enroll.offerNo, count(*) as noofstudents
from univdb.offering offering
inner join univdb.enrollment enroll on offering.offerNo = enroll.offerNo
inner join univdb.student usa on enroll.stdno = usa.stdno
group by enroll.OfferNo, offering.CourseNo
having count(*) >= all(select Count(*) from univdb.Enrollment group by OfferNo);
--10.
select DISTINCT(offering.COURSENO)
from univdb.faculty fac, univdb.offering offering
where fac.facno = offering.facno and
```

The results window shows the following output:

COURSENO	OFFERNO	NOOFSTUDENTS
1 IS460	9876	7



10.)

```
select DISTINCT(offering.COURSENO)
from univdb.faculty f, univdb.offering offering
where f.facno = offering.facno and
not (f.facfirstname = 'LEONARD' and f.faclastname = 'FIBON')
ORDER BY offering.COURSENO DESC;
```

The screenshot shows a SQL query builder interface with a 'Query Builder' tab. The query is as follows:

```
--10.
select DISTINCT(offering.COURSENO)
from univdb.faculty f, univdb.offering offering
where f.facno = offering.facno and
not (f.facfirstname = 'LEONARD' and f.faclastname = 'FIBON')
ORDER BY offering.COURSENO DESC;

--11.
select ustudent.STDNO, ustudent.STDFIRSTNAME, ustudent.STDLASTNAME
from univdb.student ustudent
```

Below the query, the 'Query Result' tab is active, showing the results of the first query. The results are displayed in a table with the following data:

COURSENO
1 IS480
2 IS320
3 FIN480
4 FIN450
5 FIN300

11.)

select s.STDNO, s.STDFIRSTNAME, s.STDLASTNAME

from univdb.student s

where not s.STDMAJOR = 'FIN'

ORDER BY s.stdfirstname, s.stdlastname ASC;

The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
ORDER BY offering.COURSENO DESC;

--11.
select s.STDNO, s.STDFIRSTNAME, s.STDLASTNAME
from univdb.student s
where not s.STDMAJOR = 'FIN'
ORDER BY s.stdfirstname, s.stdlastname ASC;

--12.
-- SUBTRACT
select offering.courseno
from univdb.student ustu, univdb.enrollment enroll, univdb.offering offering
where ustu.stdno = enroll.stdno and enroll.offerno = offering.offerno
and ustu.STDCCLASS = 'JR'
```

The results pane shows the output of the query, displaying 8 rows of student information. The columns are STDNO, STDFIRSTNAME, and STDLASTNAME. The data is as follows:

STDNO	STDFIRSTNAME	STDLASTNAME
1 234-56-7890	CANDY	KENDALL
2 876-54-3210	CRISTOPHER	COLAN
3 123-45-6789	HOMER	WELLS
4 890-12-3456	LUKE	BRAZZI
5 567-89-0123	MARIAH	DODGE
6 678-90-1234	TESS	DODGE
7 345-67-8901	WALLY	KENDALL
8 901-23-4567	WILLIAM	PILGRIM

12.)

```
select offering.courseno
from univdb.student s, univdb.enrollment enroll, univdb.offering offering
where s.stdno = enroll.stdno and enroll.offerno = offering.offerno
and s.STDCLASS = 'JR'
```

MINUS

```
select offering.courseno
from univdb.student s, univdb.enrollment enrollment, univdb.offering offering
where s.stdno = enrollment.stdno and enrollment.offerno = offering.offerno
and not student.STDCLASS = 'JR';
```

The screenshot shows a SQL query editor with multiple tabs. The active tab is 'Aashish\_Assignment2sql'. The query is as follows:

```
ORDER BY s.stdfirstname, s.stdlastname ASC;

--12.
-- SUBTRACT
select offering.courseno
from univdb.student s, univdb.enrollment enroll, univdb.offering offering
where s.stdno = enroll.stdno and enroll.offerno = offering.offerno
and s.STDCLASS = 'JR'
MINUS
select offering.courseno
from univdb.student s, univdb.enrollment enrollment, univdb.offering offering
where s.stdno = enrollment.stdno and enrollment.offerno = offering.offerno
and not s.STDCLASS = 'JR';

--13.
select fac.FACNO, fac.FACFIRSTNAME, fac.FACLASTNAME, fac.FACSALARY
```

The query results are displayed in a table with the following data:

COURSENO
1 FIN450

13.)

```
select f.facno, f.facfirstname, f.faclastname, f.facsalary, f.facdept
from univdb.faculty f
join univdb.faculty f1
on f.facsalary = f1.facsalary
where f.facsalary in
(select max(f1.facsalary) from univdb.faculty f1 group by f1.facdept);
```

The screenshot shows a SQL IDE with a query editor and a results pane. The query editor contains the following SQL code:

```
--13.
select f.facno, f.facfirstname, f.faclastname, f.facsalary, f.facdept
from univdb.faculty f
join univdb.faculty f1
on f.facsalary = f1.facsalary
where f.facsalary in
(select max(f1.facsalary) from univdb.faculty f1 group by f1.facdept);

drop table Student_tmp_assign3q1;
create table Student_tmp_assign3q1 as select * from univdb.Student;

INSERT INTO Student_tmp_assign3q1 (StdNo, StdFirstName, StdLastName, StdCity,
StdState, StdClass, StdZip, StdMajor, StdGPA)
VALUES ('999999999', 'JOE', 'STUDENT', 'SEATAC',
'WA', 'FR', '98042-1121', 'IS', 3.25);
```

The results pane shows the following data:

	FACNO	FACFIRSTNAME	FACLASTNAME	FACSALARY	FACDEPT
1	543-21-0987	VICTORIA	EMMANUEL	120000	MS
2	987-65-4321	JULIA	MILLS	75000	FIN