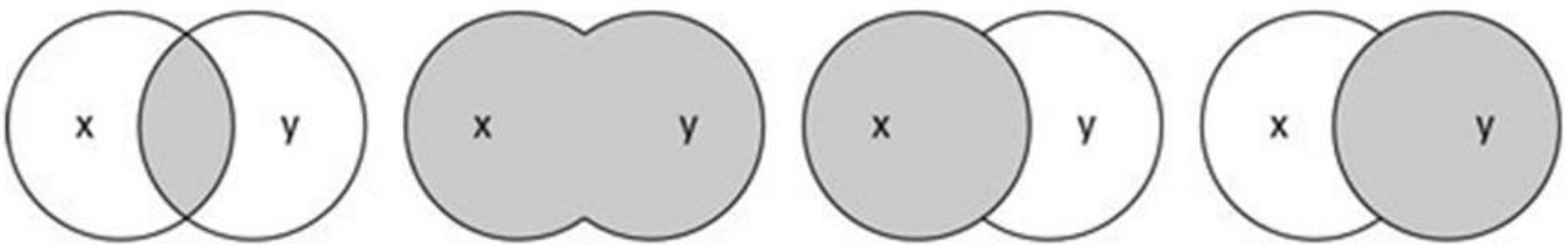
SQL: Queries with Joins

- Chapter 4 in Murach text
- SQL for retrieving data from multiple tables (JOIN … ON)

Joining Multiple Tables



Inner Join

Outer Join

Left Outer Join

Right Outer Join

A Join is a SQL statement that combines data from two or more tables using the key relationship to link records.

Inner Join Operations

The explicit syntax for an inner join

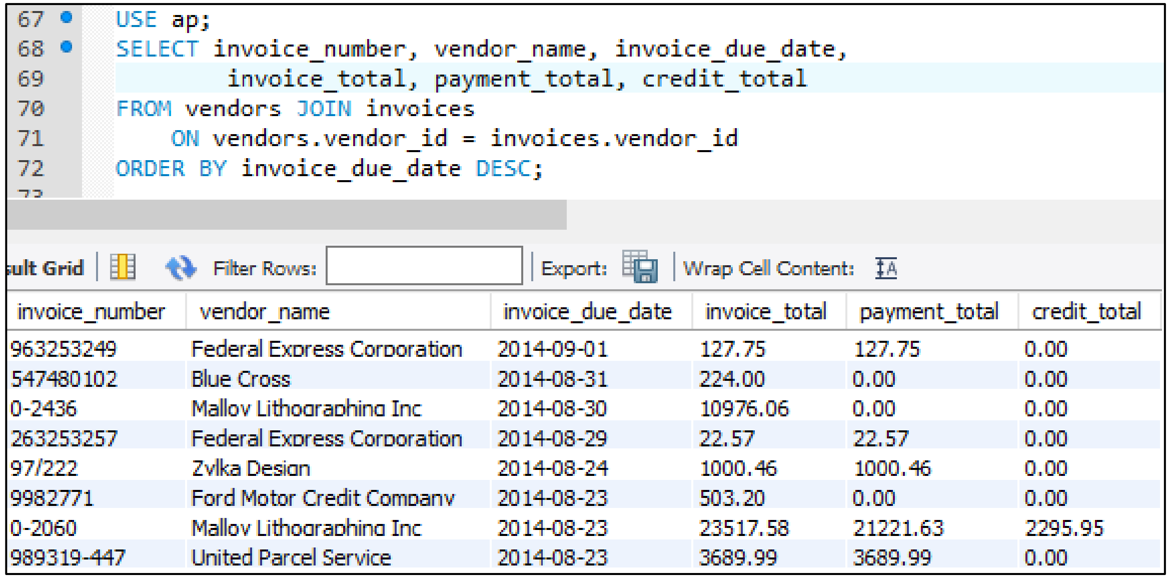
```
SELECT select_list
FROM table_1
    [INNER] JOIN table_2
        ON join_condition_1
   [[INNER] JOIN table_3
        ON join_condition_2]...
```

An inner join of the Vendors and Invoices tables

```
SELECT invoice_number, vendor_name
FROM vendors INNER JOIN invoices
    ON vendors.vendor_id = invoices.vendor_id
ORDER BY invoice_number
```

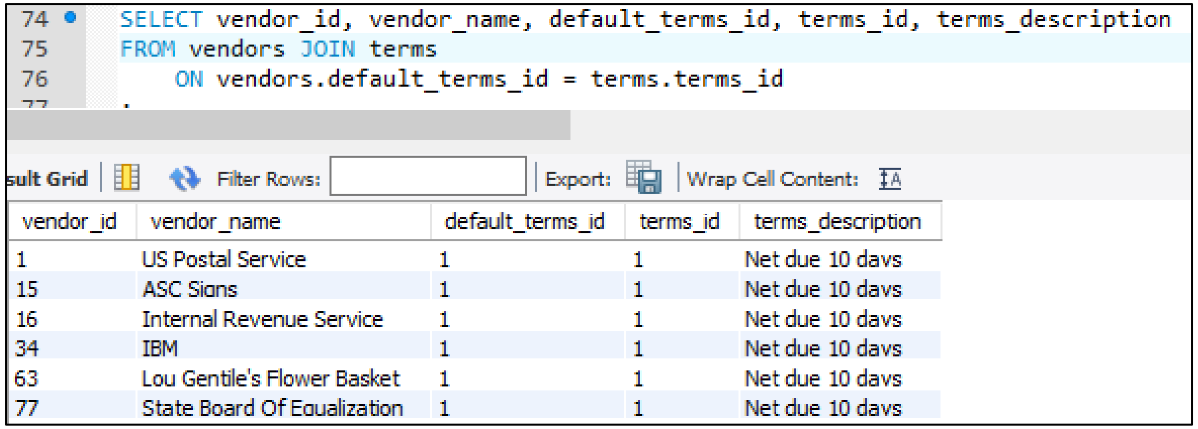
Inner Join

Keywords: SELECT, FROM, JOIN, ON, [WHERE, ORDER BY, LIMIT]



```sql
USE ap;
SELECT invoice_number, vendor_name, invoice_due_date,
        invoice_total, payment_total, credit_total
FROM vendors JOIN invoices
    ON vendors.vendor_id = invoices.vendor_id
ORDER BY invoice_due_date DESC;
```

invoice_number	vendor_name	invoice_due_date	invoice_total	payment_total	credit_total
963253249	Federal Express Corporation	2014-09-01	127.75	127.75	0.00
547480102	Blue Cross	2014-08-31	224.00	0.00	0.00
0-2436	Malloy Lithographing Inc	2014-08-30	10976.06	0.00	0.00
263253257	Federal Express Corporation	2014-08-29	22.57	22.57	0.00
97/222	Zylka Design	2014-08-24	1000.46	1000.46	0.00
9982771	Ford Motor Credit Company	2014-08-23	503.20	0.00	0.00
0-2060	Malloy Lithographing Inc	2014-08-23	23517.58	21221.63	2295.95
989319-447	United Parcel Service	2014-08-23	3689.99	3689.99	0.00

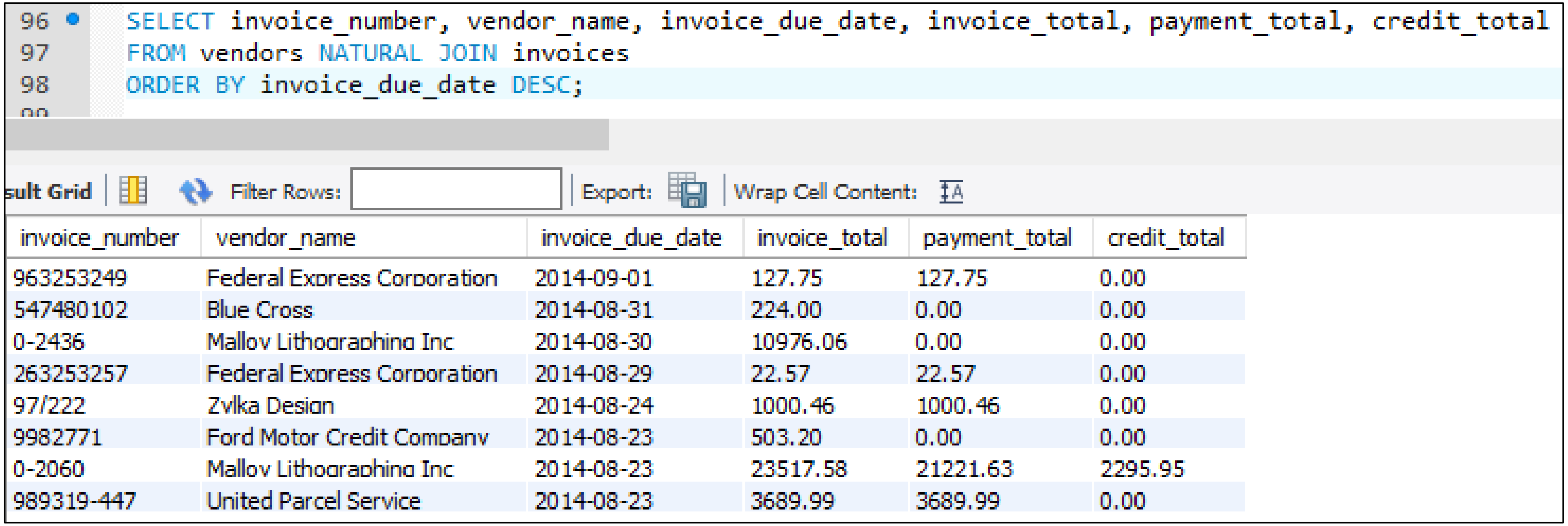


```sql
SELECT vendor_id, vendor_name, default_terms_id, terms_id, terms_description
FROM vendors JOIN terms
    ON vendors.default_terms_id = terms.terms_id
```

vendor_id	vendor_name	default_terms_id	terms_id	terms_description
1	US Postal Service	1	1	Net due 10 days
15	ASC Signs	1	1	Net due 10 days
16	Internal Revenue Service	1	1	Net due 10 days
34	IBM	1	1	Net due 10 days
63	Lou Gentile's Flower Basket	1	1	Net due 10 days
77	State Board Of Equalization	1	1	Net due 10 days

Natural Join

Keywords: SELECT, FROM, NATURAL JOIN, [WHERE, ORDER BY, LIMIT]

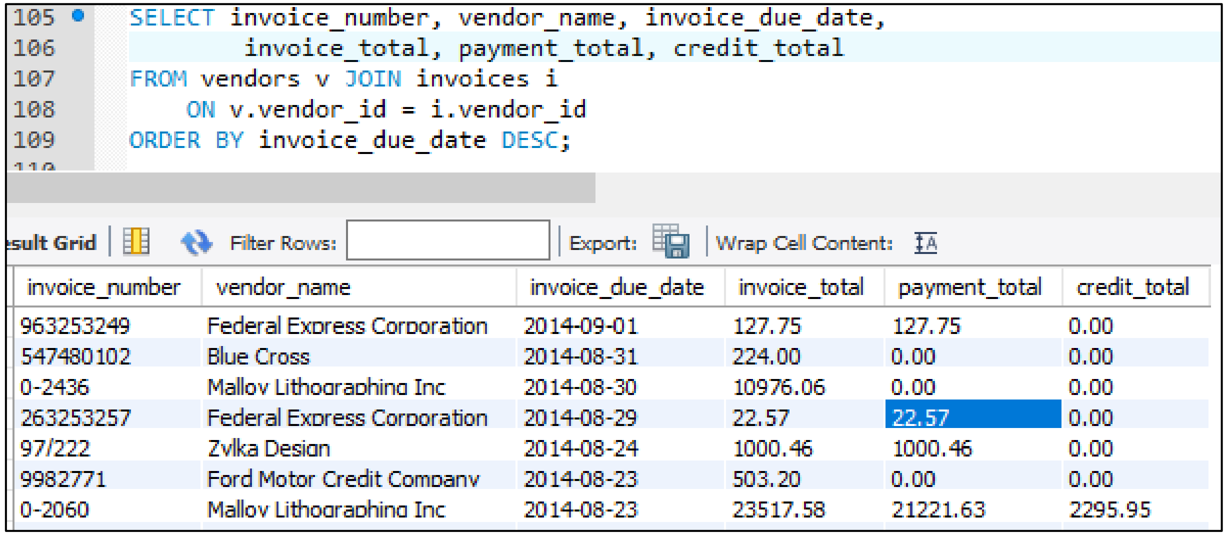


```
SELECT invoice_number, vendor_name, invoice_due_date, invoice_total, payment_total, credit_total
FROM vendors NATURAL JOIN invoices
ORDER BY invoice_due_date DESC;
```

invoice_number	vendor_name	invoice_due_date	invoice_total	payment_total	credit_total
963253249	Federal Express Corporation	2014-09-01	127.75	127.75	0.00
547480102	Blue Cross	2014-08-31	224.00	0.00	0.00
0-2436	Malloy Lithographing Inc	2014-08-30	10976.06	0.00	0.00
263253257	Federal Express Corporation	2014-08-29	22.57	22.57	0.00
97/222	Zylka Design	2014-08-24	1000.46	1000.46	0.00
9982771	Ford Motor Credit Company	2014-08-23	503.20	0.00	0.00
0-2060	Malloy Lithographing Inc	2014-08-23	23517.58	21221.63	2295.95
989319-447	United Parcel Service	2014-08-23	3689.99	3689.99	0.00

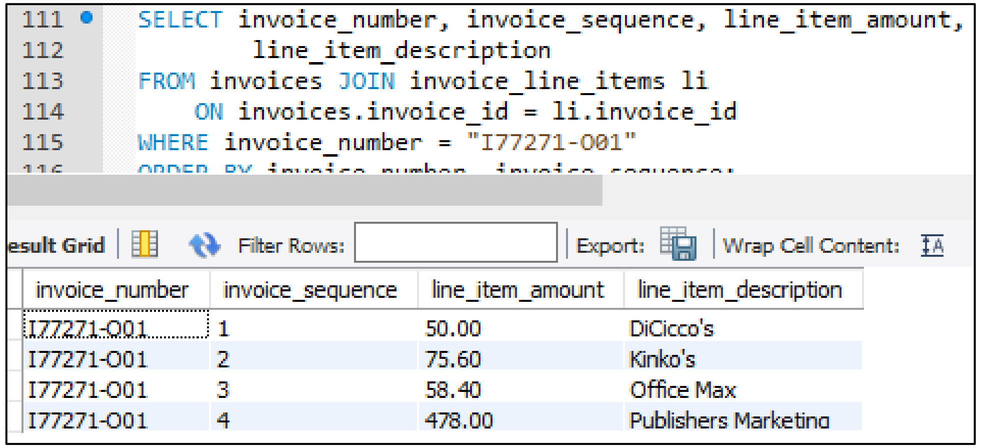
Inner Join with Alias(es)

Keywords: SELECT, FROM, JOIN, ON, [WHERE, ORDER BY, LIMIT]



```
SELECT invoice_number, vendor_name, invoice_due_date,
        invoice_total, payment_total, credit_total
FROM vendors v JOIN invoices i
    ON v.vendor_id = i.vendor_id
ORDER BY invoice_due_date DESC;
```

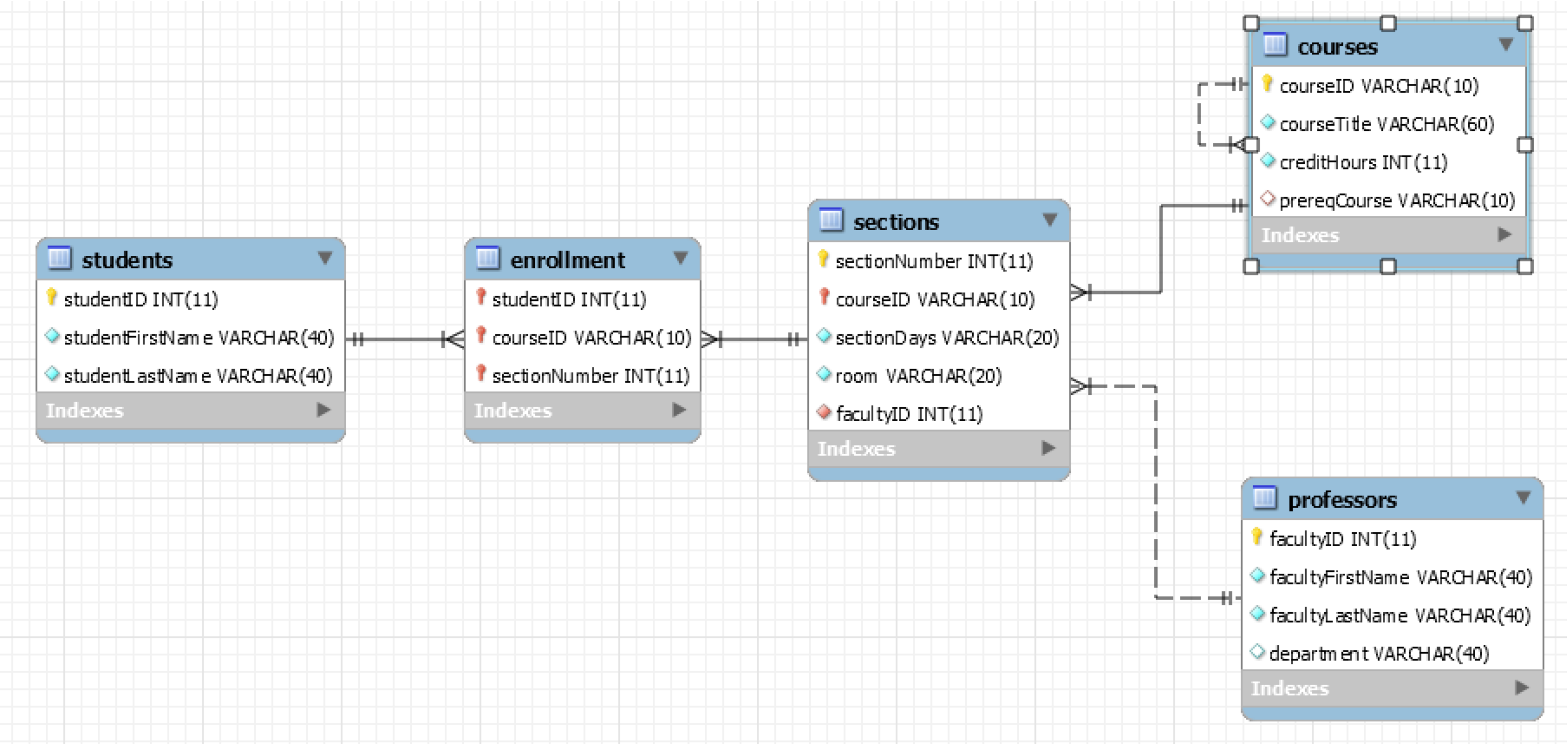
invoice_number	vendor_name	invoice_due_date	invoice_total	payment_total	credit_total
963253249	Federal Express Corporation	2014-09-01	127.75	127.75	0.00
547480102	Blue Cross	2014-08-31	224.00	0.00	0.00
0-2436	Mallov Lithographing Inc	2014-08-30	10976.06	0.00	0.00
263253257	Federal Express Corporation	2014-08-29	22.57	22.57	0.00
97/222	Zylka Design	2014-08-24	1000.46	1000.46	0.00
9982771	Ford Motor Credit Company	2014-08-23	503.20	0.00	0.00
0-2060	Mallov Lithographing Inc	2014-08-23	23517.58	21221.63	2295.95



```
SELECT invoice_number, invoice_sequence, line_item_amount,
        line_item_description
FROM invoices JOIN invoice_line_items li
    ON invoices.invoice_id = li.invoice_id
WHERE invoice_number = "I77271-O01"
ORDER BY invoice_number, invoice_sequence;
```

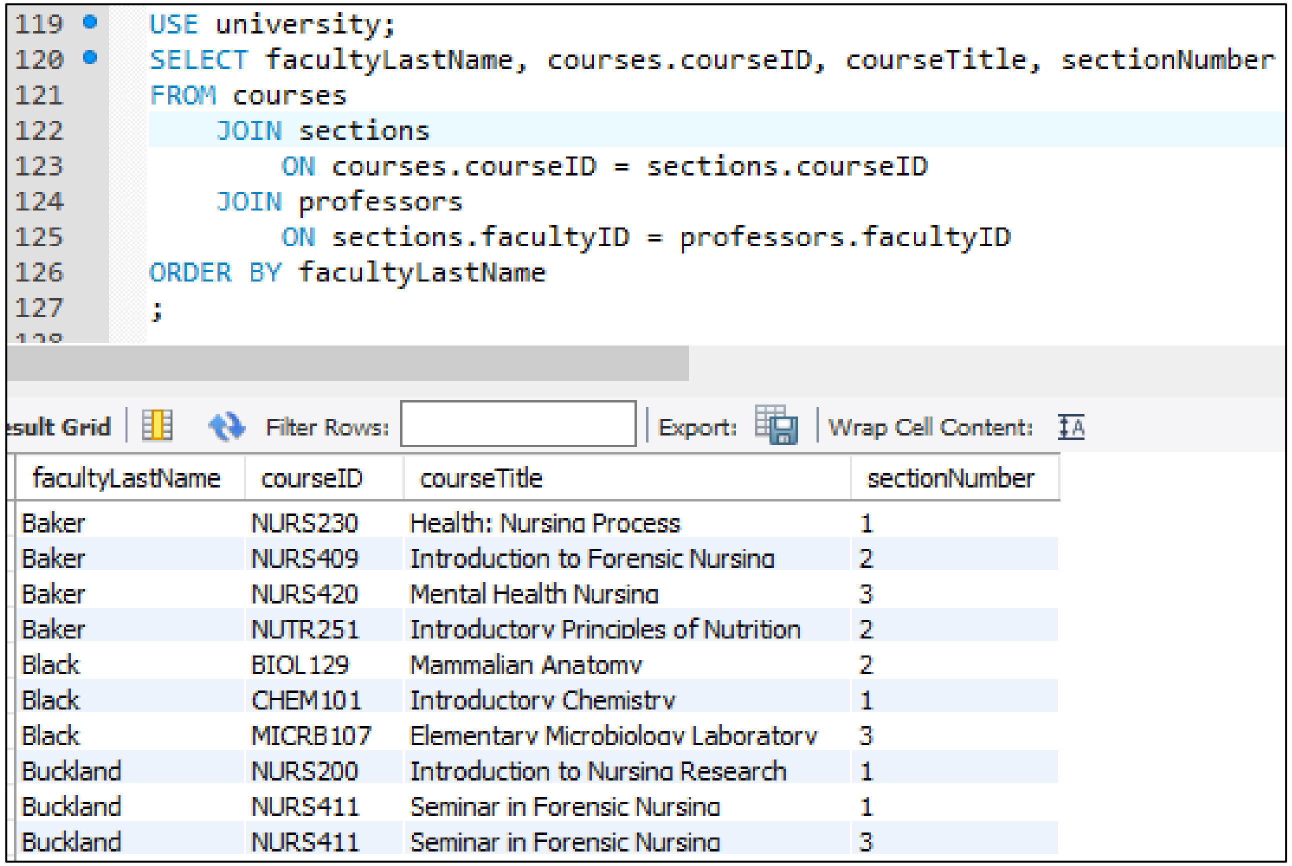
invoice_number	invoice_sequence	line_item_amount	line_item_description
I77271-O01	1	50.00	DiCicco's
I77271-O01	2	75.60	Kinko's
I77271-O01	3	58.40	Office Max
I77271-O01	4	478.00	Publishers Marketing

University DB Example



Inner Join with Multiple Tables

Keywords: SELECT, FROM, JOIN, ON, [WHERE, ORDER BY, LIMIT]

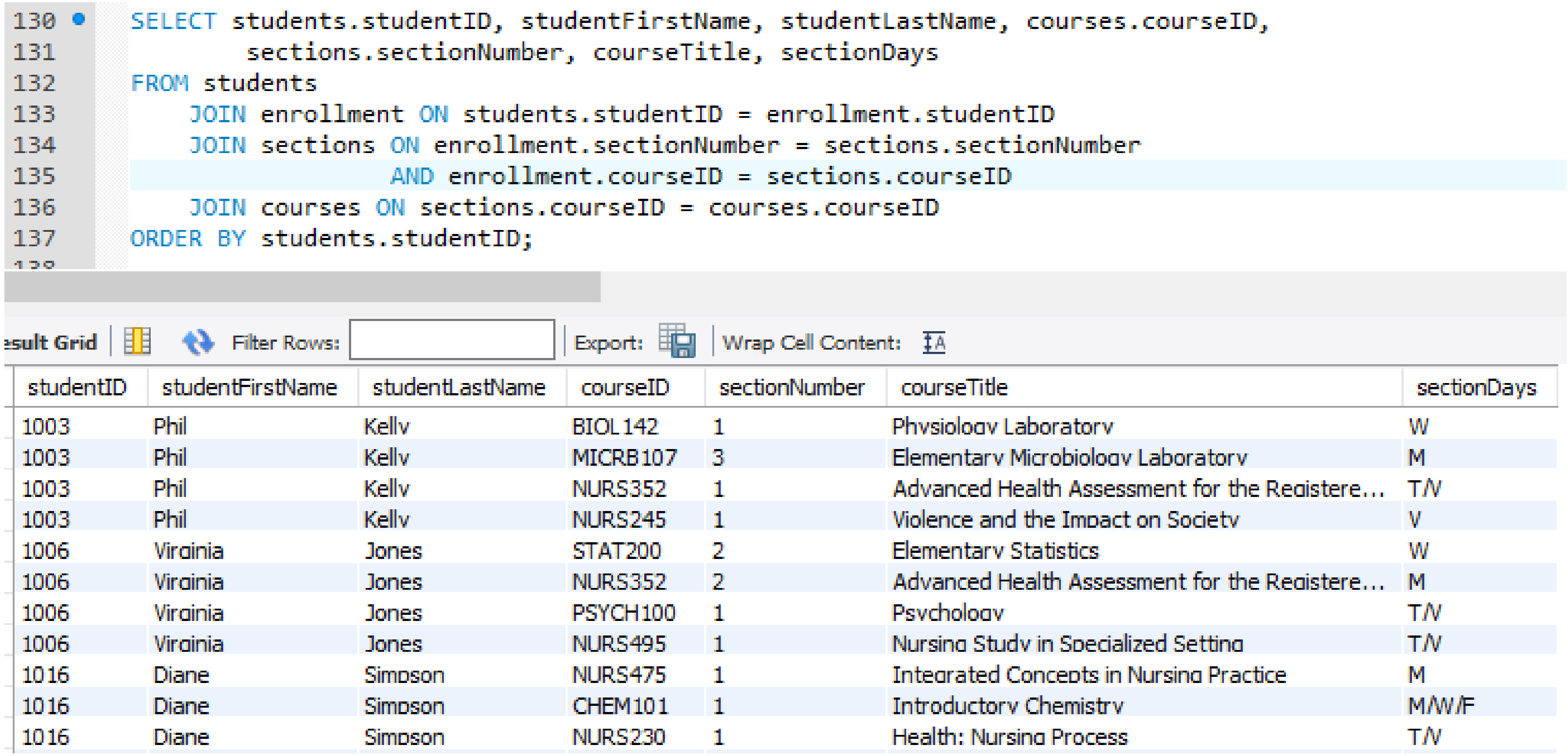


```
USE university;
SELECT facultyLastName, courses.courseID, courseTitle, sectionNumber
FROM courses
    JOIN sections
        ON courses.courseID = sections.courseID
    JOIN professors
        ON sections.facultyID = professors.facultyID
ORDER BY facultyLastName
;
```

facultyLastName	courseID	courseTitle	sectionNumber
Baker	NURS230	Health: Nursing Process	1
Baker	NURS409	Introduction to Forensic Nursing	2
Baker	NURS420	Mental Health Nursing	3
Baker	NUTR251	Introductory Principles of Nutrition	2
Black	BIOL129	Mammalian Anatomy	2
Black	CHEM101	Introductory Chemistry	1
Black	MICRB107	Elementary Microbiology Laboratory	3
Buckland	NURS200	Introduction to Nursing Research	1
Buckland	NURS411	Seminar in Forensic Nursing	1
Buckland	NURS411	Seminar in Forensic Nursing	3

Inner Join with Multiple Tables

Keywords: SELECT, FROM, JOIN, ON, [WHERE, ORDER BY, LIMIT]

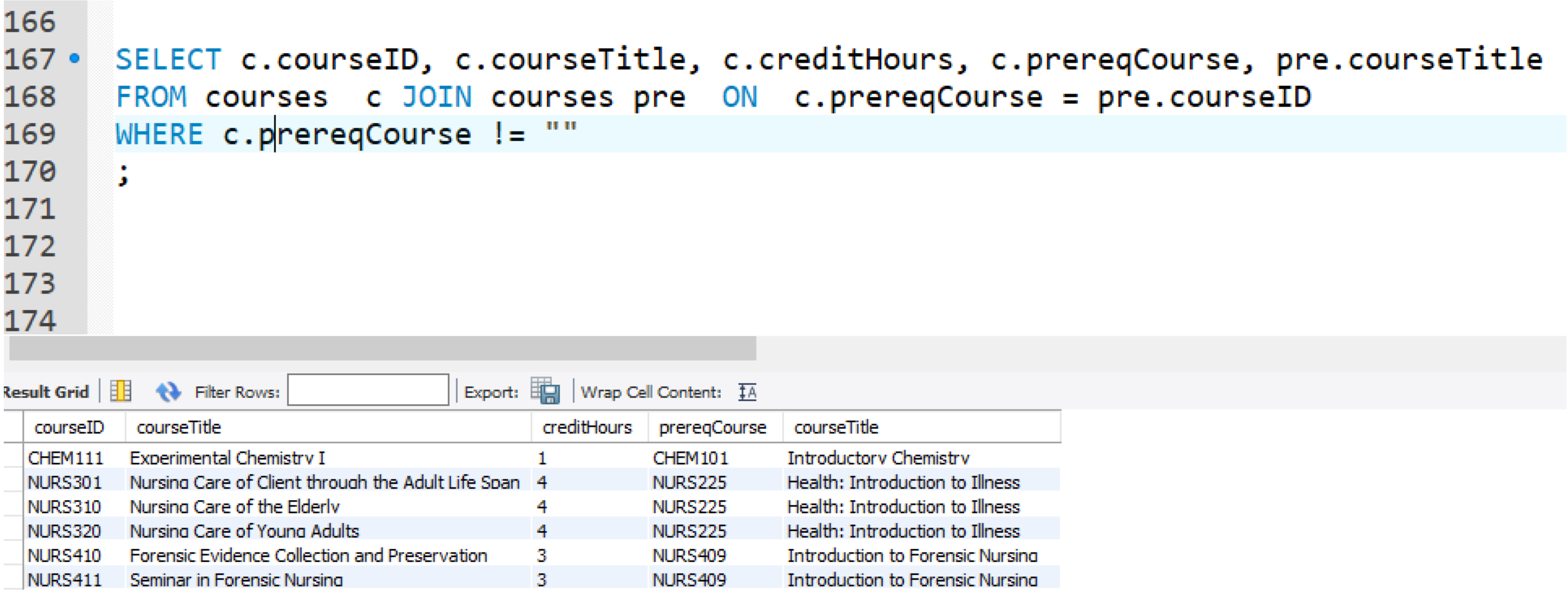


```
SELECT students.studentID, studentFirstName, studentLastName, courses.courseID,
        sections.sectionNumber, courseTitle, sectionDays
FROM students
    JOIN enrollment ON students.studentID = enrollment.studentID
    JOIN sections ON enrollment.sectionNumber = sections.sectionNumber
                  AND enrollment.courseID = sections.courseID
    JOIN courses ON sections.courseID = courses.courseID
ORDER BY students.studentID;
```

studentID	studentFirstName	studentLastName	courseID	sectionNumber	courseTitle	sectionDays
1003	Phil	Kelly	BIOL142	1	Physiology Laboratory	W
1003	Phil	Kelly	MICRB107	3	Elementary Microbiology Laboratory	M
1003	Phil	Kelly	NURS352	1	Advanced Health Assessment for the Registere...	T/W
1003	Phil	Kelly	NURS245	1	Violence and the Impact on Society	V
1006	Virginia	Jones	STAT200	2	Elementary Statistics	W
1006	Virginia	Jones	NURS352	2	Advanced Health Assessment for the Registere...	M
1006	Virginia	Jones	PSYCH100	1	Psychology	T/W
1006	Virginia	Jones	NURS495	1	Nursing Study in Specialized Setting	T/W
1016	Diane	Simpson	NURS475	1	Integrated Concepts in Nursing Practice	M
1016	Diane	Simpson	CHEM101	1	Introductory Chemistry	M/W/F
1016	Diane	Simpson	NURS230	1	Health: Nursing Process	T/W

Self-Join

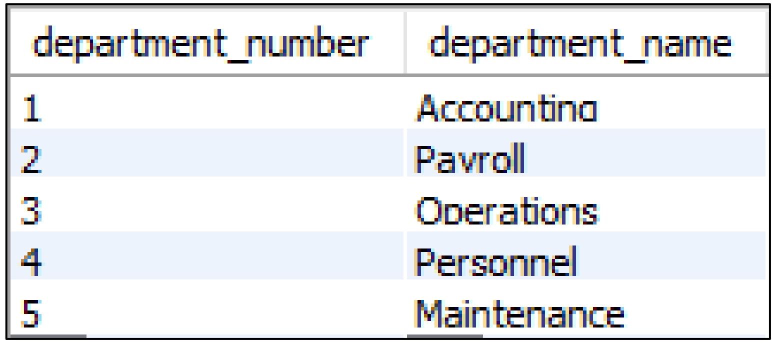
Keywords: SELECT, FROM, JOIN, ON, [WHERE, ORDER BY, LIMIT]



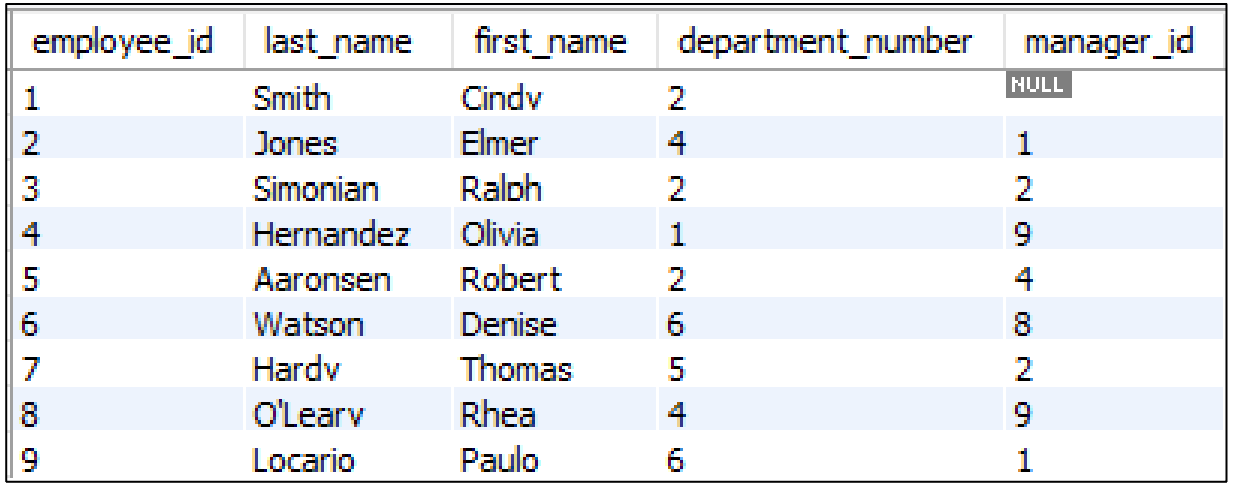
```
SELECT c.courseID, c.courseTitle, c.creditHours, c.prereqCourse, pre.courseTitle
FROM courses  c JOIN courses pre  ON  c.prereqCourse = pre.courseID
WHERE c.prereqCourse != ""
;
```

courseID	courseTitle	creditHours	prereqCourse	courseTitle
CHEM111	Experimental Chemistry I	1	CHEM101	Introductory Chemistry
NURS301	Nursing Care of Client through the Adult Life Span	4	NURS225	Health: Introduction to Illness
NURS310	Nursing Care of the Elderly	4	NURS225	Health: Introduction to Illness
NURS320	Nursing Care of Young Adults	4	NURS225	Health: Introduction to Illness
NURS410	Forensic Evidence Collection and Preservation	3	NURS409	Introduction to Forensic Nursing
NURS411	Seminar in Forensic Nursing	3	NURS409	Introduction to Forensic Nursing

Departments and Employees Tables (department_employees.sql)



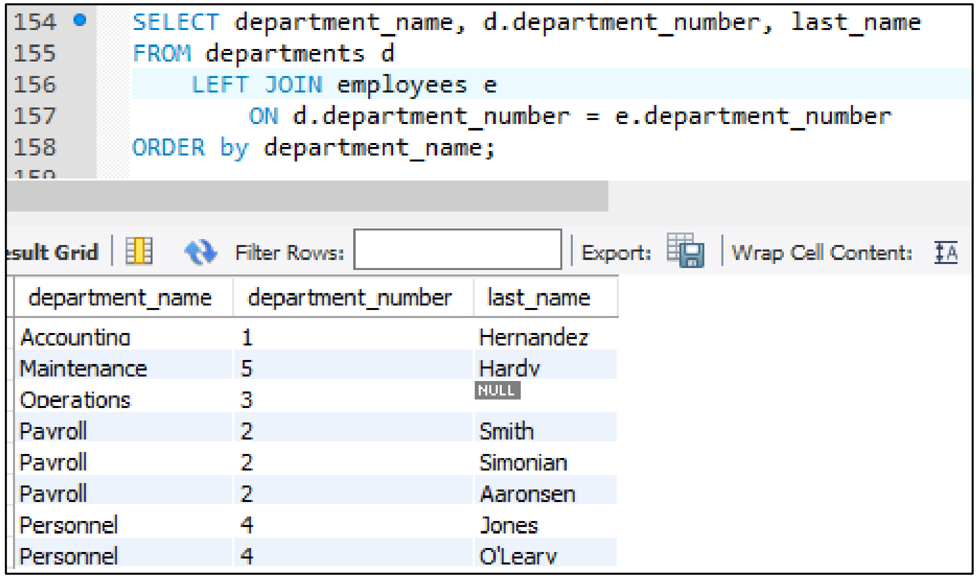
department_number	department_name
1	Accounting
2	Payroll
3	Operations
4	Personnel
5	Maintenance



employee_id	last_name	first_name	department_number	manager_id
1	Smith	Cindy	2	NULL
2	Jones	Elmer	4	1
3	Simonian	Ralph	2	2
4	Hernandez	Olivia	1	9
5	Aaronsen	Robert	2	4
6	Watson	Denise	6	8
7	Hardy	Thomas	5	2
8	O'Leary	Rhea	4	9
9	Locario	Paulo	6	1

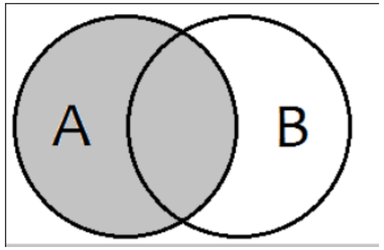
Outer Joins

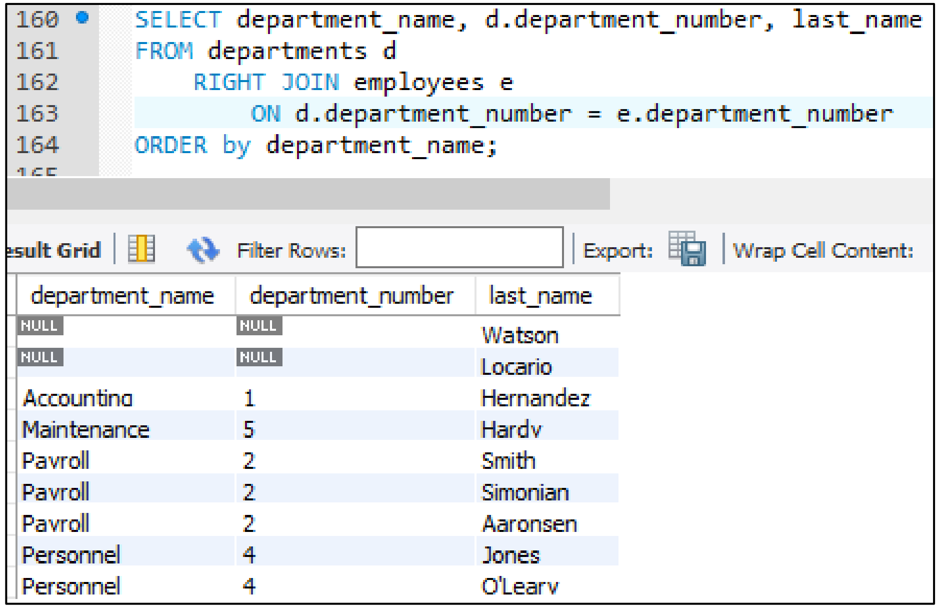
Keywords: SELECT, FROM, LEFT/RIGHT JOIN, ON, [WHERE, ORDER BY, LIMIT]



```
SELECT department_name, d.department_number, last_name
FROM departments d
    LEFT JOIN employees e
        ON d.department_number = e.department_number
ORDER by department_name;
```

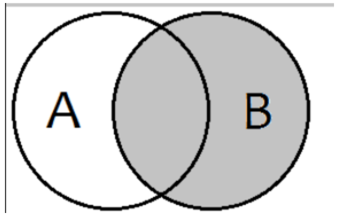
department_name	department_number	last_name
Accountina	1	Hernandez
Maintenance	5	Hardv
Ooerations	3	NULL
Pavroll	2	Smith
Pavroll	2	Simonian
Pavroll	2	Aaronsen
Personnel	4	Jones
Personnel	4	O'Learv





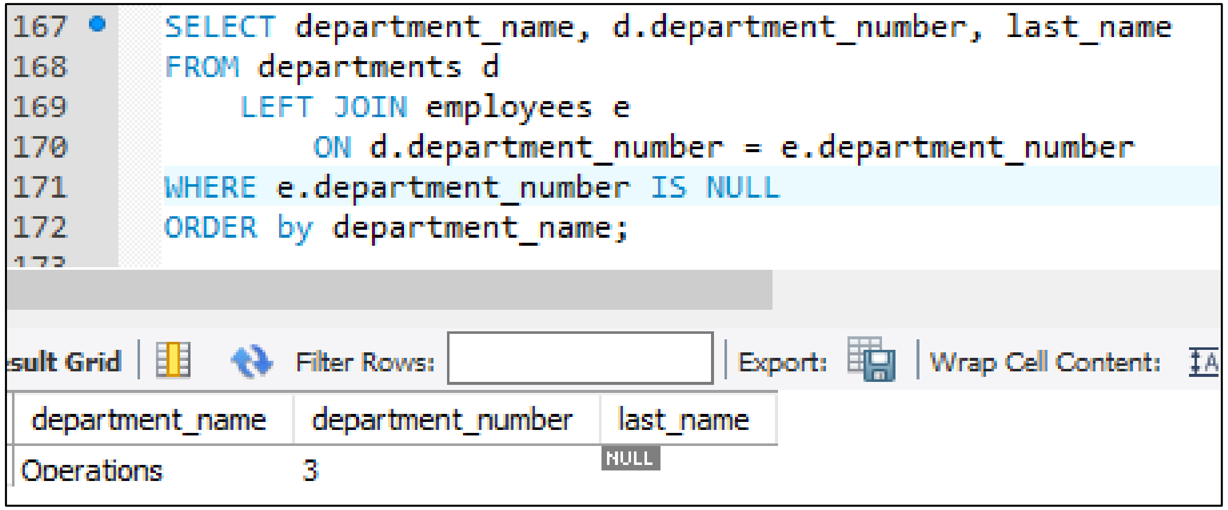
```
SELECT department_name, d.department_number, last_name
FROM departments d
    RIGHT JOIN employees e
        ON d.department_number = e.department_number
ORDER by department_name;
```

department_name	department_number	last_name
NULL	NULL	Watson
NULL	NULL	Locario
Accountina	1	Hernandez
Maintenance	5	Hardv
Pavroll	2	Smith
Pavroll	2	Simonian
Pavroll	2	Aaronsen
Personnel	4	Jones
Personnel	4	O'Learv



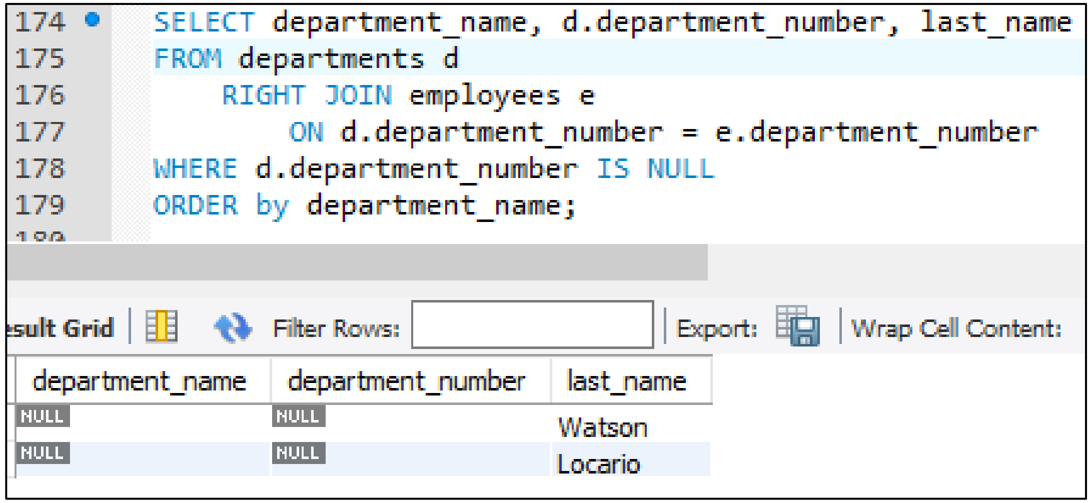
Outer Joins

Keywords: SELECT, FROM, LEFT/RIGHT JOIN, ON, [WHERE, ORDER BY, LIMIT]



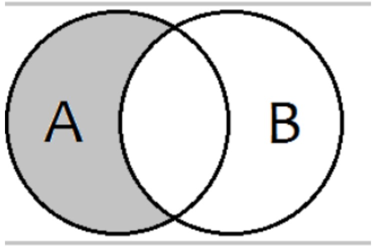
```
SELECT department_name, d.department_number, last_name
FROM departments d
    LEFT JOIN employees e
        ON d.department_number = e.department_number
WHERE e.department_number IS NULL
ORDER by department_name;
```

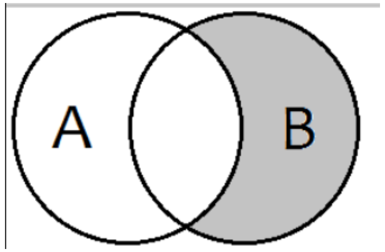
department_name	department_number	last_name
Operations	3	NULL



```
SELECT department_name, d.department_number, last_name
FROM departments d
    RIGHT JOIN employees e
        ON d.department_number = e.department_number
WHERE d.department_number IS NULL
ORDER by department_name;
```

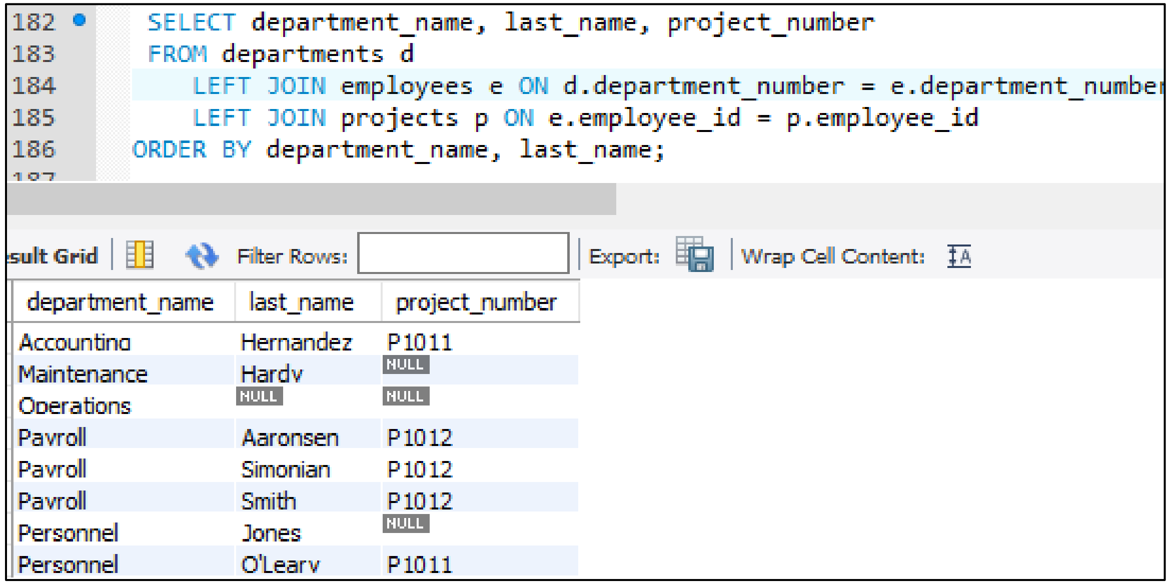
department_name	department_number	last_name
NULL	NULL	Watson
NULL	NULL	Locario





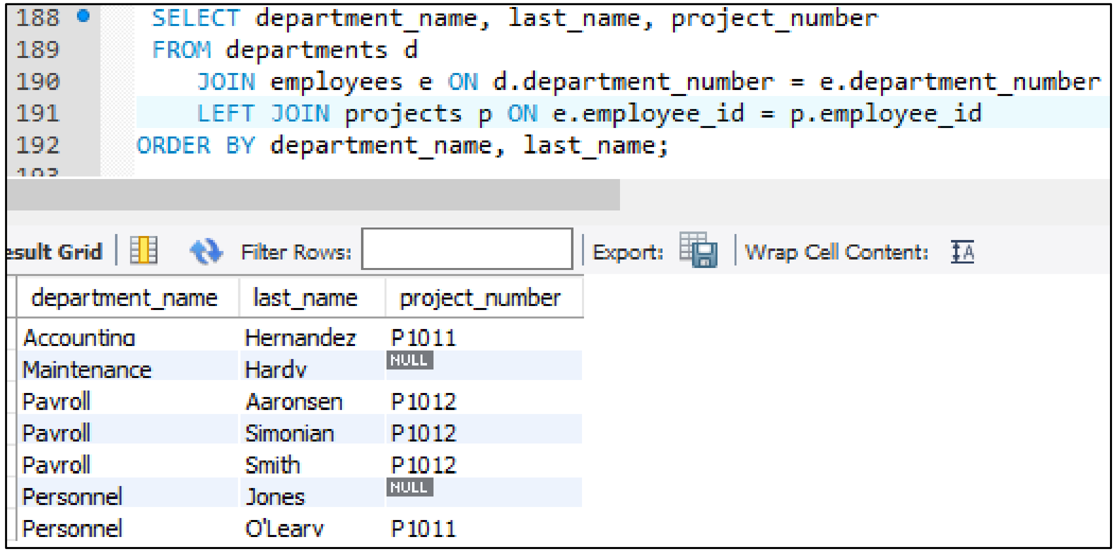
Outer Joins

Keywords: SELECT, FROM, LEFT/RIGHT JOIN, ON, [WHERE, ORDER BY, LIMIT]



```sql
 SELECT department_name, last_name, project_number
 FROM departments d
    LEFT JOIN employees e ON d.department_number = e.department_number
    LEFT JOIN projects p ON e.employee_id = p.employee_id
ORDER BY department_name, last_name;
```

department_name	last_name	project_number
Accounting	Hernandez	P1011
Maintenance	Hardy	NULL
Operations	NULL	NULL
Payroll	Aaronsen	P1012
Payroll	Simonian	P1012
Payroll	Smith	P1012
Personnel	Jones	NULL
Personnel	O'Leary	P1011



```sql
 SELECT department_name, last_name, project_number
 FROM departments d
    JOIN employees e ON d.department_number = e.department_number
    LEFT JOIN projects p ON e.employee_id = p.employee_id
ORDER BY department_name, last_name;
```

department_name	last_name	project_number
Accounting	Hernandez	P1011
Maintenance	Hardy	NULL
Payroll	Aaronsen	P1012
Payroll	Simonian	P1012
Payroll	Smith	P1012
Personnel	Jones	NULL
Personnel	O'Leary	P1011

UNION Operations

The syntax for a union operation

```
    SELECT_statement_1
UNION [ALL]
    SELECT_statement_2
[UNION [ALL]
    SELECT_statement_3]...
[ORDER BY order_by_list]
```

Rules for a union

- Each result set must return the same number of columns.
- The corresponding columns in each result set must have compatible data types.
- The column names in the final result set are taken from the first SELECT clause.

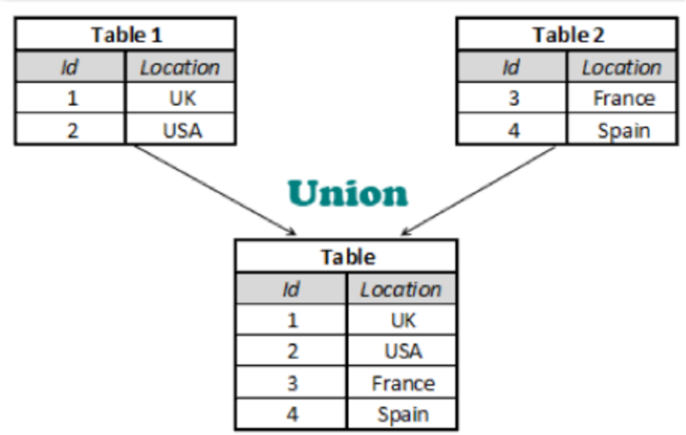


Table 1

Id	Location
1	UK
2	USA

Table 2

Id	Location
3	France
4	Spain

Table

Id	Location
1	UK
2	USA
3	France
4	Spain

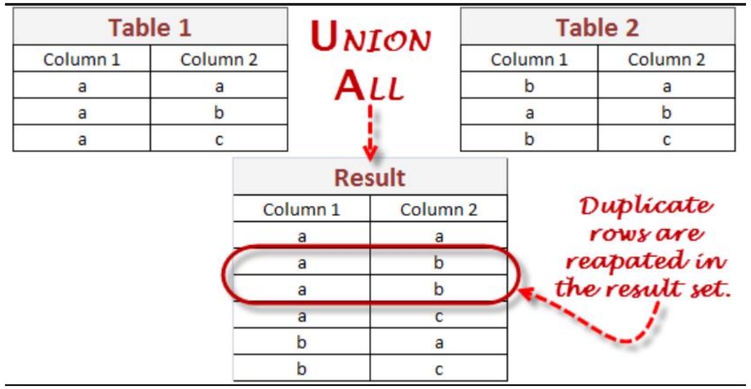


Table 1

Column 1	Column 2
a	a
a	b
a	c

Table 2

Column 1	Column 2
b	a
a	b
b	c

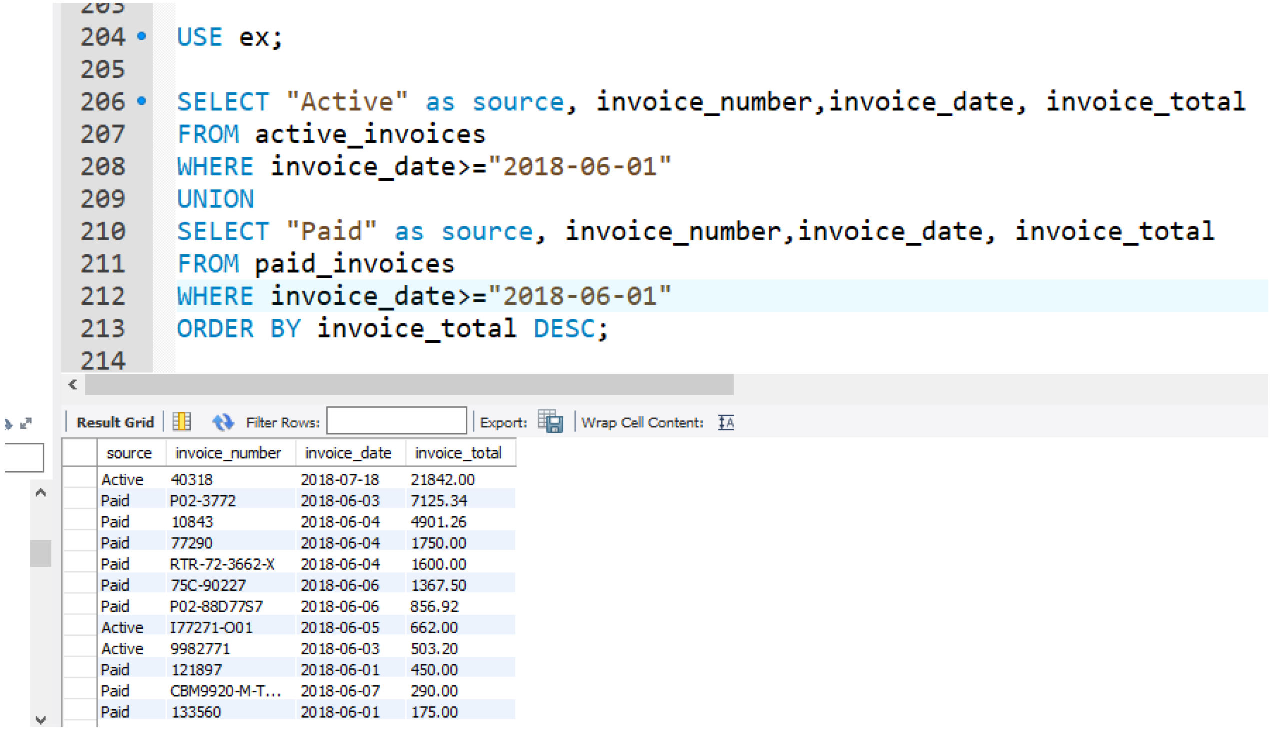
Result

Column 1	Column 2
a	a
a	b
a	b
a	c
b	a
b	c

Duplicate rows are reapated in the result set.

Unions

Keywords: UNION, SELECT, FROM, [LEFT/RIGHT] JOIN, ON, [WHERE, ORDER BY, LIMIT]



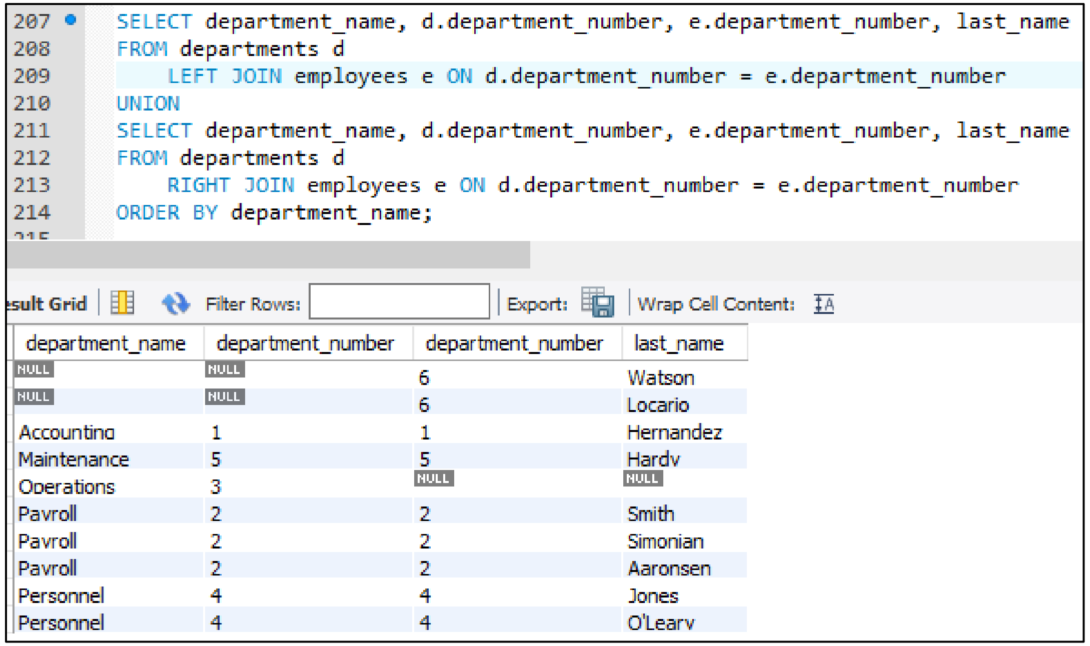
```sql
USE ex;

SELECT "Active" as source, invoice_number,invoice_date, invoice_total
FROM active_invoices
WHERE invoice_date>="2018-06-01"
UNION
SELECT "Paid" as source, invoice_number,invoice_date, invoice_total
FROM paid_invoices
WHERE invoice_date>="2018-06-01"
ORDER BY invoice_total DESC;
```

source	invoice_number	invoice_date	invoice_total
Active	40318	2018-07-18	21842.00
Paid	P02-3772	2018-06-03	7125.34
Paid	10843	2018-06-04	4901.26
Paid	77290	2018-06-04	1750.00
Paid	RTR-72-3662-X	2018-06-04	1600.00
Paid	75C-90227	2018-06-06	1367.50
Paid	P02-88D77S7	2018-06-06	856.92
Active	I77271-O01	2018-06-05	662.00
Active	9982771	2018-06-03	503.20
Paid	121897	2018-06-01	450.00
Paid	CBM9920-M-T...	2018-06-07	290.00
Paid	133560	2018-06-01	175.00

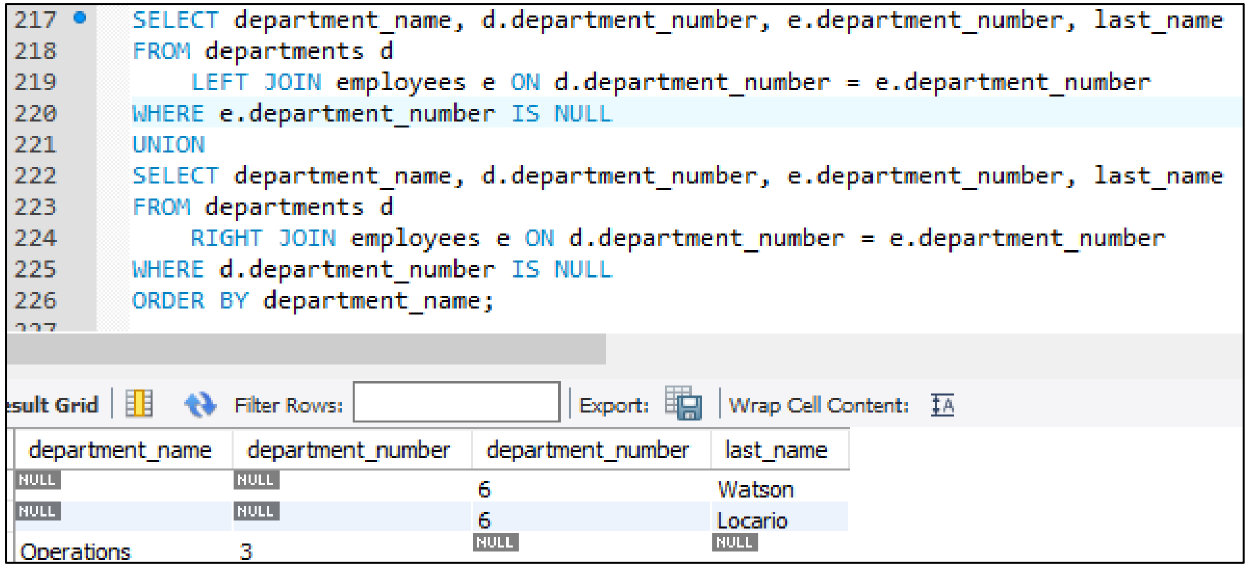
Full Outer Join with Union Statement

Keywords: UNION, SELECT, FROM, [LEFT/RIGHT] JOIN, ON, [WHERE, ORDER BY, LIMIT]



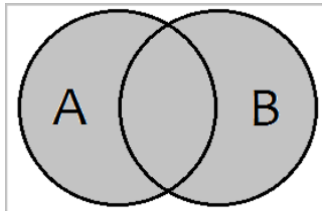
```
SELECT department_name, d.department_number, e.department_number, last_name
FROM departments d
    LEFT JOIN employees e ON d.department_number = e.department_number
UNION
SELECT department_name, d.department_number, e.department_number, last_name
FROM departments d
    RIGHT JOIN employees e ON d.department_number = e.department_number
ORDER BY department_name;
```

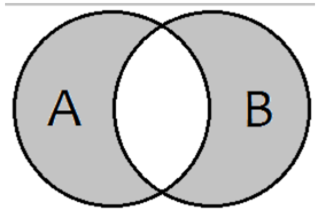
department_name	department_number	department_number	last_name
NULL	NULL	6	Watson
NULL	NULL	6	Locario
Accounting	1	1	Hernandez
Maintenance	5	5	Hardy
Operations	3	NULL	NULL
Payroll	2	2	Smith
Payroll	2	2	Simonian
Payroll	2	2	Aaronsen
Personnel	4	4	Jones
Personnel	4	4	O'Leary



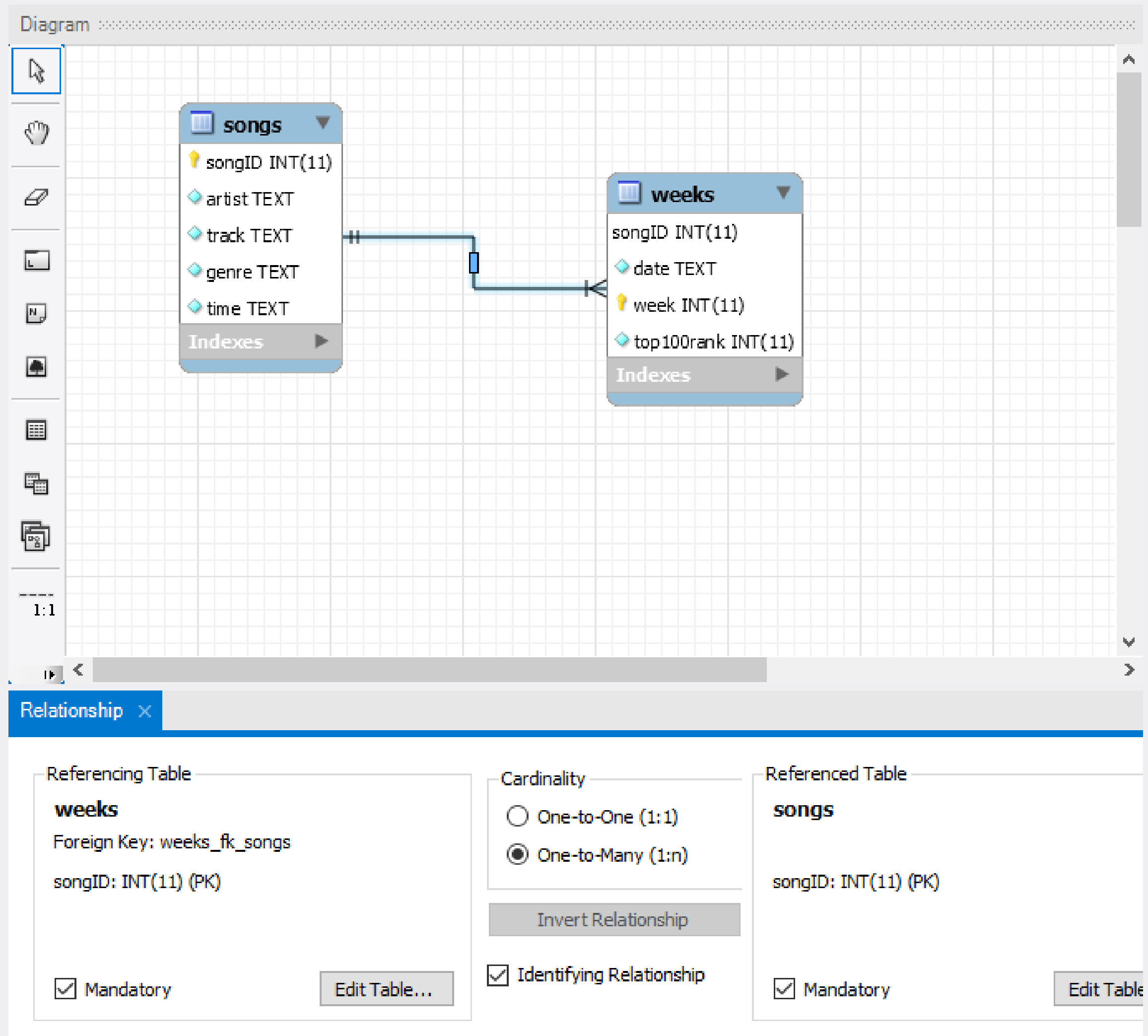
```
SELECT department_name, d.department_number, e.department_number, last_name
FROM departments d
    LEFT JOIN employees e ON d.department_number = e.department_number
WHERE e.department_number IS NULL
UNION
SELECT department_name, d.department_number, e.department_number, last_name
FROM departments d
    RIGHT JOIN employees e ON d.department_number = e.department_number
WHERE d.department_number IS NULL
ORDER BY department_name;
```

department_name	department_number	department_number	last_name
NULL	NULL	6	Watson
NULL	NULL	6	Locario
Operations	3	NULL	NULL





Music Database



songs

- songID INT(11)
- artist TEXT
- track TEXT
- genre TEXT
- time TEXT

Indexes

weeks

- songID INT(11)
- date TEXT
- week INT(11)
- top100rank INT(11)

Indexes

Referencing Table

weeks

Foreign Key: weeks_fk_songs

songID: INT(11) (PK)

☑ Mandatory

Edit Table...

Cardinality

○ One-to-One (1:1)

● One-to-Many (1:n)

Invert Relationship

☑ Identifying Relationship

Referenced Table

songs

songID: INT(11) (PK)

☑ Mandatory

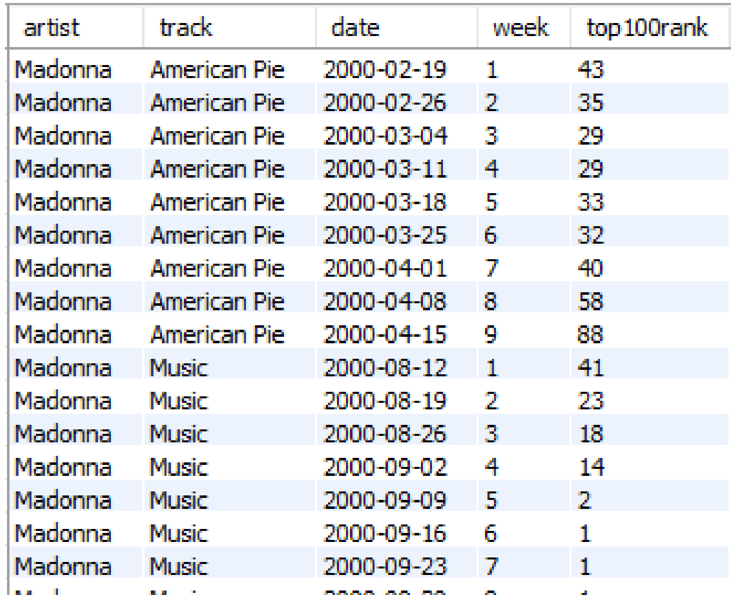
Edit Table

script files = create_music.sql

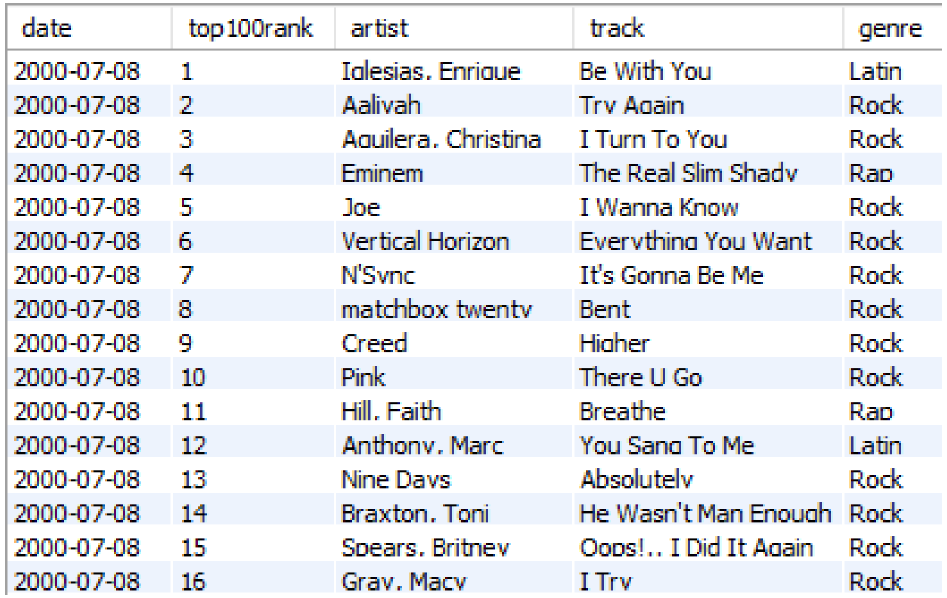
Practice Exercises

Use the musicdb: script files = create_music.sql

1. List of appearances by Madonna on Top 100 by song by week
2. Billboard Top 100 for week of July 8, 2000.



artist	track	date	week	top100rank
Madonna	American Pie	2000-02-19	1	43
Madonna	American Pie	2000-02-26	2	35
Madonna	American Pie	2000-03-04	3	29
Madonna	American Pie	2000-03-11	4	29
Madonna	American Pie	2000-03-18	5	33
Madonna	American Pie	2000-03-25	6	32
Madonna	American Pie	2000-04-01	7	40
Madonna	American Pie	2000-04-08	8	58
Madonna	American Pie	2000-04-15	9	88
Madonna	Music	2000-08-12	1	41
Madonna	Music	2000-08-19	2	23
Madonna	Music	2000-08-26	3	18
Madonna	Music	2000-09-02	4	14
Madonna	Music	2000-09-09	5	2
Madonna	Music	2000-09-16	6	1
Madonna	Music	2000-09-23	7	1



date	top100rank	artist	track	genre
2000-07-08	1	Iglesias, Enrique	Be With You	Latin
2000-07-08	2	Aaliyah	Try Again	Rock
2000-07-08	3	Aguilera, Christina	I Turn To You	Rock
2000-07-08	4	Eminem	The Real Slim Shady	Rap
2000-07-08	5	Joe	I Wanna Know	Rock
2000-07-08	6	Vertical Horizon	Everything You Want	Rock
2000-07-08	7	N'Sync	It's Gonna Be Me	Rock
2000-07-08	8	matchbox twenty	Bent	Rock
2000-07-08	9	Creed	Higher	Rock
2000-07-08	10	Pink	There U Go	Rock
2000-07-08	11	Hill, Faith	Breathe	Rap
2000-07-08	12	Anthony, Marc	You Sang To Me	Latin
2000-07-08	13	Nine Days	Absolutely	Rock
2000-07-08	14	Braxton, Toni	He Wasn't Man Enough	Rock
2000-07-08	15	Spears, Britney	Oops!.. I Did It Again	Rock
2000-07-08	16	Gray, Macy	I Try	Rock

1

2