

Assignment 4

1. Write a query to display the number, title, and cost of books that have the lowest cost in the system. Sort the results by book number.

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the schema tree under 'Administration' with 'Schemas' selected, showing 'company', 'sakila', and 'sys'. The main area contains a SQL editor window with the following code:

```
1 -- Question 1
2
3 SELECT BOOK_NUM, BOOK_TITLE, BOOK_COST FROM BOOK
4 WHERE BOOK_COST = (SELECT MIN(BOOK_COST) FROM BOOK)
5 ORDER BY BOOK_COST ASC;
```

Below the code, the 'Result Grid' shows the output:

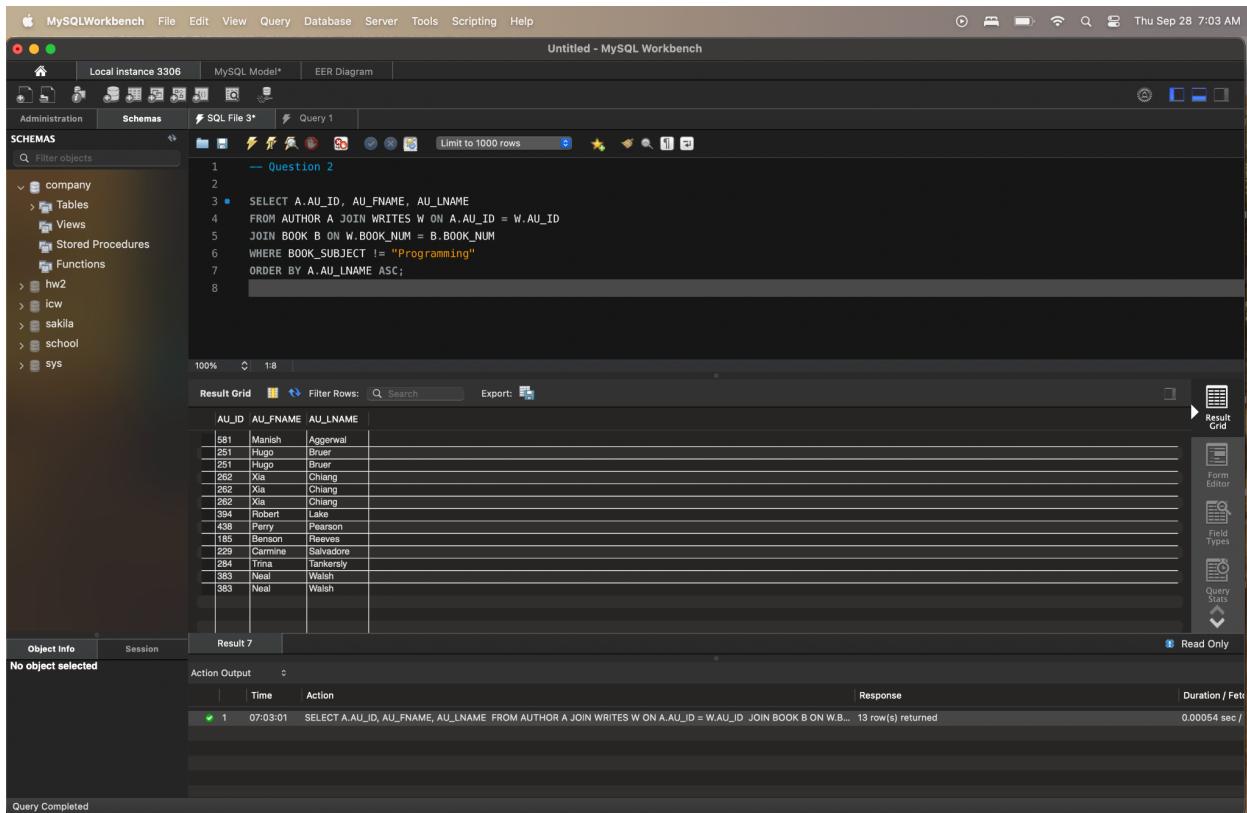
BOOK_NUM	BOOK_TITLE	BOOK_COST
5239	J++ in Mobile Apps	49.95
5241	JAVA First Steps	49.95
5248	What You Always Wanted to Know About Datab...	49.95
5254	Coding Style for Maintenance	49.95
None	None	None

The 'Action Output' section at the bottom shows the following details:

Time	A... Response	Duration / Fetch Time
18:23:07	4 row(s) returned	0.00099 sec / 0.000...

At the bottom left, it says 'Query Completed'.

2. Write a query to display the author ID, first and last name for all authors who have never written a book with the subject Programming. Sort the results by author last name



The screenshot shows the MySQL Workbench interface with a query editor and a results grid.

Query Editor:

```
1 -- Question 2
2
3 SELECT A.AU_ID, AU_FNAME, AU_LNAME
4 FROM AUTHOR A JOIN WRITES W ON A.AU_ID = W.AU_ID
5 JOIN BOOK B ON W.BOOK_NUM = B.BOOK_NUM
6 WHERE BOOK SUBJECT != "Programming"
7 ORDER BY A.AU_LNAME ASC;
8
```

Result Grid:

AU_ID	AU_FNAME	AU_LNAME
581	Manish	Aggarwal
251	Hugo	Bruer
251	Hugo	Bruer
282	Xia	Chiang
282	Xia	Chiang
322	Xia	Chiang
394	Robert	Lake
438	Perry	Pearson
185	Benson	Reeves
229	Carmine	Salvadore
284	Trina	Tankersky
383	Neal	Walsh
383	Neal	Walsh

Action Output:

Action	Time	Response	Duration / Feti
SELECT A.AU_ID, AU_FNAME, AU_LNAME FROM AUTHOR A JOIN WRITES W ON A.AU_ID = W.AU_ID JOIN BOOK B ON W.B...	07:03:01	13 row(s) returned	0.00054 sec /

3. Write a query to display the book number, title, subject, and cost for all books that are on the subjects of “Middleware” or “Cloud,” and that cost more than \$70 sorted by book number.

The screenshot shows the MySQL Workbench interface with the following details:

- Top Bar:** MySQLWorkbench, File, Edit, View, Query, Database, Server, Tools, Scripting, Help. Date: Thu Sep 28 7:07 AM.
- Left Sidebar (Schemas):** Local Instance 3306, Administration, Schemas. Schemas listed: company, Tables, Views, Stored Procedures, Functions, hw2, icw, sakila, school, sys.
- Central Area (Query Editor):** Untitled - MySQL Workbench. SQL File 3*. Query 1. The query is:

```
1 -- Question 3
2
3 SELECT BOOK_NUM, BOOK_TITLE, BOOK SUBJECT, BOOK_COST FROM BOOK
4 WHERE BOOK SUBJECT = "Middleware" OR BOOK SUBJECT = "Cloud"
5 AND BOOK_COST > 70;
```
- Result Grid:** Shows the results of the query. The columns are BOOK_NUM, BOOK_TITLE, BOOK SUBJECT, and BOOK_COST. The data is:

BOOK_NUM	BOOK_TITLE	BOOK SUBJECT	BOOK_COST
S236	Database in the Cloud	Cloud	78.95
S242	CF in Middleware Deployment	Middleware	59.95
S245	The Golden Road to Platform Independence	Middleware	119.95
S250	Reengineering the Middle Tier	Middleware	89.95
NULL	NULL	NULL	NULL
- Bottom Panel (Object Info):** No object selected.
- Log Panel (Action Output):** Shows the executed query and its result.

Action	Time	Response	Duration / Feti
SELECT BOOK_NUM, BOOK_TITLE, BOOK SUBJECT, BOOK_COST FROM BOOK WHERE BOOK SUBJECT = "Middleware" O...	07:06:56	4 row(s) returned	0.0011 sec / 0.

4. Write a query to display the author's last name, first name, book title, and replacement cost for each book. Sort the results by book number and then author ID.

The screenshot shows the MySQL Workbench interface with the following details:

- Toolbar:** MySQLWorkbench, File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Central Area:** Untitled - MySQL Workbench. It displays a SQL editor window titled "Question 4" containing the following query:

```
1 -- Question 4
2
3   SELECT A.AU_LNAME, A.AU_FNAME, B.BOOK_TITLE, B.BOOK_COST FROM AUTHOR A
4   JOIN WRITES W ON A.AU_ID = W.AU_ID
5   JOIN BOOK B ON W.BOOK_NUM = B.BOOK_NUM;
```
- Result Grid:** Shows the output of the query in a tabular format with columns: AU_LNAME, AU_FNAME, BOOK_TITLE, and BOOK_COST. The results are sorted by book number and then author ID. The data includes rows for authors like Reeves, Benson, Chiang, etc., and their corresponding book titles and costs.
- Right Panel:** Contains various tools and panels: Result Grid, Form Editor, Field Types, Query Stats, and Execution Plan.
- Bottom Status:** Action Output, Time (07:15:17), Action (SELECT ...), Response (25 row(s) returned), Duration / Feti (0.00074 sec /).

5. Write a query to display the book number and the number of times each book has been checked out. Do not include books that have never been checked out. Sort the results by the number of times checked out in descending order and then by book number in descending order.

The screenshot shows the MySQL Workbench interface with the following details:

- Toolbar:** MySQLWorkbench, File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Query Editor:** Untitled - MySQL Workbench, Local instance 3306, MySQL Model*, EER Diagram. It contains a SQL query:

```
1 -- Question 5
2
3 • SELECT COUNT(CHECK_OUT_DATE) AS "Check out count", BOOK_NUM FROM CHECKOUT
4 GROUP BY BOOK_NUM ORDER BY COUNT(CHECK_OUT_DATE) DESC, BOOK_NUM DESC;
```
- Result Grid:** Shows the output of the query with columns "Check out cou..." and "BOOK_NUM". The data is as follows:

Check out cou...	BOOK_NUM
12	5298
9	5295
7	5240
6	5238
5	5237
4	5254
4	5252
4	5249
4	5248
4	5244
4	5242
3	5248
2	5243

- Action Output:** Shows the query execution log:

Action	Time	Response
SELECT COUNT(CHECK_OUT_DATE) AS "Check out count", BOOK_...	07:54:45	13 row(s) returned
- Status:** Query Completed.

6. Generate a listing of all purchases made by the customers by displaying customer code, invoice number, date (For example: Display as '01-Jan-19'), p_descript, line units and line price. Sort the results by customer code, invoice number, and product description.

The screenshot shows the Oracle SQL Developer interface with the following details:

- Top Bar:** Local instance 3306
- Schemas:** company, hw4, sakila, sys
- Query Editor:**

```

1 -- Question 6
2
3
4 SELECT C.CUS_CODE, I.INV_NUMBER, DATE_FORMAT(I.INV_DATE, "%d-%b-%y") AS INV_DATE, P.P_DESCRIP, L.LINE_UNITS, L.LINE_PRICE
5 FROM CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE
6 JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER
7 JOIN PRODUCT P ON L.P_CODE = P.P_CODE
8 ORDER BY C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP;
9

```
- Result Grid:** Displays the query results in a tabular format. The columns are CUS_CODE, INV_NUMBER, INV_DATE, P_DESCRIP, LINE_UNITS, and LINE_PRICE. The data includes various items like claw hammers, rat-tail files, and PVC pipes.
- Right Panel:** Shows icons for Result Grid, Form Editor, Field Types, Query Stats, and Execution Plan.
- Action Output:** Shows the execution details: 1 row(s) returned, 0.0017 sec / 0.00001...
- Status:** Query Completed

7. Using the output shown below as your guide, generate a list of customer purchases, including the subtotals for each of the invoice line numbers. The subtotal is a derived attribute calculated by multiplying LINE_UNITS by LINE_PRICE. Sort the output by customer code, invoice number, and product description. Be certain to use the column aliases as shown below. **Be certain to use the column aliases as shown in the figure**

```

1 -- Question 7
2
3 SELECT C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP, L.LINE_UNITS AS "Units Bought", L.LINE_PRICE AS "Unit Price", L.LINE_UNITS * L.LINE_PRICE AS "Subtotal"
4 FROM CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE
5 JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER
6 JOIN PRODUCT P ON P.P_CODE = P.P_CODE
7 ORDER BY C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP;
8

```

CUS_CODE	INV_NUMBER	P_DESCRIP	Units Bought	Unit Price	Subtotal
10011	1002	1.25-in. metal screw, 25	2.00	4.99	9.9800
10011	1002	2.5-in. wd. screw, 50	2.00	4.99	9.9800
10011	1002	7.25-in. pwr. saw blade	2.00	4.99	9.9800
10011	1002	9.00-in. pwr. saw blade	2.00	4.99	9.9800
10011	1002	B&D cordless drill, 1/2-in.	2.00	4.99	9.9800
10011	1002	B&D jigsaw, 12-in. blade	2.00	4.99	9.9800
10011	1002	B&D jigsaw, 8-in. blade	2.00	4.99	9.9800
10011	1002	Clew hammer	2.00	4.99	9.9800
10011	1002	Recip chain saw, 16 in.	2.00	4.99	9.9800
10011	1002	Recip saw, 14-in., 2x50	2.00	4.99	9.9800
10011	1002	Hrd cloth, 1/4-in., 2x50	2.00	4.99	9.9800
10011	1002	Power painter, 15 psi...	2.00	4.99	9.9800
10011	1002	PVC pipe, 3.5-in., 8-ft	2.00	4.99	9.9800
10011	1002	Rat-tail file, 1/8-in. fine	2.00	4.99	9.9800
10011	1002	Sledge hammer, 12 lb.	2.00	4.99	9.9800
10011	1002	Steel matting, 4x8x1/8"	2.00	4.99	9.9800
10011	1004	1.25-in. metal screw, 25	3.00	4.99	14.9700
10011	1004	1.25-in. metal screw, 25	2.00	9.95	19.9000
10011	1004	2.5-in. wd. screw, 50	2.00	9.95	19.9000
10011	1004	2.5-in. wd. screw, 50	3.00	4.99	14.9700

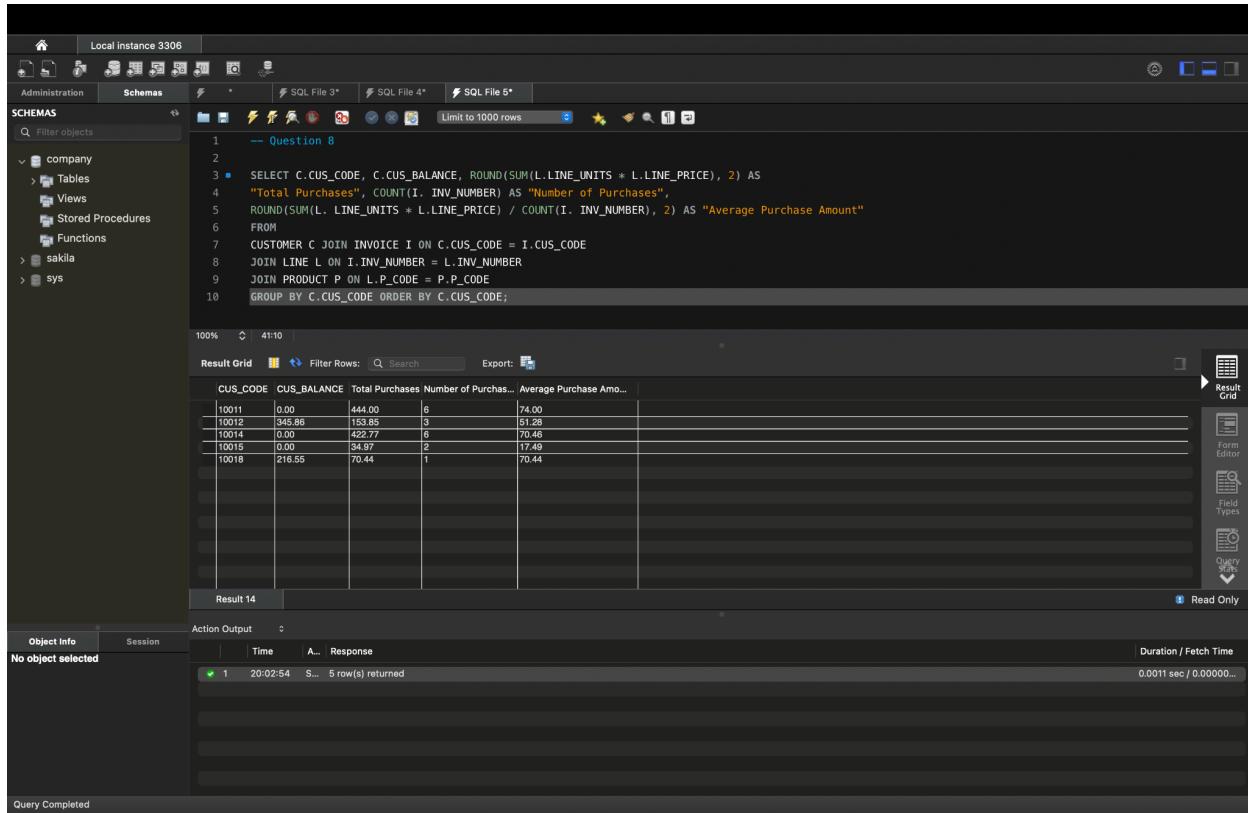
Result 15

Action Output

Time	Action	Response	Duration / Fetch Time
21:17:29	SELECT C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP, L.LINE_UNITS AS "Units Bought", L.LINE_PRICE AS "Unit Price", L.LINE_UNITS * L.LINE_PRICE AS "Subtotal" FROM CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER JOIN PRODUCT P ON P.P_CODE = P.P_CODE ORDER BY C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP;	288 row(s) returned	0.0026 sec / 0.00006...
21:18:18	SELECT C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP, L.LINE_UNITS AS "Units Bought", L.LINE_PRICE AS "Unit Price", L.LINE_UNITS * L.LINE_PRICE AS "Subtotal" FROM CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER JOIN PRODUCT P ON P.P_CODE = P.P_CODE ORDER BY C.CUS_CODE, I.INV_NUMBER, P.P_DESCRIP;	288 row(s) returned	0.0016 sec / 0.00008...

Query Completed

8. Use a query to compute the total of all purchases, the number of purchases, and the average purchase amount made by each customer. Sort the results by customer code.
Your output values must match those shown in Figure



The screenshot shows the Oracle SQL Developer interface with a query editor window. The query is:

```
-- Question 8
SELECT C.CUS_CODE, C.CUS_BALANCE, ROUND(SUM(L.LINE_UNITS * L.LINE_PRICE), 2) AS "Total Purchases", COUNT(I. INV_NUMBER) AS "Number of Purchases", ROUND(SUM(L. LINE_UNITS * L.LINE_PRICE) / COUNT(I. INV_NUMBER), 2) AS "Average Purchase Amount"
FROM
CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE
JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER
JOIN PRODUCT P ON L.P_CODE = P.P_CODE
GROUP BY C.CUS_CODE ORDER BY C.CUS_CODE;
```

The result grid displays the following data:

CUS_CODE	CUS_BALANCE	Total Purchases	Number of Purchases	Average Purchase Amo...
10011	0.00	444.00	6	74.00
10012	345.86	153.85	3	51.28
10014	0.00	422.77	6	70.46
10015	0.00	34.97	2	17.49
10018	216.55	70.44	1	70.44

The Action Output panel shows the following information:

Action	Time	Response	Duration / Fetch Time
1	20:02:54	S... 5 row(s) returned	0.0011 sec / 0.00000...

9. Use a query to show the invoices and invoice totals. Sort the results by customer code and then by invoice number. **Your output values must match those shown in Figure.**

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the schema tree with 'company' selected, containing Tables, Views, Stored Procedures, and Functions. The central workspace contains a SQL editor window with the following query:

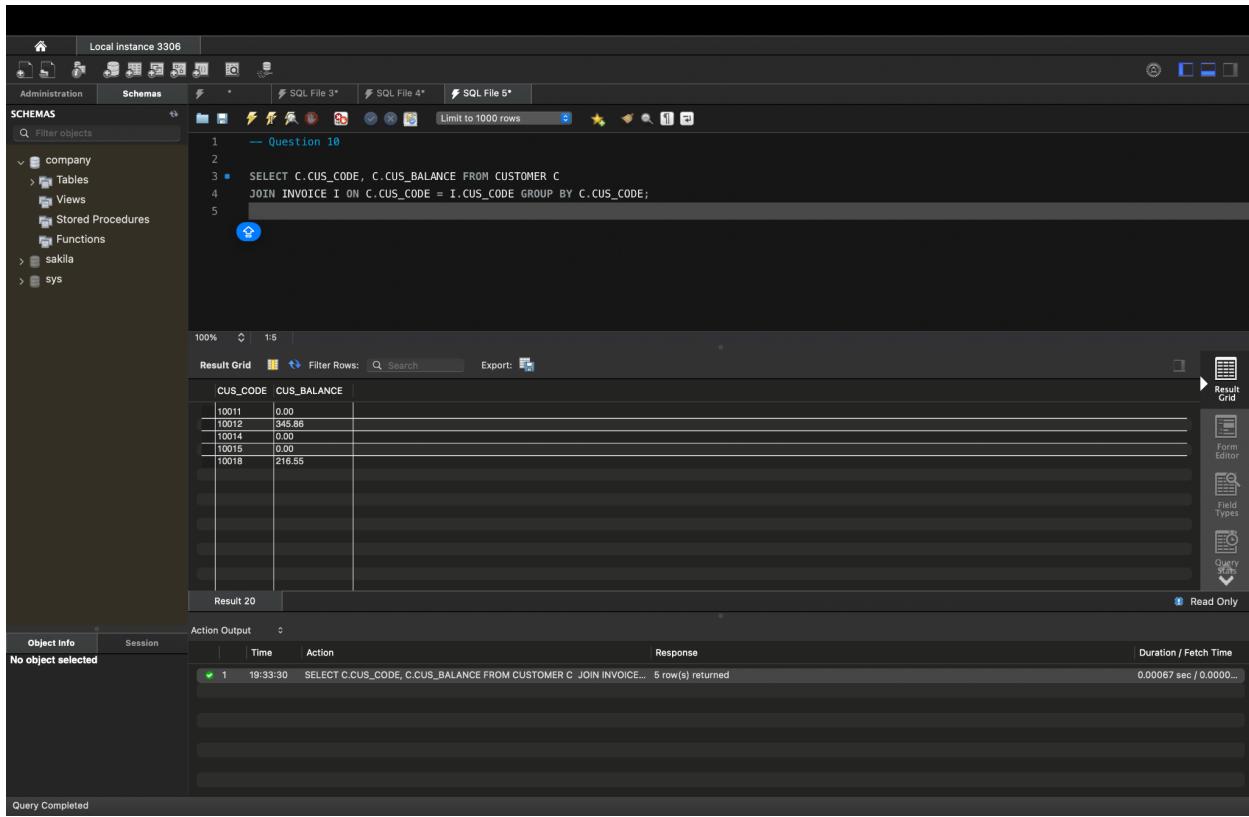
```
-- Question 9
SELECT C.CUS_CODE, I.INV_NUMBER, ROUND(SUM(L.LINE_UNITS * L.LINE_PRICE), 2) AS "Invoice Total"
FROM CUSTOMER C JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE
JOIN LINE L ON I.INV_NUMBER = L.INV_NUMBER
JOIN PRODUCT P ON L.P_CODE = P.P_CODE
GROUP BY C.CUS_CODE, I.INV_NUMBER
ORDER BY C.CUS_CODE, I.INV_NUMBER;
```

The result grid shows the output of the query:

CUS_CODE	INV_NUMBER	Invoice Total
10011	1002	9.98
10011	1004	34.87
10011	1008	399.15
10011	1003	153.88
10014	1001	24.84
10014	1006	527.83
10015	1007	34.97
10018	1005	70.44

The bottom status bar indicates 'Query Completed'.

10. List the balances of customers who have made purchases during the current invoice cycle—that is, for the customers who appear in the INVOICE table. Be certain to display CUS_CODE and CUS_BALANCE.



The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the schema tree with 'company' and 'sakila' schemas selected. The central workspace contains a query editor window with the following SQL code:

```
-- Question 10
SELECT C.CUS_CODE, C.CUS_BALANCE FROM CUSTOMER C
JOIN INVOICE I ON C.CUS_CODE = I.CUS_CODE GROUP BY C.CUS_CODE;
```

The result grid shows the following data:

CUS_CODE	CUS_BALANCE
10011	0.00
10012	345.88
10014	0.00
10015	0.00
10018	216.55

The bottom pane shows the action output with one row listed:

Action	Time	Response	Duration / Fetch Time
SELECT C.CUS_CODE, C.CUS_BALANCE FROM CUSTOMER C JOIN INVOICE...	19:33:30	5 row(s) returned	0.00067 sec / 0.0000...

A status message at the bottom left says "Query Completed".

11. 3NF from the given dependency diagram

A	C	E	F
---	---	---	---

A	D
---	---

C	B
---	---

E	G
---	---