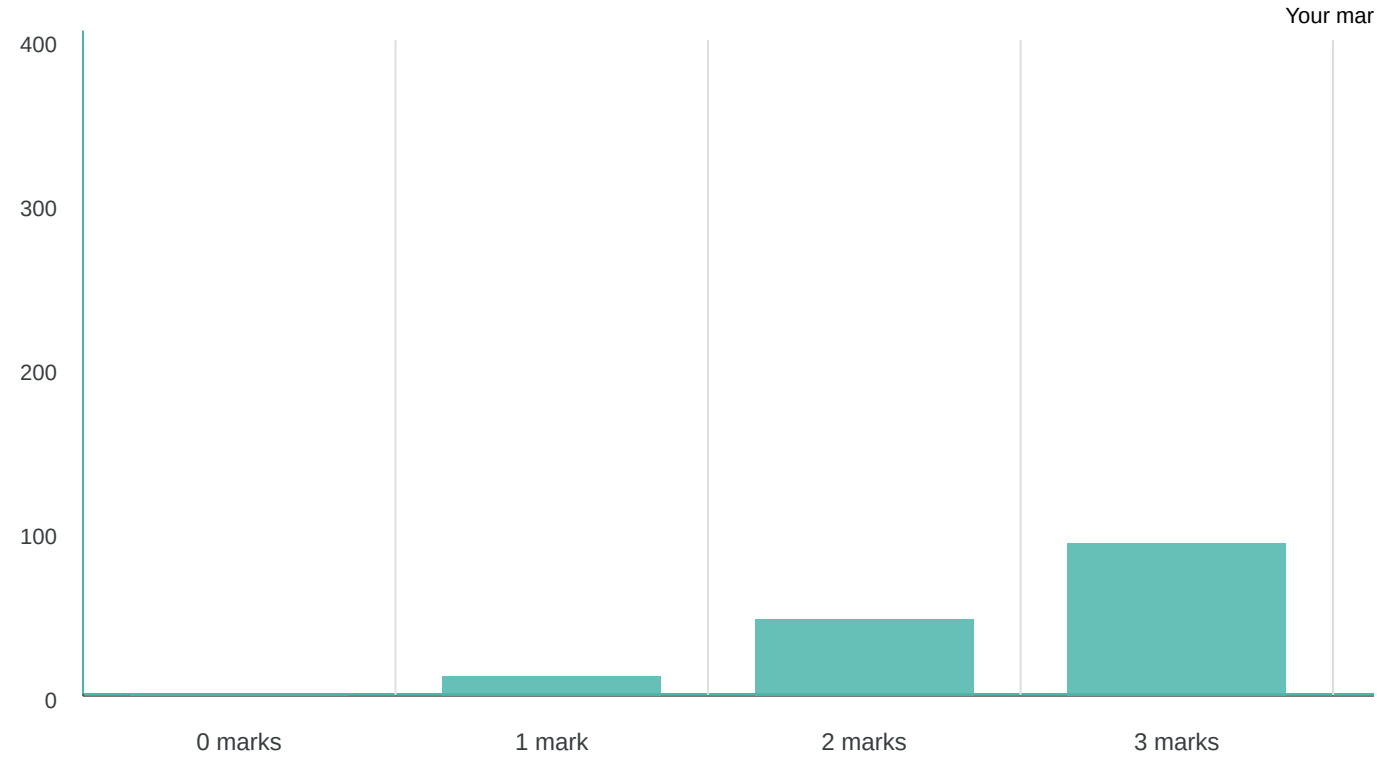
Quiz 2

Student Mark Distribution



Your Individual Results

Deadline	Friday, 11 October 2019 at 11:59PM
Latest Submission	Friday, 11 October 2019 at 11:30AM
Raw Mark	4.00/4.00 (100.00%)
Late Penalty	N/A
Final Mark	4.00/4.00 (100.00%)

Question 1 (1 mark)

Use the following tables to answer Questions 1, 2, and 3.

Account table instance:

```
branchName  | accountNo | balance
------------+-----------+---------
UNSW        | U-245     | 1000
UNSW        | U-291     | 2000
Randwick    | R-245     | 20000
Coogee      | C-123     | 15000
Coogee      | C-124     | 25000
Clovelly    | Y-123     | 1000
Maroubra    | M-222     | 5000
Maroubra    | M-225     | 12000
```

Owner table instance:

```
account | customer
--------+---------
U-245   | 12345
U-291   | 12345
U-291   | 12666
R-245   | 12666
C-123   | 32451
C-124   | 22735
Y-123   | 76543
M-222   | 92754
M-225   | 12345
```

Q. What query would produce a count of how much money (total balance) is held in each branchName (suburb)?

(a) ○	SELECT a1.branchName, SUM(a1.balance) FROM Account a1, Account a2 WHERE a1.branchName = a2.branchName
(b) ○	SELECT a1.branchName, COUNT(balance) FROM Account a1, Account a2 WHERE a1.branchName = a2.branchName
(c) ◉	SELECT branchName, SUM(balance) FROM Account GROUP BY branchName
(d) ○	SELECT branchName, COUNT(balance) FROM Account GROUP BY branchName
(e) ○	None of the above queries solves the problem.

✔ Your response was correct.
Mark: 1.00

Question 2 (1 mark)

Use the tables in Question 1 to answer this question.

Q. What query would successfully remove all rows in Account whose branchName's start with C and whose balances are over 10000? You can assume that the tables have been defined in such a way that deleting from Accounts will not be prevented by the existence of the account in the Owner table.

(a) ○	DELETE FROM Account WHERE branchName LIKE '%C' WHERE balance > 10000
(b) ○	DELETE FROM Account WHERE branchName LIKE '%C%' AND balance >= 10000
(c) ○	DELETE FROM Account WHERE branchName LIKE 'C%' WHERE balance >= 10000
(d) ◉	DELETE FROM Account WHERE branchName LIKE 'C%' AND balance > 10000
(e) ○	None of the above SQL statements solves the problem.

✔ Your response was correct.
Mark: 1.00

Question 3 (1 mark)

Use the tables in Question 1 to answer this question.

Q. What queries (may be multiple) would produce tuples of customer IDs and account balance, for all customers with accounts with balances less than 15,000? Note that there is one tuple for each account, so a customer may appear in multiple tuples if they own multiple accounts.

(a) ☑	SELECT o.customer, a.balance FROM Account as a, Owner as o WHERE a.accountNo = o.account AND a.balance < 15000
(b) ☐	SELECT customer, balance FROM Account JOIN Owner WHERE balance < 15000
(c) ☑	SELECT o.customer, a.balance FROM Account as a JOIN Owner as o ON a.accountNo = o.account AND a.balance < 15000

(d) ☐	SELECT o.customer, a.balance FROM Account as a, Owner as o OUTER LEFT JOIN ON a.accountNo = o.account AND a.balance < 15000

✔ Your response was correct.
Mark: max(0.50 + 0.50, 0) = 1.00

Question 4 (1 mark)

Use the following tables to answer Question 4.

Student table instance:

```
zid     | name        | Age
------------+---------+----------
1234567 | Harry Roo   | 21
2345678 | Chen Zhang  | 25
3456789 | Patricia Li | 18
```

Class table instance:

```
code  | name
------+------
3311  | Databases with the Shepherd
6441  | Security learning
1521  | MIPS and fun
```

Enrolment table instance:

```
student | class | mark
--------+-------+------
1234567 | 3311  | NULL
2345678 | 3311  | 76
2345678 | 1521  | 77
2345678 | 6441  | 77
1234567 | 1521  | NULL
```

Q. Which of the following (may be multiple) creates a view that produces a table of tuples containing (student name, class name, mark) for all students and all classes they've taken.

(a) ☐	CREATE VIEW all_marks(student,course,mark) AS SELECT name, name, mark FROM Class as c JOIN Student as s ON s.zid = c.code
(b) ☑	CREATE VIEW all_marks(student,course,mark) AS SELECT s.name, c.name, e.mark FROM Enrolment as e JOIN Class as c ON c.code = e.class JOIN Student as s ON s.zid = e.student

(c) ☑	CREATE VIEW all_marks(student,course,mark) AS SELECT s.name, c.name, e.mark FROM Student as s JOIN Enrolment as e ON s.zid = e.student JOIN Class as c ON c.code = e.class
(d) ☐	CREATE VIEW all_marks(student,course,mark) AS SELECT * FROM Enrolment as e JOIN Class as c ON c.code = e.class JOIN Student as s ON s.zid = e.student WHERE s.name, c.name, e.mark

✔ Your response was correct.
Mark: max(0.50 + 0.50, 0) = 1.00