Appendix B

158.337 Project Marking Sheet

(Print and attach this page to your project report BEFORE you turn it in.)

(Please make sure you provide all the necessary details)

Group Number: 02

Oracle Account: GROUPxx Group02@//inms-oracle.massey.ac.nz:1521/orcl.massey.ac.nz

Group Member 1 - ID number, Name

Peter Fredatovich 98141269

Group Member 2 - ID number, Name

Leonard Phillips 15232331

(Grader's section, please do not write below this)

PART B: _____/60marks

Part B Comments:

Total Marks for the Project

_____/100 is equivalent to_____/25 marks

Worksheet | Query Builder

```
1  --a. Write a query that will list all the books with the price difference (between retail
2  --and cost) of $10 or more. Display your results in the decreasing order of the price difference. (1 mark)
3  SELECT book_title "Book Title", book_cost "Cost", book_retail "RRP", (book_retail- book_cost) AS "Difference"
4      FROM book
5      WHERE (book_retail- book_cost )>= 10
6      ORDER BY "Difference" DESC;
```

Script Output | Query Result | SQL | All Rows Fetched: 10 in 0.073 seconds

	Book Title	Cost	RRP	Difference
1	PAINLESS CHILD-REARING	48	89.95	41.95
2	HOLY GRAIL OF ORACLE	47.25	75.95	28.7
3	REVENGE OF MICKEY	14.2	42.59	28.39
4	DATABASE IMPLEMENTATION	31.4	55.95	24.55
5	BUILDING A CAR WITH TOOTHPICKS	37.8	59.95	22.15
6	SHORTEST POEMS	21.85	39.95	18.1
7	E-BUSINESS THE EASY WAY	37.9	54.5	16.6
8	HOW TO MANAGE THE MANAGER	15.4	31.95	16.55
9	BODYBUILD IN 10 MINUTES A DAY	18.75	30.95	12.2
10	HOW TO GET FASTER PIZZA	17.85	29.95	12.1

Worksheet | Query Builder

```
1  --b. Write a query that will list books in COMPUTER category along with other details (e.g. author(s), etc.).
2  --The query should work for all the case-variations of category values (i.e. 'computer', 'Computer', etc.)
3  --in the database. (1.5 mark)
4  SELECT b.book_title "Title", b.book_cost "Cost", b.book_retail "RRP", a.author_fname ||', '||  a.author_lname "Author"
5      FROM bookauthor ba, book b, author a
6      WHERE  b.book_isbn = ba.ba_isbn
7      AND Ba.ba_authorid = a.author_id
8      AND LOWER(b.book_category) = LOWER('COMPUTER')
9      ORDER BY b.book_title;
```

Script Output | Query... | SQL | All Rows Fetched: 7 in 0.385 seconds

	Title	Cost	RRP	Author
1	DATABASE IMPLEMENTATION	31.4	55.95	JUAN, ADAMS
2	DATABASE IMPLEMENTATION	31.4	55.95	TINA, PETERSON
3	DATABASE IMPLEMENTATION	31.4	55.95	JAMES, AUSTIN
4	E-BUSINESS THE EASY WAY	37.9	54.5	JANICE, JONES
5	HANDCRANKED COMPUTERS	21.8	25	LISA, WHITE
6	HANDCRANKED COMPUTERS	21.8	25	WILLIAM, WHITE
7	HOLY GRAIL OF ORACLE	47.25	75.95	JAMES, AUSTIN

Worksheet | Query Builder

```
1  --c. Write a query that will list books that have retail price $30 or less and were published
2  --in any of the years 1999 or 2001. Display results in the increasing order of the publication year
3  --(and not the publication date) and decreasing retail price. Display the year of publication with
4  --the column titled Publication Year. (1.5 mark)
5  SELECT book_title "Book Title", book_retail "RRP", book_pubdate "Publication Year"
6      FROM book
7      WHERE book_retail <= 30
8      AND (EXTRACT(YEAR FROM book_pubdate)=1999
9      OR EXTRACT(YEAR FROM book_pubdate)=2001)
10     ORDER BY EXTRACT(YEAR FROM book_pubdate), "RRP" DESC;
```

Script Output | Query Result | SQL | All Rows Fetched: 2 in 0.028 seconds

	Book Title	RRP	Publication Year
1	HANDCRANKED COMPUTERS	25	21-JAN-01
2	BIG BEAR AND LITTLE DOVE	8.95	08-NOV-01

Worksheet | Query Builder

```
1  --d. Write a query that lists both customer and author details (only their ids, first and last names).
2  --Provide suitable headings for the merged list. (1.5 mark)
3  SELECT author_id AS "ID", author_fname ||', '|| author_lname AS "Name" FROM author
4      UNION
5      SELECT TO_CHAR(cust_num), cust_fname ||', '|| cust_lname FROM customer;
```

Script Output | Query Result | SQL | All Rows Fetched: 34 in 0.011 seconds

	ID	Name
1	1001	BONITA, MORALES
2	1002	RYAN, THOMPSON
3	1003	LEILA, SMITH
4	1004	THOMAS, PIERSON
5	1005	CINDY, GIRARD
6	1006	MESHIA, CRUZ
7	1007	TAMMY, GIANA
8	1008	KENNETH, JONES
9	1009	JORGE, PEREZ
10	1010	JAKE, LUCAS
11	1011	REESE, MCGOVERN
12	1012	WILLIAM, MCKENZIE
13	1013	NICHOLAS, NGUYEN
14	1014	JASMINE, LEE
15	1015	STEVE, SCHELL
16	1016	MICHELL, DAUM
17	1017	BECCA, NELSON
18	1018	GREG, MONTIASA
19	1019	JENNIFER, SMITH
20	1020	KENNETH, FALAH
21	A100	JAMES, AUSTIN
22	A105	JUAN, ADAMS
23	B100	JACK, BAKER
24	F100	OSCAR, FIELDS
25	J100	JANICE, JONES
26	K100	TAMARA, KZOCHSKY
27	M100	SHEILA, MARTINEZ
28	P100	LISA, PORTER
29	P105	TINA, PETERSON
30	R100	ROBERT, ROBINSON
31	S100	SAM, SMITH
32	W100	WILLIAM, WHITE
33	W105	LISA, WHITE
34	W110	ANTHONY, WILKINSON

Worksheet | Query Builder

```
1  --e. Write a query that will list all the publishers, their details (name, etc.) and total number of published books.
2  --Display your output in the decreasing order of total number of publications. (1.5 marks)
3  SELECT p.pub_name "Publisher Name", p.pub_contact "Contact", p.pub_phone "Phone", COUNT(b.book_pubid) AS "Published Books"
4      FROM publisher p , book b
5      WHERE b.book_pubid = p.pub_id
6      GROUP BY p.pub_name, p.pub_contact, p.pub_phone
7      ORDER BY "Published Books" DESC;
```

Script Output | Query Result | SQL | All Rows Fetched: 5 in 0.138 seconds

	Publisher Name	Contact	Phone	Published Books
1	READING MATERIALS INC.	RENEE SMITH	800-555-9743	5
2	REED-N-RITE	SEBASTIAN JONES	800-555-8284	4
3	AMERICAN PUBLISHING	DAVID DAVIDSON	800-555-1211	3
4	PUBLISH OUR WAY	JANE TOMLIN	010-410-0010	2
5	PRINTING IS US	TOMMIE SEYMOUR	000-714-8321	1

Worksheet | Query Builder

```
1  --f. Write a query that will display the states with more than one customer
2  --Display the state with maximum customers first. (2 marks)
3  SELECT cust_state "Customer State", COUNT(cust_state) "# of customers"
4      FROM customer
5      GROUP BY cust_state
6      HAVING COUNT(cust_state)>1
7      ORDER BY COUNT(cust_state) DESC;
```

Script Output | Query Result

SQL | All Rows Fetched: 5 in 0.016 seconds

	Customer State	# of customers
1	FL	4
2	CA	3
3	GA	2
4	NJ	2
5	WY	2

Worksheet | Query Builder

```
1  --g. Write a query that will list the publisher(s) with the maximum number of published books. If there is more than
2  --one publisher (e.g. 2 publishers) with maximum publications, your query should and list all (i.e. both if 2). (3 marks)
3  SELECT p.pub_name, COUNT(b.book_pubid)
4      FROM book b, publisher p
5      WHERE  b.book_pubid = p.pub_id
6      GROUP BY p.pub_name
7      HAVING COUNT(book_pubid)=(SELECT MAX(COUNT(book_pubid))
8                                FROM book GROUP BY book_pubid);
```

Script Output | Query Result

SQL | All Rows Fetched: 1 in 0.016 seconds

	PUB_NAME	COUNT(B.BOOK_PUBID)
1	READING MATERIALS INC.	4

Worksheet | Query Builder

```
1  --h. Write a query that will list the customer(s) who had ordered maximum number of items (two copies of the same book
2  --will be counted as two items). Again, like g) there can be more than one customer. (3 marks)
3  SELECT c.cust_num "Customer Number", c.cust_fname||' '|| c.cust_lname "Customer", SUM(boi.boi_qty) "Total Order Qty"
4      FROM bookorder bo, customer c, bookorderitem boi
5      WHERE c.cust_num = bo.bo_custnum
6      AND boi.boi_ordernum = bo.bo_ordernum
7      GROUP BY c.cust_num, c.cust_fname||' '|| c.cust_lname
8      HAVING SUM(boi.boi_qty) = (SELECT MAX(SUM(boi.boi_qty))
9                                 FROM bookorderitem boi, customer c, bookorder bo
10                                where c.cust_num = bo.bo_custnum
11                                and boi.boi_ordernum = bo.bo_ordernum
12                                GROUP BY c.cust_num);
```

Script Output | Query Result

SQL | All Rows Fetched: 1 in 0.041 seconds

	Customer Number	Customer	Total Order Qty
1	1007	TAMMY GIANA	8

```
Worksheet  Query Builder
1  --i. Write a query that will display the customer(s) that referred maximum number of customers. Again, like g) there can
2  --be more than one customer. (3 marks)
3  SELECT  cust_referred "Customer Number", COUNT(cust_referred) "# Customers Referred"
4      FROM customer
5      GROUP BY cust_referred
6      HAVING COUNT(cust_referred)=(SELECT MAX(mycount) "# Customers Referred"
7      FROM (select cust_referred,
8             COUNT(cust_referred) mycount
9             FROM customer
10            GROUP BY cust_referred));
```

Script Output | Query Result

SQL | All Rows Fetched: 1 in 0.008 seconds

	Customer Number	# Customers Referred
1	1003	3

```
Worksheet  Query Builder
1  --j. Write a query to list all the books that have multiple authors.
2  --Also, display the number of authors who wrote the book. (2 marks)
3  SELECT b.book_title, a.author_fname ||', '|| a.author_lname "Author"
4      FROM bookauthor ba, book b, author a
5      WHERE b.book_isbn = ba.ba_isbn
6      AND a.author_id = ba.ba_authorid
7      AND ba.ba_isbn IN(
8          SELECT ba_isbn
9          FROM bookauthor
10         GROUP BY ba_isbn
11         HAVING COUNT(*)>1    )
12     ORDER BY b.book_title;
```

Script Output | Query Result

SQL | All Rows Fetched: 10 in 0.072 seconds

	BOOK_TITLE	Author
1	BODYBUILD IN 10 MINUTES A DAY	LISA, PORTER
2	BODYBUILD IN 10 MINUTES A DAY	SAM, SMITH
3	DATABASE IMPLEMENTATION	JAMES, AUSTIN
4	DATABASE IMPLEMENTATION	JUAN, ADAMS
5	DATABASE IMPLEMENTATION	TINA, PETERSON
6	HANDCRANKED COMPUTERS	LISA, WHITE
7	HANDCRANKED COMPUTERS	WILLIAM, WHITE
8	PAINLESS CHILD-REARING	ROBERT, ROBINSON
9	PAINLESS CHILD-REARING	OSCAR, FIELDS
10	PAINLESS CHILD-REARING	JACK, BAKER

Worksheet | Query Builder

```
1  --K)ROW LEVEL TRIGGER
2  --A row-level trigger that uses a sequence to allow customers to be inserted with a null number value
3  --(As per the STATEMENT LEVEL TRIGGER test code)
4
5  --CREATE SEQUENCE
6  CREATE SEQUENCE seq_cust_num
7      INCREMENT BY 1
8      START WITH 1
9      maxvalue 1000
10     nocache
11     nocycle;
12
13 CREATE OR REPLACE TRIGGER trg_dept_add
14     BEFORE INSERT ON customer
15     FOR EACH ROW
16     DECLARE
17     v_custr_num customer.cust_num%TYPE;
18 BEGIN
19     IF :NEW.cust_num IS NULL THEN
20         SELECT seq_cust_num.NextVal
21         INTO v_custr_num
22         FROM dual;
23         :NEW.cust_num := v_custr_num;
24         END IF;
25 END;
26 /
27 show error;
```

Script Output | Query Result

Task completed in 0.312 seconds

```
Sequence SEQ_CUST_NUM created.


Trigger TRG_DEPT_ADD compiled

No errors.
```

Assingment2 | CUSTOMER

Worksheet | Query Builder

```
1  --K) ROW LEVEL TRIGGER Test Code
2  INSERT INTO customer VALUES(NULL, 'DOE', 'JOHN', 'P.O BOX 1231', 'LOS ANGELES','CA','27389', 1001);
3  --See JPEG images for test results
```

Script Output

Task completed in 0.078 seconds

```
1 row inserted.
```

Assingment2 | CUSTOMER

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. | Filter:

	CUST_NUM	CUST_LNAME	CUST_FNAME	CUST_ADDRESS	CUST_CITY	CUST_STATE	CUST_ZIP	CUST_REFERRED
1	1	DOE	JOHN	P.O BOX 1231	LOS ANGELES	CA	27389	1001
2	1001	MORALES	BONITA	P.O. BOX 651	EASTPOINT	FL	32328	(null)
3	1002	THOMPSON	RYAN	P.O. BOX 9835	SANTA MONICA	CA	90404	(null)
4	1003	SMITH	LEILA	P.O. BOX 66	TALLAHASSEE	FL	32306	(null)
5	1004	PIERSON	THOMAS	69821 SOUTH AVENUE	BOISE	ID	83707	(null)

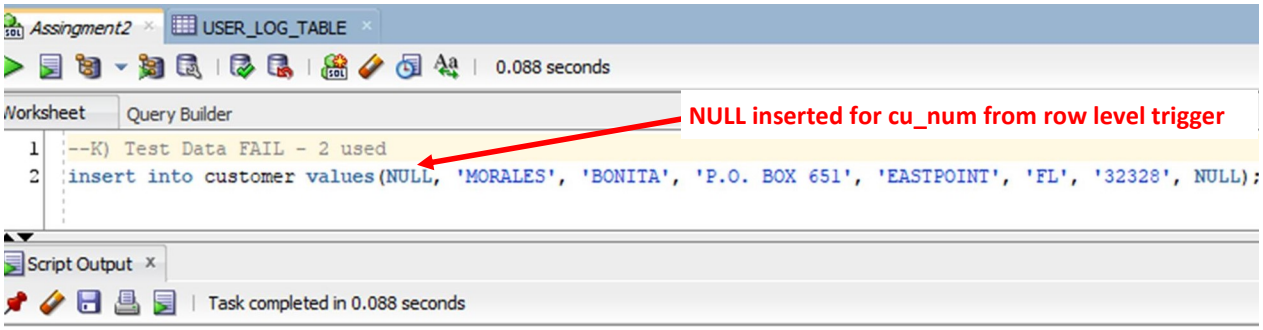
Worksheet | Query Builder

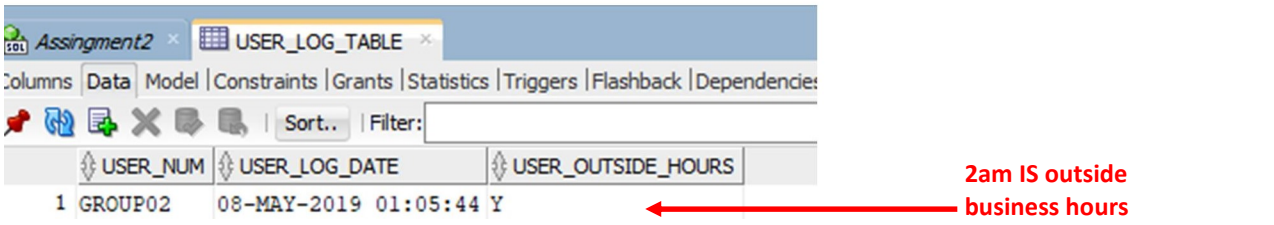
```
1  -- k) STATEMENT LEVEL TRIGGER This trigger logs which users inserts, updates or deletes a value in the
2  -- customer table in the user_log_table. It inserts the user ID and a current datetime stamp
3  -- when the customer table was altered. If the customer table is altered outside of normal
4  -- business hours, user information is still inserted into the user_log_table and a flag (Y)
5  -- is logged in the 'USER_OUTSIDE_HOURS' column.
6  -- The business hours are all week days between 9am and 4pm.
7
8  DROP TABLE user_log_table; --Wouldn't be used in final version.
9
10 CREATE TABLE user_log_table(
11     user_num VARCHAR2(60 BYTE) NOT NULL,
12     user_log_date VARCHAR2(60),
13     user_outside_hours VARCHAR2(1)
14 );
15
16 CREATE OR REPLACE TRIGGER tr_log_customer
17 BEFORE INSERT OR UPDATE OR DELETE ON customer
18 DECLARE
19     cur_date_temp VARCHAR2(10) := TO_CHAR(SYSDATE, 'Day');--'Sunday'; Used to test (outside business hours)
20     cur_date VARCHAR2(3);
21     cur_time INTEGER := TO_NUMBER(TO_CHAR(SYSDATE, 'HH24')); --2;-- Used to test (outside business hours)
22 BEGIN
23 cur_date := SUBSTR(cur_date_temp, 0, 3);
24 IF (cur_date != 'Sat' AND cur_date != 'Sun') AND (cur_time >= 9 AND cur_time <= 16) THEN
25     INSERT INTO user_log_table VALUES(USER, TO_CHAR(SYSDATE, 'DD-MON-YYYY HH:MM:SS'), 'N');
26 ELSE
27     INSERT INTO user_log_table VALUES(USER, TO_CHAR(SYSDATE, 'DD-MON-YYYY HH:MM:SS'), 'Y');
28 END IF;
29 END;
30 /
31 show error
32
```

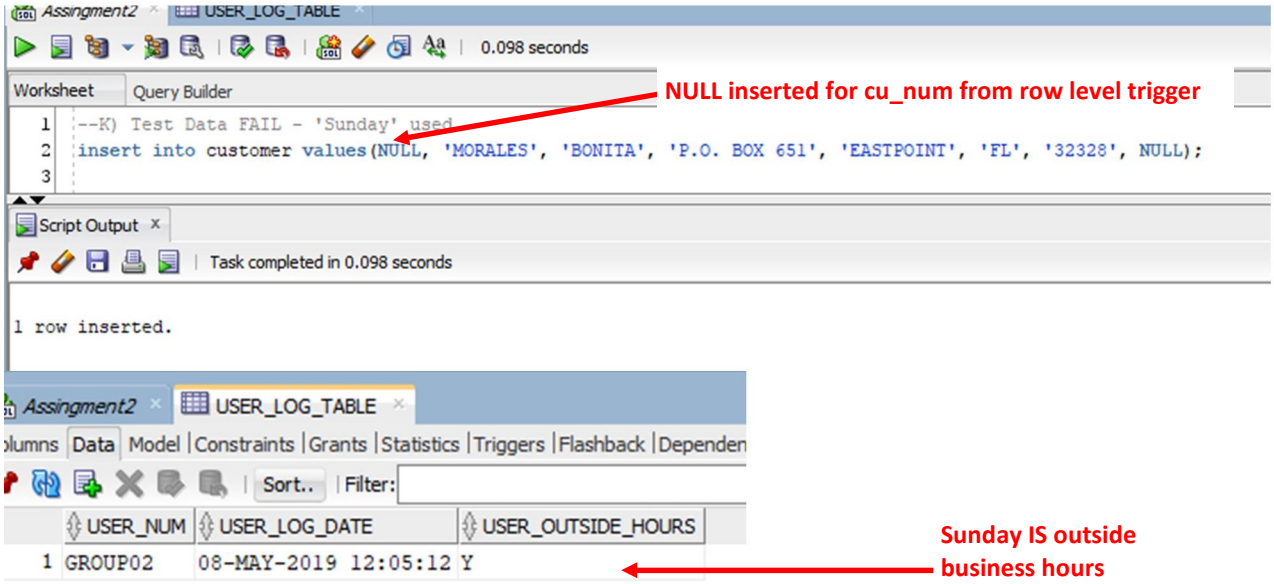
Script Output x | Query Result x

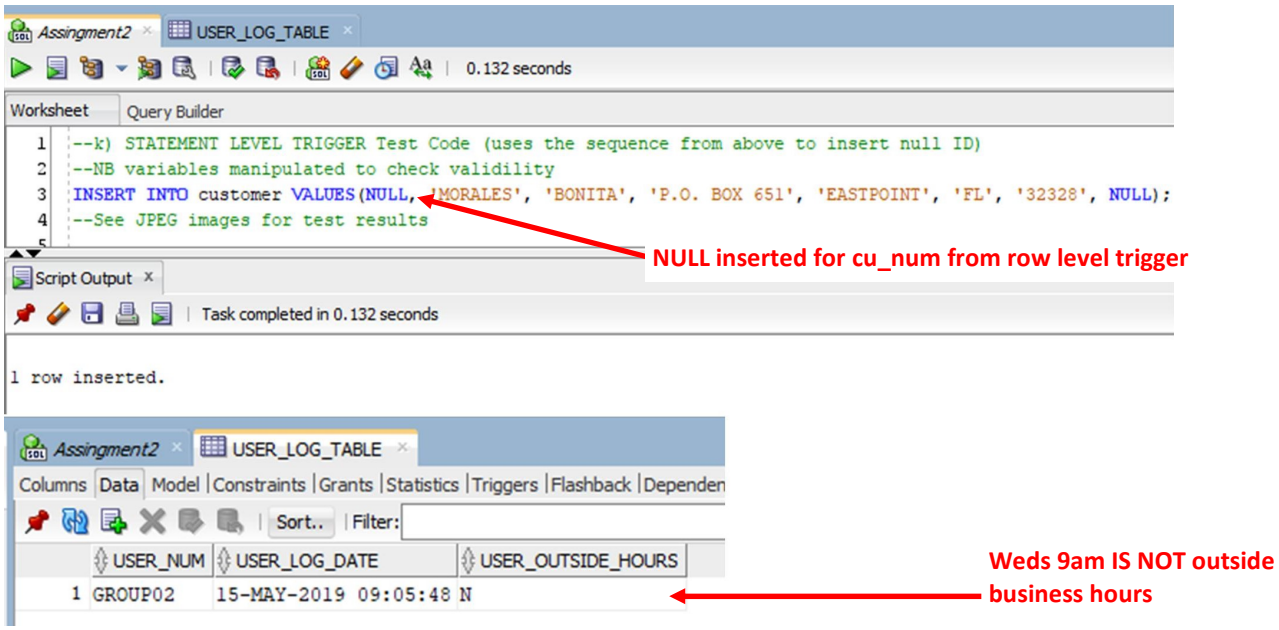
Task completed in 0.301 seconds

```
Table USER_LOG_TABLE dropped.


Table USER_LOG_TABLE created.


Trigger TR_LOG_CUSTOMER compiled

No errors.
```


Assignment2
USER_LOG_TABLE
0.088 seconds
Worksheet
Query Builder
NULL inserted for cu_num from row level trigger
--K) Test Data FAIL - 2 used
insert into customer values(NULL, 'MORALES', 'BONITA', 'P.O. BOX 651', 'EASTPOINT', 'FL', '32328', NULL);
Script Output
Task completed in 0.088 seconds
1 row inserted.


Assignment2
USER_LOG_TABLE
Columns
Data
Model
Constraints
Grants
Statistics
Triggers
Flashback
Dependencies
Sort..
Filter:
USER_NUM
USER_LOG_DATE
USER_OUTSIDE_HOURS
1 GROUP02 08-MAY-2019 01:05:44 Y
2am IS outside business hours


Assignment2
USER_LOG_TABLE
0.098 seconds
Worksheet
Query Builder
NULL inserted for cu_num from row level trigger
--K) Test Data FAIL - 'Sunday' used
insert into customer values(NULL, 'MORALES', 'BONITA', 'P.O. BOX 651', 'EASTPOINT', 'FL', '32328', NULL);
Script Output
Task completed in 0.098 seconds
1 row inserted.
Assignment2
USER_LOG_TABLE
Data
Model
Constraints
Grants
Statistics
Triggers
Flashback
Sort..
Filter:
USER_NUM
USER_LOG_DATE
USER_OUTSIDE_HOURS
1 GROUP02 08-MAY-2019 12:05:12 Y
Sunday IS outside business hours


Assignment2
USER_LOG_TABLE
0.132 seconds
Worksheet
Query Builder
--k) STATEMENT LEVEL TRIGGER Test Code (uses the sequence from above to insert null ID)
--NB variables manipulated to check validility
INSERT INTO customer VALUES(NULL, 'MORALES', 'BONITA', 'P.O. BOX 651', 'EASTPOINT', 'FL', '32328', NULL);
--See JPEG images for test results
NULL inserted for cu_num from row level trigger
Script Output
Task completed in 0.132 seconds
1 row inserted.
Assignment2
USER_LOG_TABLE
Columns
Data
Model
Constraints
Grants
Statistics
Triggers
Flashback
Sort..
Filter:
USER_NUM
USER_LOG_DATE
USER_OUTSIDE_HOURS
1 GROUP02 15-MAY-2019 09:05:48 N
Weds 9am IS NOT outside business hours

```
Worksheet  Query Builder
1  --1)
2  --Write a procedure to insert a new book record. The procedure should also automatically calculate the book retail value.
3  --This retail is calculated as 112.5% of the book cost price plus 8.5% of the average cost price of the existing books.
4  --Provide rest of the attributes' values as input parameters. Execute your procedure to insert at least one book record. (3 marks)
5  CREATE OR REPLACE PROCEDURE InsertBookRecord
6  (b_isbn IN VARCHAR2, b_title IN VARCHAR2, b_pubdate IN DATE, b_pubid IN NUMBER, b_cost IN NUMBER, b_category IN VARCHAR2)
7  IS
8      b_retail NUMBER(5,2);
9  BEGIN
10     SELECT 0.085*(AVG(book_cost))
11     INTO b_retail
12     FROM BOOK;
13     b_retail := b_retail +( 1.125*b_cost);
14
15     INSERT INTO book(book_isbn, book_title, book_pubdate, book_pubid, book_cost, book_retail, book_category)
16     VALUES (b_isbn, b_title, b_pubdate, b_pubid, b_cost, b_retail, b_category);
17
18 EXCEPTION
19     WHEN OTHERS THEN
20     dbms_output.put_line('Invlaid data entry. REASON: '||SQLERRM);
21 END;
22 /
23 show error
```

Script Output × | Query Result ×

Task completed in 0.698 seconds

```
rocedure INSERTBOOKRECORD compiled

o errors.
```

Worksheet | Query Builder

```
1  --Test Code --PASS --Correctly increases RRP
2  EXECUTE InsertBookRecord(98141269, 'Database Theory', TO_DATE('19-04-2019', 'DD-MM-YYYY'), 4, 20, 'business' );
```

Script Output × | Query Result ×

Task completed in 0.096 seconds

```
PL/SQL procedure successfully completed.
```

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. | Filter:

	BOOK_ISBN	BOOK_TITLE	BOOK_PUBDATE	BOOK_PUBID	BOOK_COST	BOOK_RETAIL	BOOK_CATEGORY
1	9247381001	HOW TO MANAGE THE MANAGER	09-MAY-99	1	15.4	31.95	BUSINESS
2	0401140733	REVENGE OF MICKEY	14-DEC-01	1	14.2	22	FAMILY LIFE
3	4981341710	BUILDING A CAR WITH TOOTHPICKS	18-MAR-02	2	37.8	59.95	CHILDREN
4	9959789321	E-BUSINESS THE EASY WAY	01-MAR-02	2	37.9	54.5	COMPUTER
5	8843172113	DATABASE IMPLEMENTATION	04-JUN-99	3	31.4	55.95	COMPUTER
6	3957136468	HOLY GRAIL OF ORACLE	31-DEC-01	3	47.25	75.95	COMPUTER
7	1915762492	HANDCRANKED COMPUTERS	21-JAN-01	3	21.8	25	COMPUTER
8	98141269	Database Theory	19-APR-19	4	20	24.62	business
9	3437212490	COOKING WITH MUSHROOMS	28-FEB-00	4	12.5	19.95	COOKING
10	0299282519	THE WOK WAY TO COOK	11-SEP-00	4	19	28.75	COOKING
11	1059831198	BODYBUILD IN 10 MINUTES A DAY	21-JAN-01	4	18.75	30.95	FITNESS

```
Worksheet  Query Builder
 1  --m. Write a trigger that does not allow the book retail price to be updated when the increase (in retail price) is over 25%.
 2  --Provide test data and corresponding results to confirm that the trigger works. (4 marks)
 3  CREATE OR REPLACE TRIGGER retail_price_update
 4  BEFORE UPDATE OF book_retail ON book
 5  FOR EACH ROW
 6  DECLARE
 7      max_price exception;
 8  BEGIN
 9      IF :NEW.book_retail > (1.25*:OLD.book_retail)
10      THEN
11      RAISE max_price;
12      END IF;
13  EXCEPTION
14  WHEN max_price THEN
15      raise_application_error(-20020, 'Price increase is over 25%');
16  END;
17  /
18  show error
```

Script Output | Query Result | Task completed in 0.172 seconds

```
Trigger RETAIL_PRICE_UPDATE compiled

No errors.
```

Worksheet | Query Builder

```
1  --Test Code FAIL - Book retail increase over 25%
2  UPDATE book SET book_retail = book_retail*1.27 WHERE book_isbn = 0401140733;
3
4  --Test Code PASS
5  UPDATE book SET book_retail = book_retail*1.2 WHERE book_isbn = 0401140733;
```

Script Output | Query Result | Task completed in 0.138 seconds

```
Error starting at line : 2 in command -
UPDATE book SET book_retail = book_retail*1.27 WHERE book_isbn = 0401140733
Error report -
ORA-20020: Price increase is over 25%
ORA-06512: at "GROUP02.RETAIL_PRICE_UPDATE", line 10
ORA-04088: error during execution of trigger 'GROUP02.RETAIL_PRICE_UPDATE'


1 row updated.
```

Assignment2 | BOOK

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. | Filter:

	BOOK_ISBN	BOOK_TITLE	BOOK_PUBDATE	BOOK_PUBID	BOOK_COST	BOOK_RETAIL	BOOK_CATEGORY
1	9247381001	HOW TO MANAGE THE MANAGER	09-MAY-99	1	15.4	31.95	BUSINESS
2	0401140733	REVENGE OF MICKEY	14-DEC-01	1	14.2	26.4	FAMILY LIFE
3	4981341710	BUILDING A CAR WITH TOOTHPICKS	18-MAR-02	2	37.8	59.95	CHILDREN
4	9959789321	E-BUSINESS THE EASY WAY	01-MAR-02	2	37.9	54.5	COMPUTER
5	8843172113	DATABASE IMPLEMENTATION	04-JUN-99	3	31.4	55.95	COMPUTER
6	3957136468	HOLY GRAIL OF ORACLE	31-DEC-01	3	47.25	75.95	COMPUTER

Assingment2 | USER_LOG_TABLE

0.22499999 seconds

Worksheet | Query Builder

```
1  --n. Write a trigger that does not allow more than three author names to be associated
2  --with books under FITNESS category e.g. if a Book is added, it should only allow up to
3  --3 book authors to be recorded in BookAuthor table for category FITNESS books).
4  --Provide the appropriate test data and results. (4 marks)
5
6  create or replace TRIGGER max_3_author_update
7  BEFORE INSERT OR UPDATE  ON bookauthor
8  FOR EACH ROW
9  DECLARE
10     max_3_authors EXCEPTION;
11     num_authors NUMBER;
12 BEGIN
13     SELECT COUNT(*)
14     INTO num_authors
15     FROM book b, bookauthor ba
16     WHERE b.book_isbn = ba.ba_isbn
17     AND b.book_category = 'FITNESS';
18     IF num_authors >2
19     THEN
20         RAISE max_3_authors;
21     END IF;
22 EXCEPTION
23 WHEN max_3_authors THEN
24     raise_application_error(-20030, 'Only 3 authors allowed within FITNESS category');
25     END
26 ;
27 /
28 show error
```

Script Output | Task completed in 0.225 seconds

```
Trigger MAX_3_AUTHOR_UPDATE compiled

No errors.
```

Worksheet | Query Builder

```
1  INSERT INTO bookauthor VALUES('1059831198','J100');
2  INSERT INTO bookauthor VALUES('1059831198','K100');
3  INSERT INTO bookauthor VALUES('1059831198','P105');
```

Script Output | Task completed in 0.107 seconds

```
1 row inserted.


1 row inserted.


Error starting at line : 3 in command -
INSERT INTO bookauthor VALUES('1059831198','P105')
Error report -
ORA-20030: Only 3 authors allowed within FITNESS category
ORA-06512: at "GROUP02.MAX_3_AUTHOR_UPDATE", line 16
ORA-04088: error during execution of trigger 'GROUP02.MAX_3_AUTHOR_UPDATE'
```

BOOKAUTHOR BEFORE INSERT

BOOKAUTHOR

Columns | Data | Model | Constraints | Grants

	BA_ISBN	BA_AUTHORID
1	0132149871	S100
2	0299282519	S100
3	0401140733	J100
4	1059831198	S100
5	1915762492	W105
6	1915762492	W100
7	2147428890	W105

BOOKAUTHOR AFTER INSERT

BOOKAUTHOR

Columns | Data | Model | Constraints | Grants

	BA_ISBN	BA_AUTHORID
1	0132149871	S100
2	0299282519	S100
3	0401140733	J100
4	1059831198	S100
5	1059831198	K100
6	1059831198	J100
7	1915762492	W105
8	1915762492	W100
9	2147428890	W105

Worksheet | Query Builder

```
1  set serveroutput on
2  --o. Write a cursor to list book authors for all the COMPUTER category books (along with their book title, cost and retail).
3  --Use appropriate exception handling. (3 marks)
4  DECLARE
5  CURSOR book_cursor IS
6      SELECT b.book_title, a.author_fname, a.author_lname, b.book_cost, b.book_retail
7      FROM book b, author a, bookauthor ba
8      WHERE b.book_isbn = ba.ba_isbn
9      AND ba.ba_authorid = a.author_id
10     AND LOWER(book_category) = LOWER('COMPUTER');
11
12 BEGIN
13 FOR book IN book_cursor
14 LOOP
15     dbms_output.put_line(book.book_title || ','|| ' Author: '|| book.author_fname || ' '||
16         book.author_lname ||' Cost: ' || book.book_cost || ' RRP: '|| book.book_retail);
17     END LOOP;
18 EXCEPTION
19     WHEN OTHERS
20     THEN
21         IF book_cursor%ISOPEN
22         THEN CLOSE book_cursor;
23         END IF;
24 END;
25 /
```

Script Output | Query Result | Task completed in 0.083 seconds

```
DATABASE IMPLEMENTATION, Author: JAMES AUSTIN Cost: 31.4 RRP: 55.95
DATABASE IMPLEMENTATION, Author: JUAN ADAMS Cost: 31.4 RRP: 55.95
DATABASE IMPLEMENTATION, Author: TINA PETERSON Cost: 31.4 RRP: 55.95
HOLY GRAIL OF ORACLE, Author: JAMES AUSTIN Cost: 47.25 RRP: 75.95
HANDCRANKED COMPUTERS, Author: WILLIAM WHITE Cost: 21.8 RRP: 25
HANDCRANKED COMPUTERS, Author: LISA WHITE Cost: 21.8 RRP: 25
E-BUSINESS THE EASY WAY, Author: JANICE JONES Cost: 37.9 RRP: 54.5


PL/SQL procedure successfully completed.
```

Worksheet | Query Builder

```
1  --p. Write a function to format book cost, retail price to $99.99. Use this function in a SQL statement
2  --for displaying books' costs and retail prices. (2 marks)
3  CREATE OR REPLACE
4  FUNCTION FormatBookCost(b_cost NUMBER)
5  RETURN VARCHAR2
6  IS
7  book_price VARCHAR2(10);
8  BEGIN
9  SELECT TO_CHAR(b_cost, '$99.99')
10 INTO book_price
11 FROM dual;
12 RETURN book_price;
13 END;
```

Script Output | Query Result | Task completed in 0.105 seconds

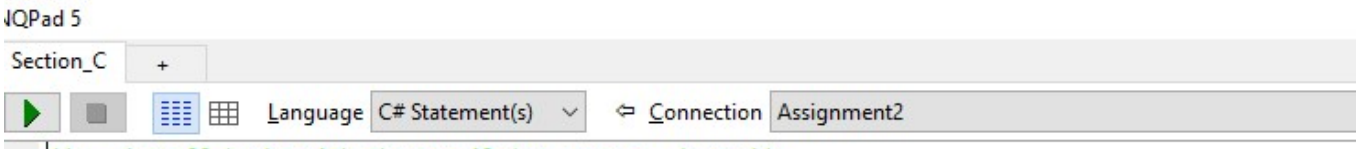
```
Function FORMATBOOKCOST compiled
```

Worksheet | Query Builder

```
1  --Test code
2  SELECT book_title "Book Title", FormatBookCost(book_cost) "Book Cost", FormatBookCost(book_retail) "Book Retail"
3  FROM book;
```

Script Output | Query Result | SQL | All Rows Fetched: 15 in 0.022 seconds

	Book Title	Book Cost	Book Retail
1	BODYBUILD IN 10 MINUTES A DAY	$18.75	$30.95
2	REVENGE OF MICKEY	$14.20	$26.40
3	BUILDING A CAR WITH TOOTHPICKS	$37.80	$59.95
4	DATABASE IMPLEMENTATION	$31.40	$55.95
5	COOKING WITH MUSHROOMS	$12.50	$19.95
6	HOLY GRAIL OF ORACLE	$47.25	$75.95
7	HANDCRANKED COMPUTERS	$21.80	$25.00




```
//q. List all books with the retail $22 or over. (1 mark)
var over22 =
    from b in Books
    where b.BookRetail >= 22
    select b;

over22.Dump();

//r. List all books that have the word "HOW" in the book title. (1 mark)
var containsHow =
    from b in Books
    where b.BookTitle.Contains("HOW")
    select b;

containsHow.Dump();

//s. List all book categories, book titles and publisher names. (1 mark)
var BookDetails =
    from b in Books join p in Publishers
    on b.BookPubid equals p.PubID
    select new{
    BookTitle = b.BookTitle,
    Category = b.BookCategory,
    Publisher = p.PubName};
BookDetails.Dump();

//t. Display total number of books by each publisher in the order of publisher ID. (1.5 marks)
var BookTotalByPublisher =
    from b in Books join p in Publishers
    on b.BookPubid equals p.PubID
    group b.BookPubid  by new{
        ID = b.BookPubid,
        Name = p.PubName
    }
    into T
    select new{
        ID = T.Key.ID,
        PublisherName = T.Key.Name,
        Count = T.Count()
    };
BookTotalByPublisher.Dump();

//u. Display the count of books in each category in the increasing order of category. (1.5 marks)
//Im assuming the above means in alphabetical order of the category.
var BooksInCategory =
    from b in Books
    group b.BookCategory by
    new{
        Category = b.BookCategory
    }into T
    orderby T.Key.Category ascending
    select new{
        Category = T.Key.Category,
        Count = T.Count()
    };

BooksInCategory.Dump();
```

Results are in question alphabetical order (q, r, s, t, u)

Query<Book> (12 items)

BookIsbn	BookTitle	BookPubdate	BookPubid	BookCost	BookRetail	BookCategory	BoiIsbnBookorderitems	BaIsbnBookauthors
1059831198	BODYBUILD IN 10 MINUTES A DAY	21-Jan-01 12:00:00 AM	4	18.75	30.95	FITNESS	BoiIsbnBookorderitems	BaIsbnBookauthors
0401140733	REVENGE OF MICKEY	14-Dec-01 12:00:00 AM	1	14.2	22	FAMILY LIFE	BoiIsbnBookorderitems	BaIsbnBookauthors
4981341710	BUILDING A CAR WITH TOOTHPICKS	18-Mar-02 12:00:00 AM	2	37.8	59.95	CHILDREN	BoiIsbnBookorderitems	BaIsbnBookauthors
8843172113	DATABASE IMPLEMENTATION	04-Jun-99 12:00:00 AM	3	31.4	55.95	COMPUTER	BoiIsbnBookorderitems	BaIsbnBookauthors
3957136468	HOLY GRAIL OF ORACLE	31-Dec-01 12:00:00 AM	3	47.25	75.95	COMPUTER	BoiIsbnBookorderitems	BaIsbnBookauthors
1915762492	HANDCRANKED COMPUTERS	21-Jan-01 12:00:00 AM	3	21.8	25	COMPUTER	BoiIsbnBookorderitems	BaIsbnBookauthors
9959789321	E-BUSINESS THE EASY WAY	01-Mar-02 12:00:00 AM	2	37.9	54.5	COMPUTER	BoiIsbnBookorderitems	BaIsbnBookauthors
2491748320	PAINLESS CHILD-REARING	17-Jul-00 12:00:00 AM	5	48	89.95	FAMILY LIFE	BoiIsbnBookorderitems	BaIsbnBookauthors
0299282519	THE WOK WAY TO COOK	11-Sep-00 12:00:00 AM	4	19	28.75	COOKING	BoiIsbnBookorderitems	BaIsbnBookauthors
0132149871	HOW TO GET FASTER PIZZA	11-Nov-02 12:00:00 AM	4	17.85	29.95	SELF HELP	BoiIsbnBookorderitems	BaIsbnBookauthors
9247381001	HOW TO MANAGE THE MANAGER	09-May-99 12:00:00 AM	1	15.4	31.95	BUSINESS	BoiIsbnBookorderitems	BaIsbnBookauthors
2147428890	SHORTEST POEMS	01-May-01 12:00:00 AM	5	21.85	39.95	LITERATURE	BoiIsbnBookorderitems	BaIsbnBookauthors
			37	331.20	544.85			

Query<Book> (2 items)

BookIsbn	BookTitle	BookPubdate	BookPubid	BookCost	BookRetail	BookCategory	BoiIsbnBookorderitems	BaIsbnBookauthors
0132149871	HOW TO GET FASTER PIZZA	11-Nov-02 12:00:00 AM	4	17.85	29.95	SELF HELP	BoiIsbnBookorderitems	BaIsbnBookauthors
9247381001	HOW TO MANAGE THE MANAGER	09-May-99 12:00:00 AM	1	15.4	31.95	BUSINESS	BoiIsbnBookorderitems	BaIsbnBookauthors
			5	33.25	61.90			

Query<> (14 items)

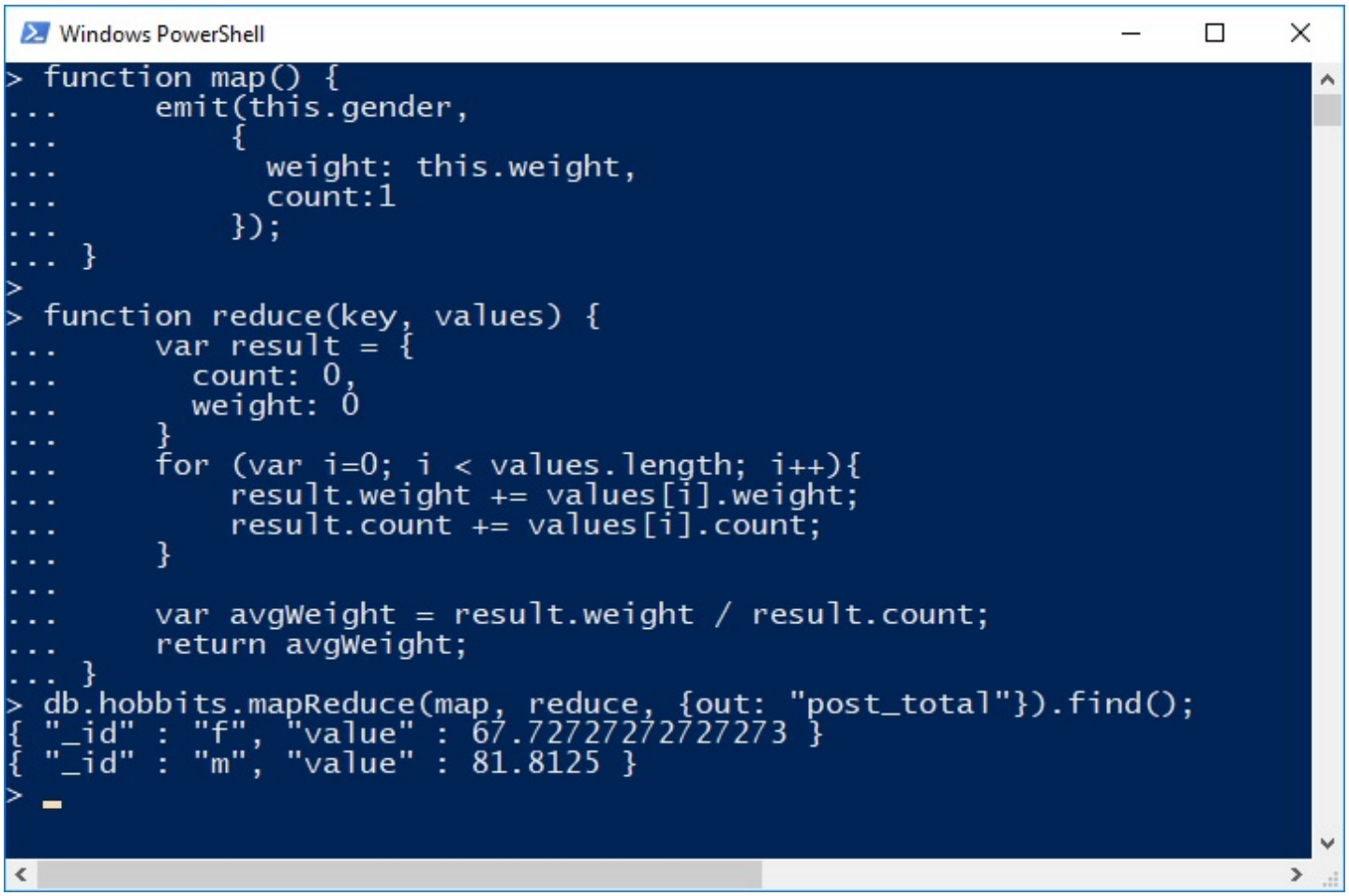
BookTitle	Category	Publisher
BODYBUILD IN 10 MINUTES A DAY	FITNESS	READING MATERIALS INC.
REVENGE OF MICKEY	FAMILY LIFE	PRINTING IS US
BUILDING A CAR WITH TOOTHPICKS	CHILDREN	PUBLISH OUR WAY
DATABASE IMPLEMENTATION	COMPUTER	AMERICAN PUBLISHING
COOKING WITH MUSHROOMS	COOKING	READING MATERIALS INC.
HOLY GRAIL OF ORACLE	COMPUTER	AMERICAN PUBLISHING
HANDCRANKED COMPUTERS	COMPUTER	AMERICAN PUBLISHING
E-BUSINESS THE EASY WAY	COMPUTER	PUBLISH OUR WAY
PAINLESS CHILD-REARING	FAMILY LIFE	REED-N-RITE
THE WOK WAY TO COOK	COOKING	READING MATERIALS INC.
BIG BEAR AND LITTLE DOVE	CHILDREN	REED-N-RITE
HOW TO GET FASTER PIZZA	SELF HELP	READING MATERIALS INC.
HOW TO MANAGE THE MANAGER	BUSINESS	PRINTING IS US
SHORTEST POEMS	LITERATURE	REED-N-RITE

Query<> (5 items)

ID	PublisherName	Count
1	PRINTING IS US	2
3	AMERICAN PUBLISHING	3
4	READING MATERIALS INC.	4
5	REED-N-RITE	3
2	PUBLISH OUR WAY	2
		14

Query<> (8 items)

Category	Count
BUSINESS	1
CHILDREN	2
COMPUTER	4
COOKING	2
FAMILY LIFE	2
FITNESS	1
LITERATURE	1
SELF HELP	1
	14


Windows PowerShell
> function map() {
... emit(this.gender,
... {
... weight: this.weight,
... count:1
... });
... }
>
> function reduce(key, values) {
... var result = {
... count: 0,
... weight: 0
... }
... for (var i=0; i < values.length; i++){
... result.weight += values[i].weight;
... result.count += values[i].count;
... }
...
... var avgWeight = result.weight / result.count;
... return avgWeight;
... }
> db.hobbits.mapReduce(map, reduce, {out: "post_total"}).find();
{ "_id" : "f", "value" : 67.72727272727273 }
{ "_id" : "m", "value" : 81.8125 }
>