Section A

a)

account: account.id

codeCat: codeCat.category

spend: spend.merchant, spending.transactionDate

b)

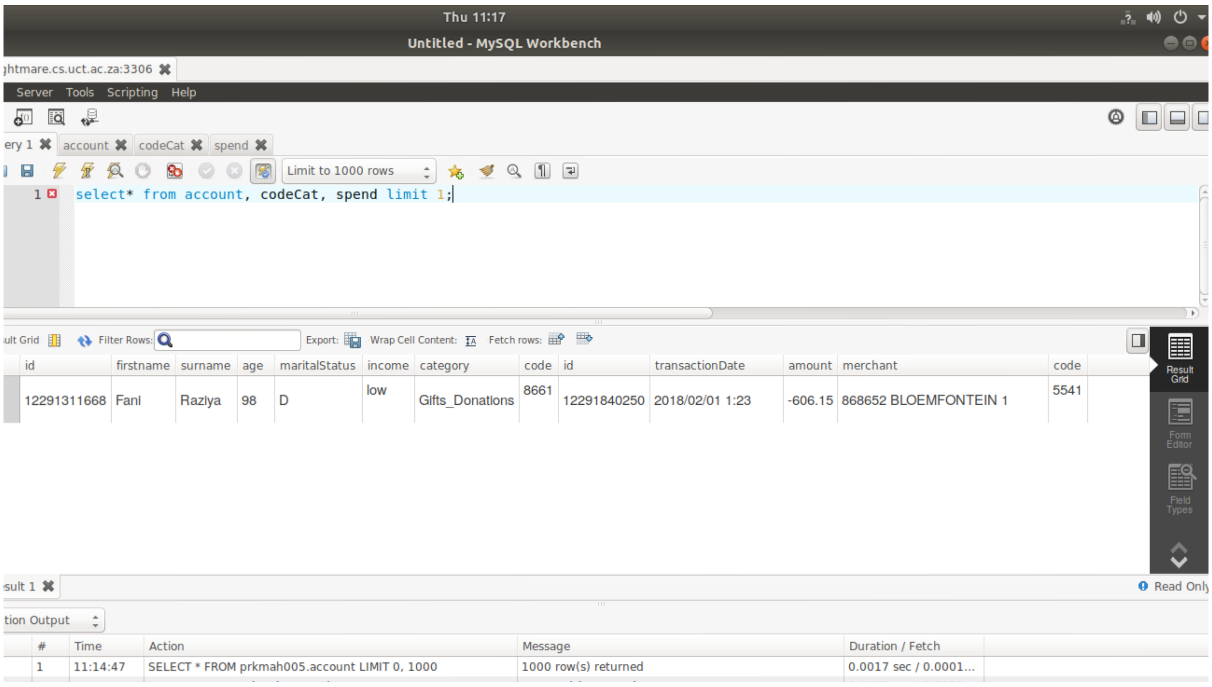
account: spending.id

codeCat: none

spend: account.id, codeCat.code

Section B

1. Show any one complete tuple (just 1, no more) from each of your 3 relations.

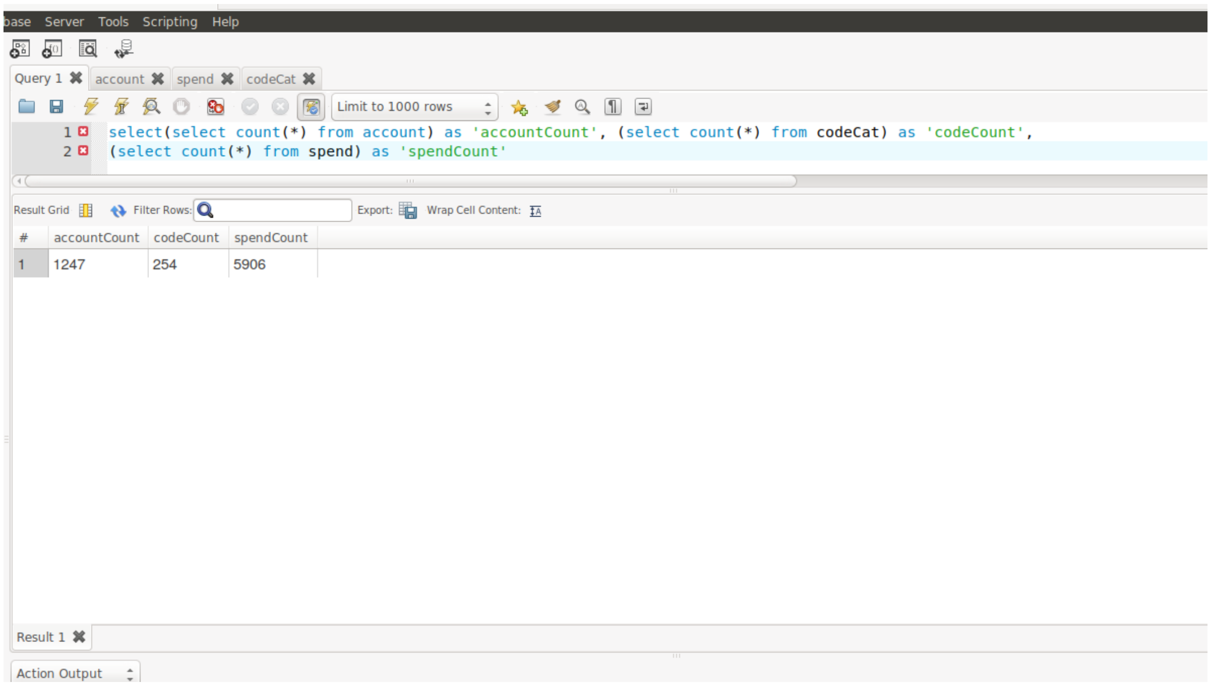


```
select* from account, codeCat, spend limit 1;
```

id	firstname	surname	age	maritalStatus	income	category	code	id	transactionDate	amount	merchant	code
12291311668	Fani	Raziya	98	D	low	Gifts_Donations	8661	12291840250	2018/02/01 1:23	-606.15	868652 BLOEMFONTEIN 1	5541

#	Time	Action	Message	Duration / Fetch
1	11:14:47	SELECT * FROM prkmah005.account LIMIT 0, 1000	1000 row(s) returned	0.0017 sec / 0.0001...

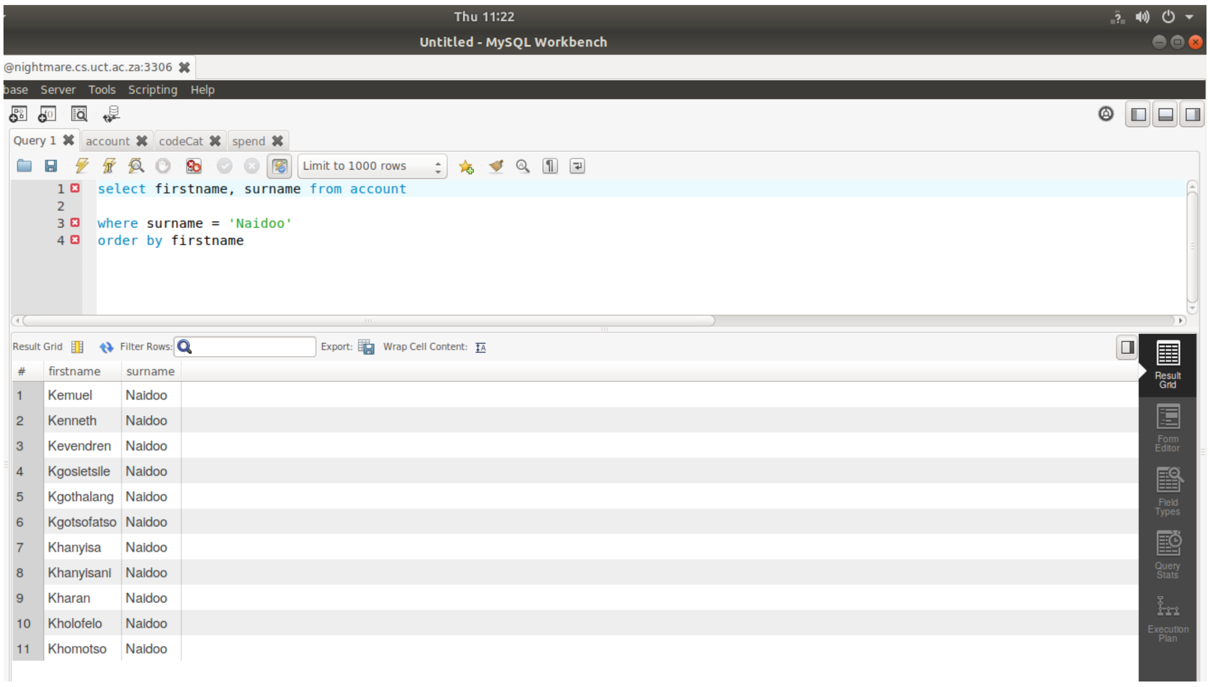
2. Output the number of tuples in each of your 3 relations.



```
select(select count(*) from account) as 'accountCount', (select count(*) from codeCat) as 'codeCount',
(select count(*) from spend) as 'spendCount'
```

#	accountCount	codeCount	spendCount
1	1247	254	5906

3. List first name and surname of everyone with surname “Naidoo”, in alphabetical order of first name.

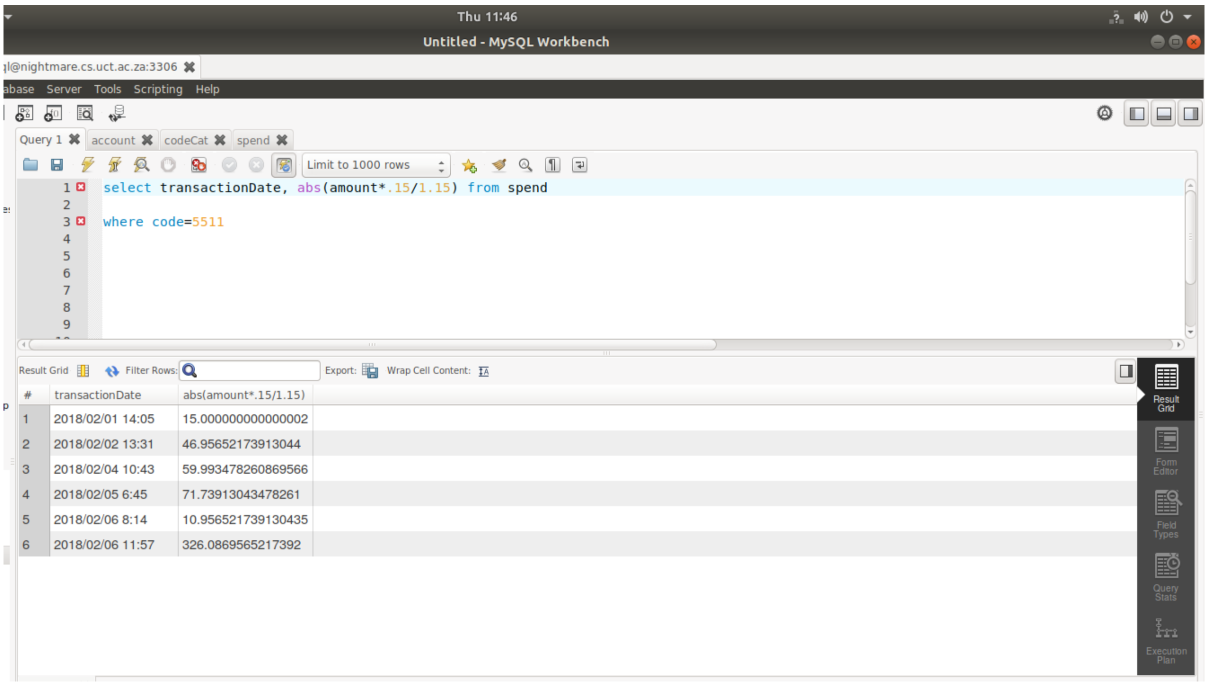


```
select firstname, surname from account

where surname = 'Naidoo'
order by firstname
```

#	firstname	surname
1	Kemuel	Naidoo
2	Kenneth	Naidoo
3	Kevendren	Naidoo
4	Kgosietsile	Naidoo
5	Kgothalang	Naidoo
6	Kgotsofatso	Naidoo
7	Khanyisa	Naidoo
8	Khanyisani	Naidoo
9	Kharan	Naidoo
10	Kholofelo	Naidoo
11	Khomotso	Naidoo

4. For each code 5511 transaction, give the transactionDate along with the VAT on that transation, where the VAT was 15% of the amount spent (e.g. if the amount was -115 then the VAT was 15).

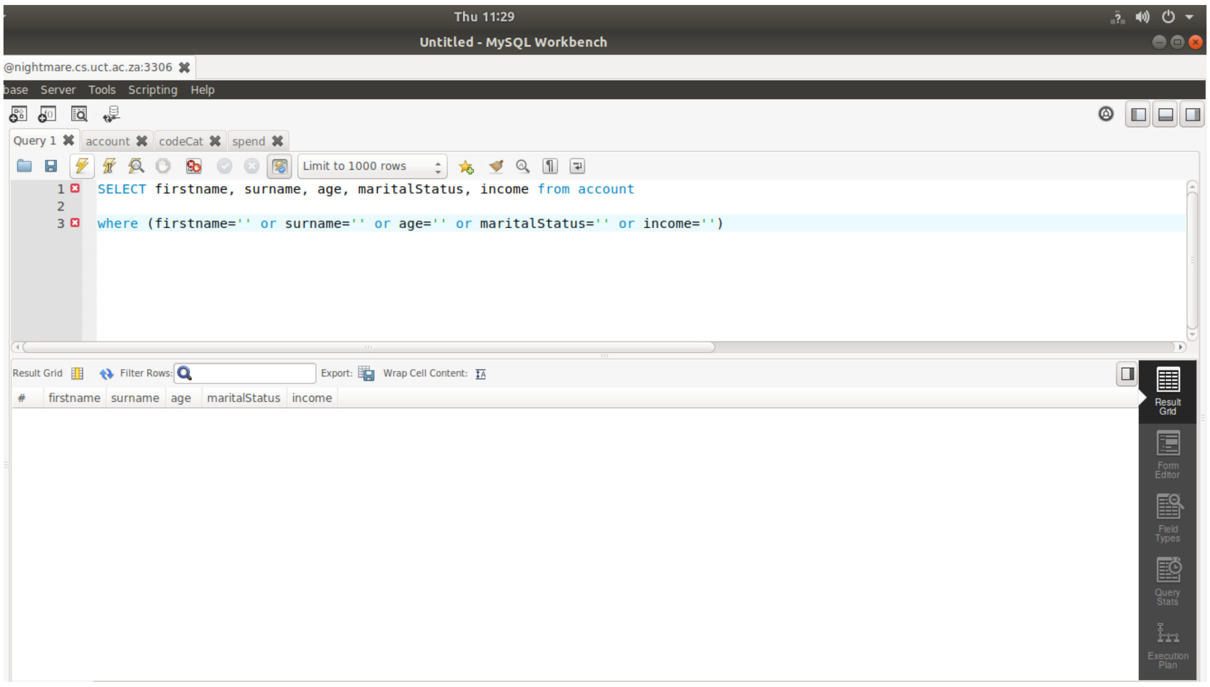


```
select transactionDate, abs(amount*.15/1.15) from spend

where code=5511
```

#	transactionDate	abs(amount*.15/1.15)
1	2018/02/01 14:05	15.000000000000002
2	2018/02/02 13:31	46.95652173913044
3	2018/02/04 10:43	59.993478260869566
4	2018/02/05 6:45	71.73913043478261
5	2018/02/06 8:14	10.956521739130435
6	2018/02/06 11:57	326.0869565217392

5. Show the firstname, surname, age, marital-status and income category of everyone who has any missing data in any of these fields.

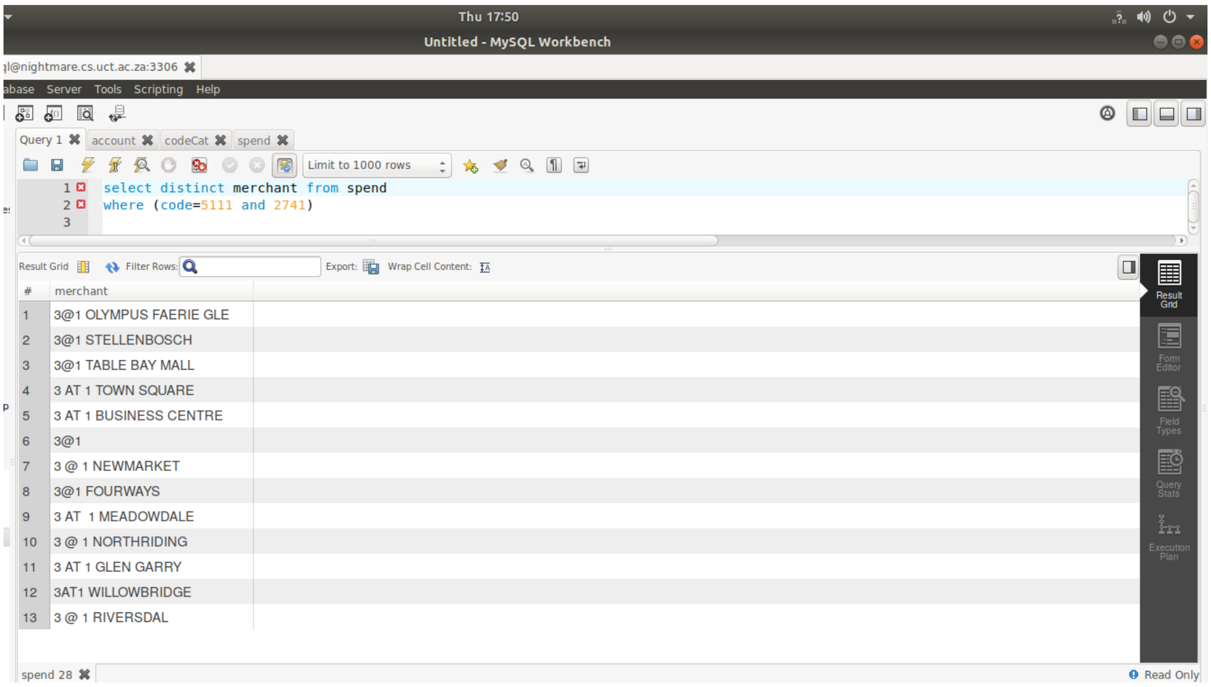


```
SELECT firstname, surname, age, maritalStatus, income from account

where (firstname='' or surname='' or age='' or maritalStatus='' or income='')
```

#	firstname	surname	age	maritalStatus	income

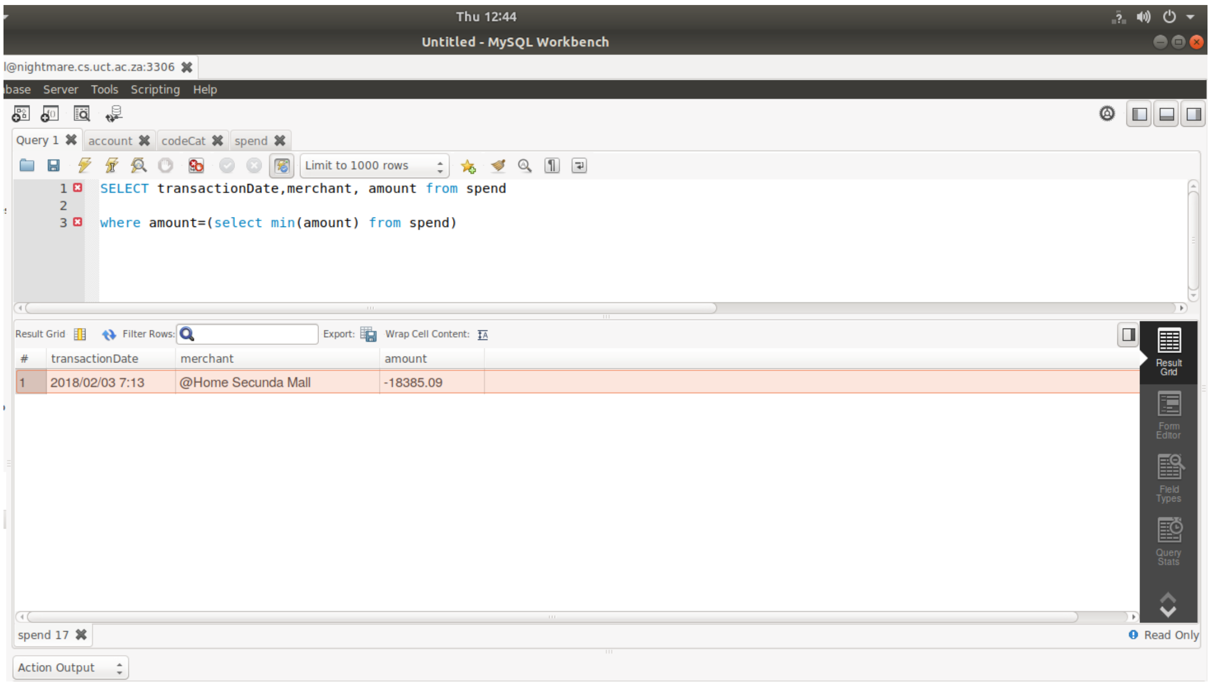
6. Give the names of merchants who have had transactions of both codes 5111 and 2741.



```
select distinct merchant from spend
where (code=5111 and 2741)
```

```
#   merchant
1   3@1 OLYMPUS FAERIE GLE
2   3@1 STELLENBOSCH
3   3@1 TABLE BAY MALL
4   3 AT 1 TOWN SQUARE
5   3 AT 1 BUSINESS CENTRE
6   3@1
7   3 @ 1 NEWMARKET
8   3@1 FOURWAYS
9   3 AT  1 MEADOWDALE
10  3 @ 1 NORTHRIDING
11  3 AT 1 GLEN GARRY
12  3AT1 WILLOWBRIDGE
13  3 @ 1 RIVERSDAL
```

7. Find the transactionDate and merchant associated with the highest amount in the database, along with that amount.

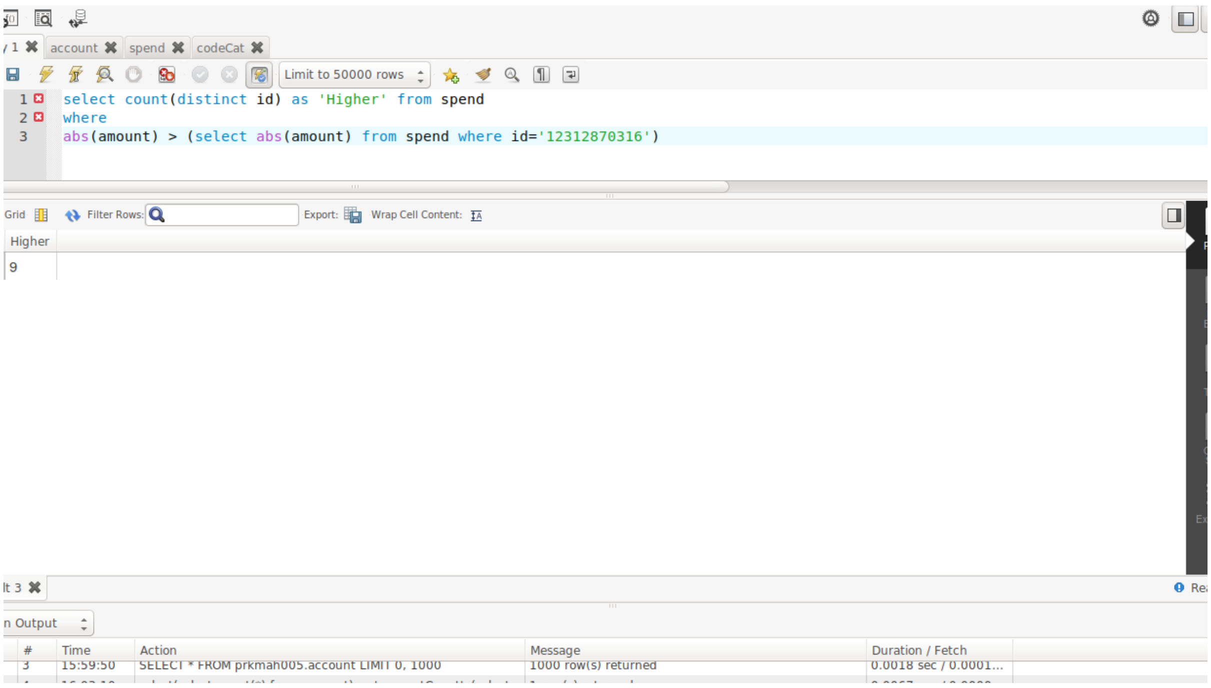


```
SELECT transactionDate,merchant, amount from spend

where amount=(select min(amount) from spend)
```

```
#  transactionDate   merchant             amount
1  2018/02/03 7:13   @Home Secunda Mall   -18385.09
```

8. How many people (ids) have any transaction amount that is higher than every amount ever spent by the person with id 12312870316 ?

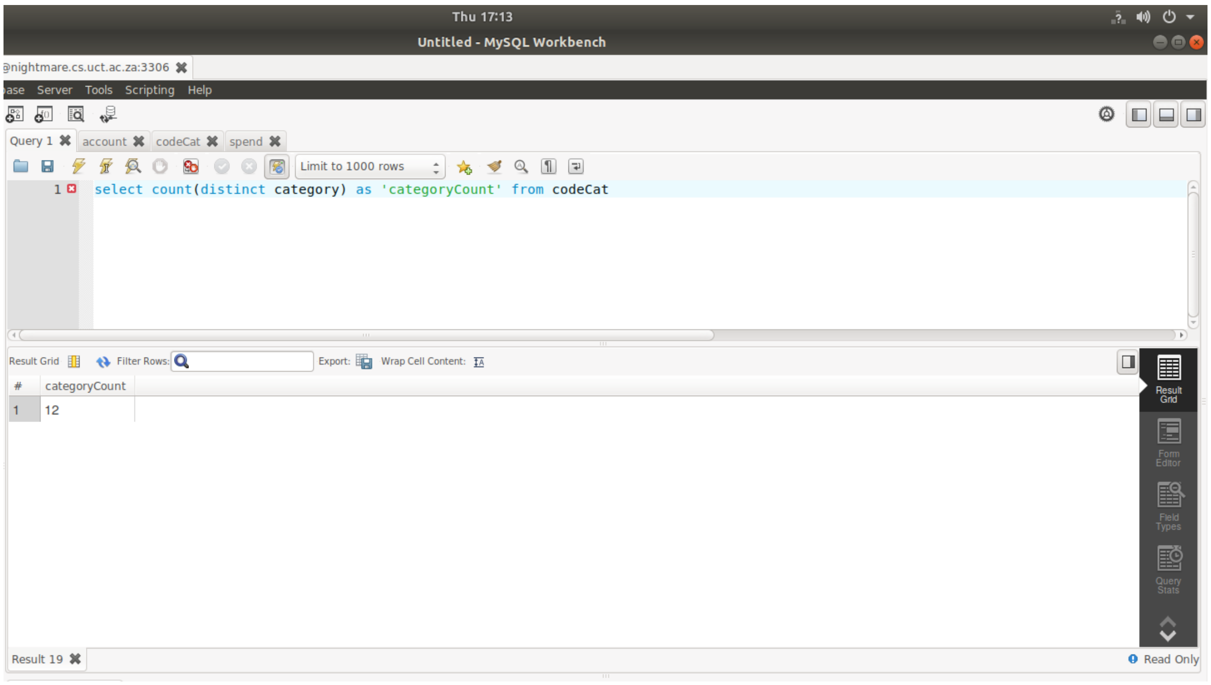


```
select count(distinct id) as 'Higher' from spend
where
abs(amount) > (select abs(amount) from spend where id='12312870316')
```

```
Higher
9
```

```
#   Time       Action                                       Message                 Duration / Fetch
3   15:59:50   SELECT * FROM prkmah005.account LIMIT 0, 1000   1000 row(s) returned   0.0018 sec / 0.0001...
```

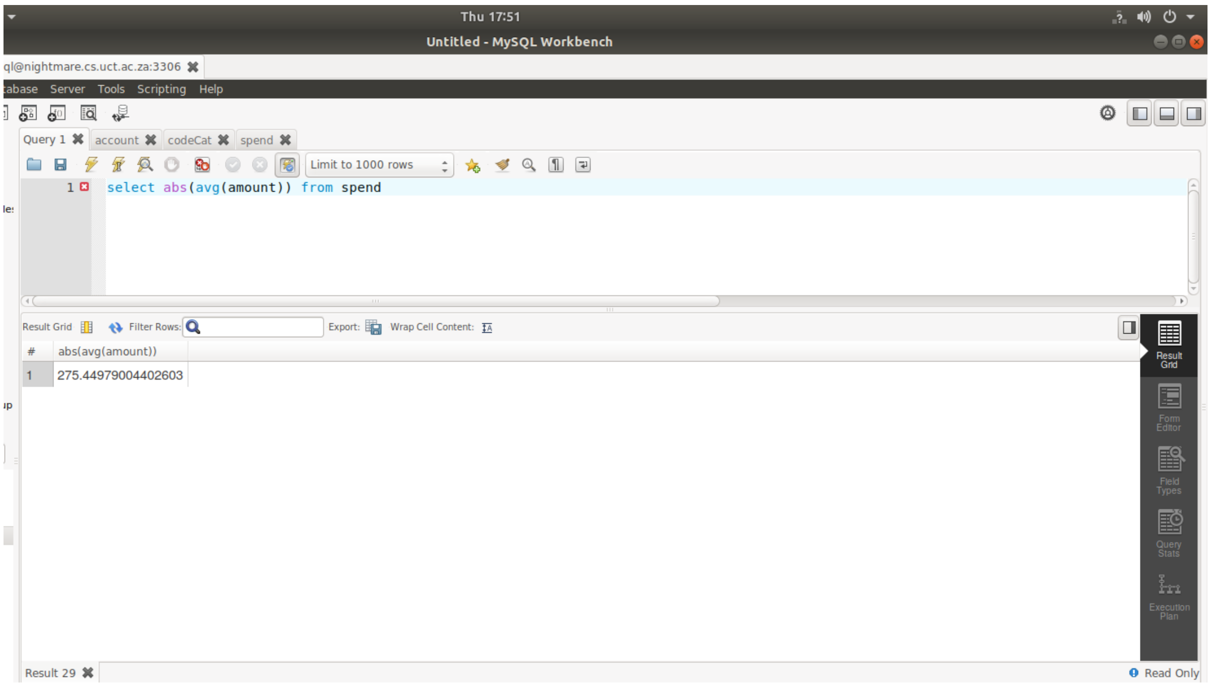
9. How many categories are there? (i.e. how many different category values?)



```
select count(distinct category) as 'categoryCount' from codeCat
```

```
#   categoryCount
1   12
```

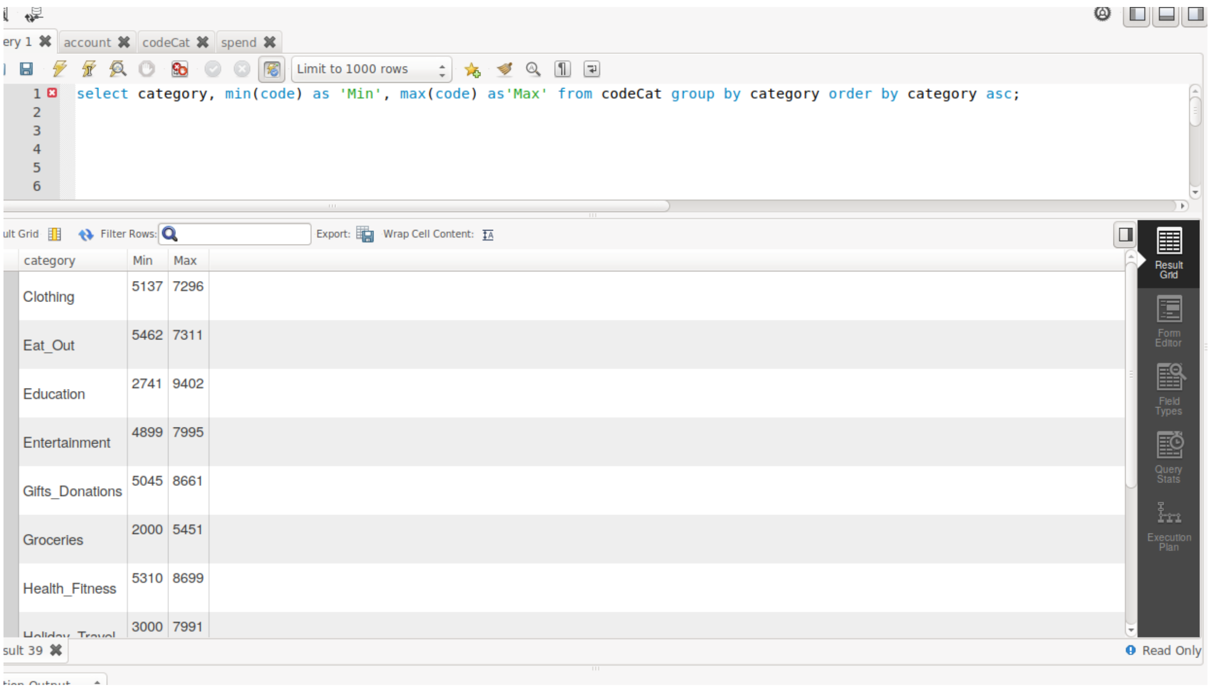
10. Give the average transaction amount (a single value in the answer).



```
select abs(avg(amount)) from spend
```

#	abs(avg(amount))
1	275.44979004402603

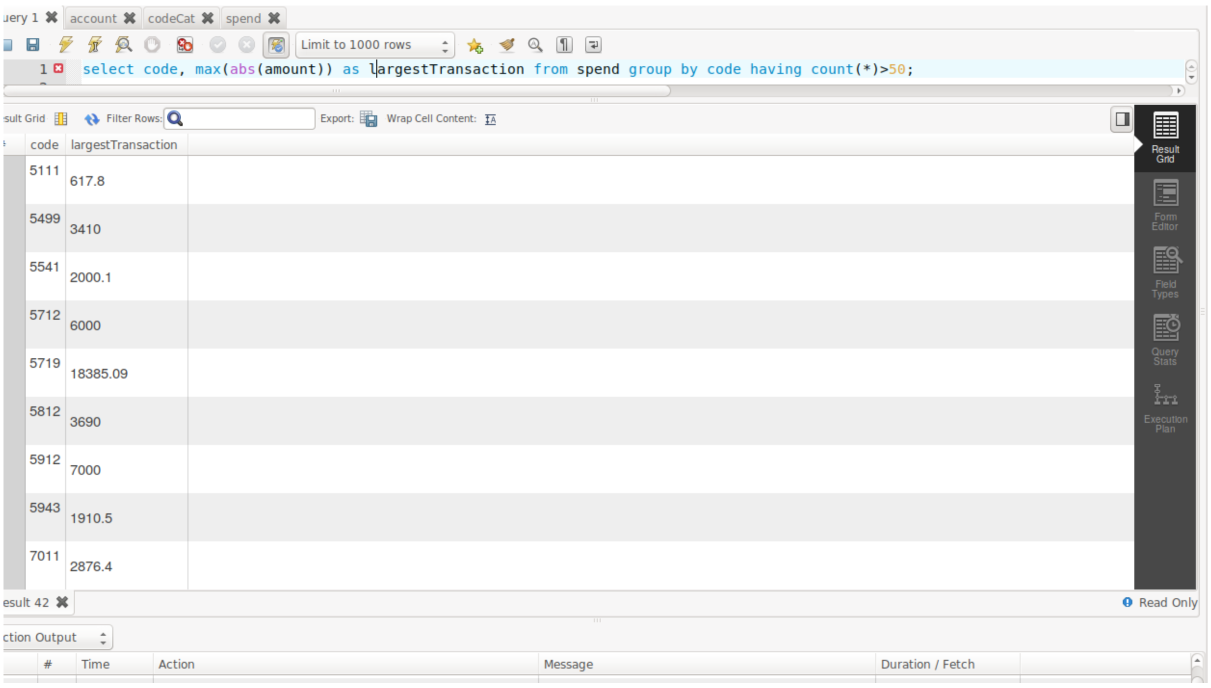
11. Find the code range for each category (i.e. each result line will have category name, lowest code and highest code for that category).



```
select category, min(code) as 'Min', max(code) as'Max' from codeCat group by category order by category asc;
```

category	Min	Max
Clothing	5137	7296
Eat_Out	5462	7311
Education	2741	9402
Entertainment	4899	7995
Gifts_Donations	5045	8661
Groceries	2000	5451
Health_Fitness	5310	8699
Holiday_Travel	3000	7991

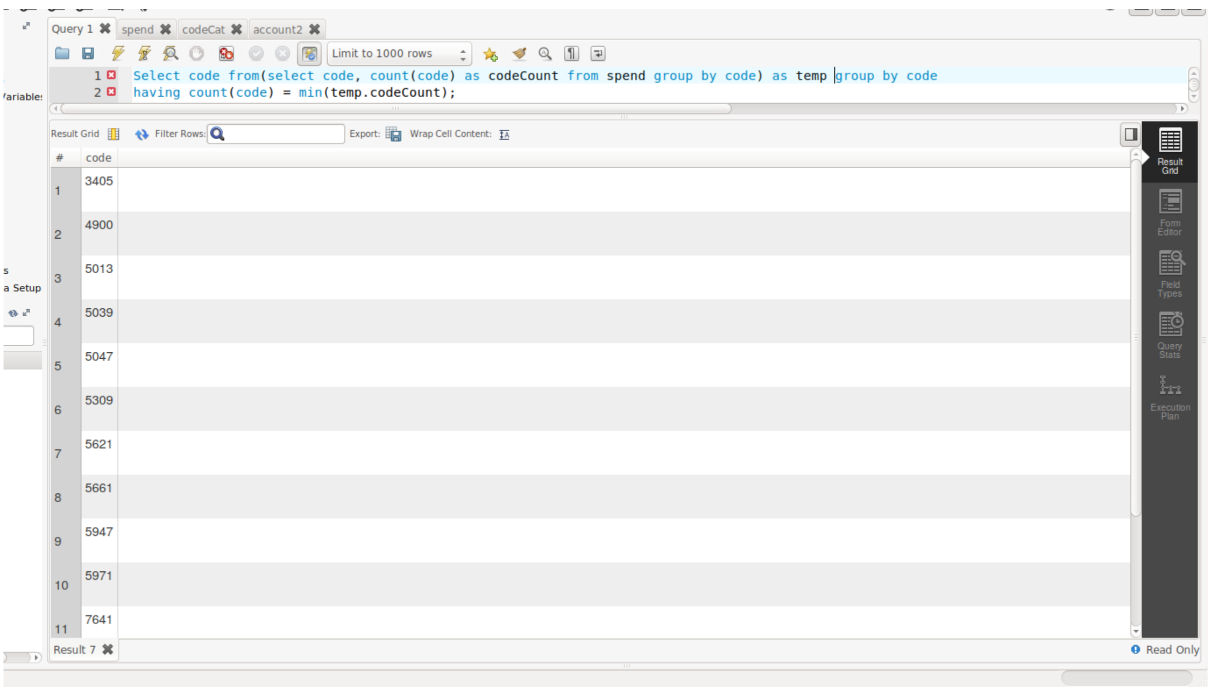
12. Find the largest transaction amount for each code that is associated with more than 50 transactions.



```
select code, max(abs(amount)) as largestTransaction from spend group by code having count(*)>50;
```

code	largestTransaction
5111	617.8
5499	3410
5541	2000.1
5712	6000
5719	18385.09
5812	3690
5912	7000
5943	1910.5
7011	2876.4

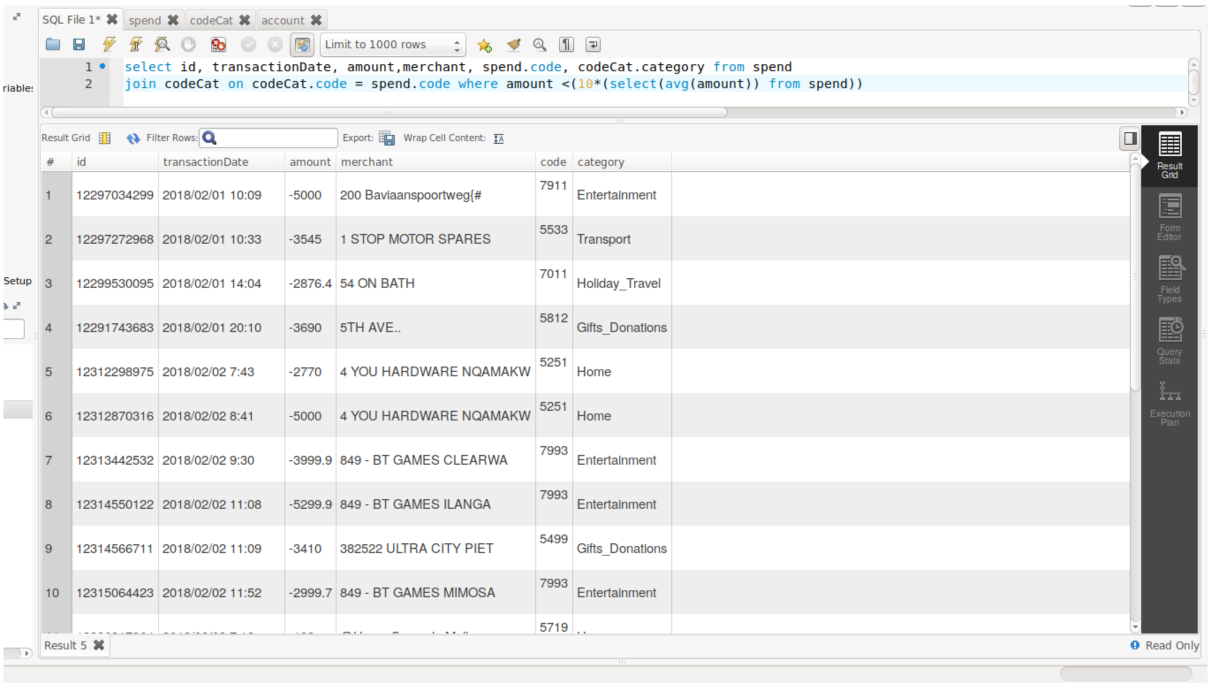
13. Which code(s) have the least number of transactions i.e. which code(s) have the fewest transactions?



```
Select code from(select code, count(code) as codeCount from spend group by code) as temp group by code
having count(code) = min(temp.codeCount);
```

#	code
1	3405
2	4900
3	5013
4	5039
5	5047
6	5309
7	5621
8	5661
9	5947
10	5971
11	7641

14. Show all information on all large transactions including the category involved. A large transaction is one with an amount that exceeds ten times the average transaction amount (e.g. if the average transaction amount is R100, a large transaction has an amount above R1000).

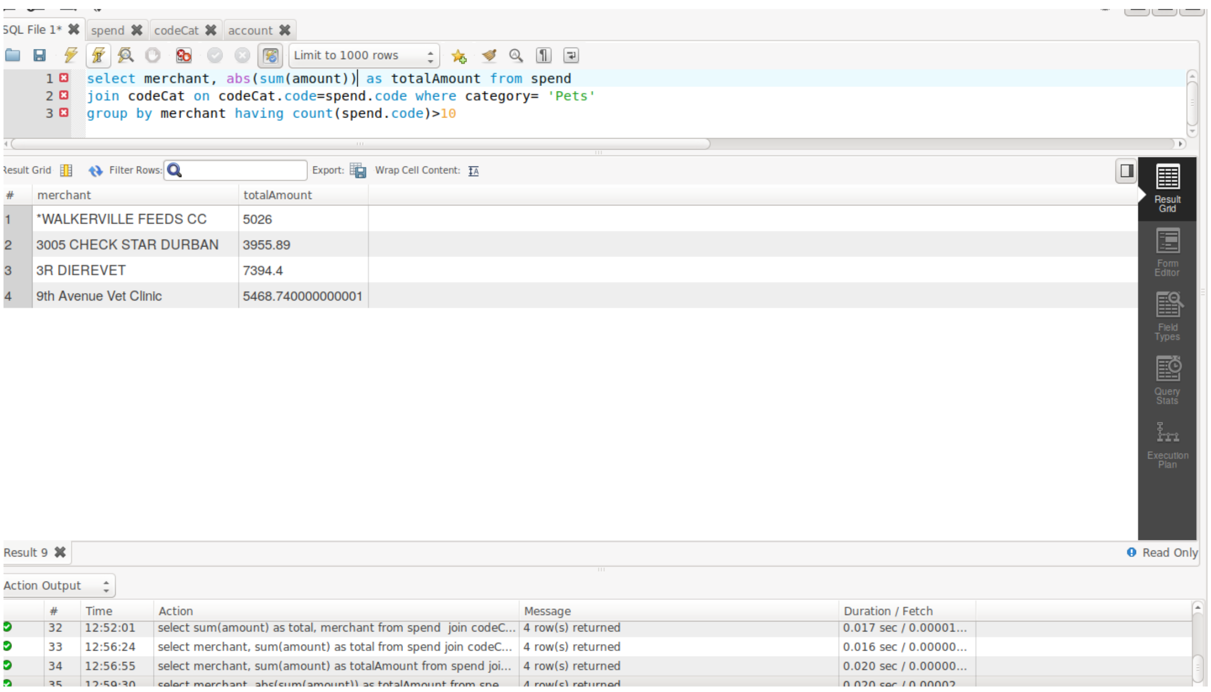


```sql
select id, transactionDate, amount,merchant, spend.code, codeCat.category from spend
join codeCat on codeCat.code = spend.code where amount <(10*(select(avg(amount)) from spend))
```

#	id	transactionDate	amount	merchant	code	category
1	12297034299	2018/02/01 10:09	-5000	200 Baviaanspoortweg{#	7911	Entertainment
2	12297272968	2018/02/01 10:33	-3545	1 STOP MOTOR SPARES	5533	Transport
3	12299530095	2018/02/01 14:04	-2876.4	54 ON BATH	7011	Holiday_Travel
4	12291743683	2018/02/01 20:10	-3690	5TH AVE..	5812	Gifts_Donations
5	12312298975	2018/02/02 7:43	-2770	4 YOU HARDWARE NQAMAKW	5251	Home
6	12312870316	2018/02/02 8:41	-5000	4 YOU HARDWARE NQAMAKW	5251	Home
7	12313442532	2018/02/02 9:30	-3999.9	849 - BT GAMES CLEARWA	7993	Entertainment
8	12314550122	2018/02/02 11:08	-5299.9	849 - BT GAMES ILANGA	7993	Entertainment
9	12314566711	2018/02/02 11:09	-3410	382522 ULTRA CITY PIET	5499	Gifts_Donations
10	12315064423	2018/02/02 11:52	-2999.7	849 - BT GAMES MIMOSA	7993	Entertainment
					5719	

Result 5 ✖ Read Only

15. Find the total amount spent on Pets for each merchant where there have been more than 10 transactions in the Pets category.



```sql
select merchant, abs(sum(amount)) as totalAmount from spend
join codeCat on codeCat.code=spend.code where category= 'Pets'
group by merchant having count(spend.code)>10
```

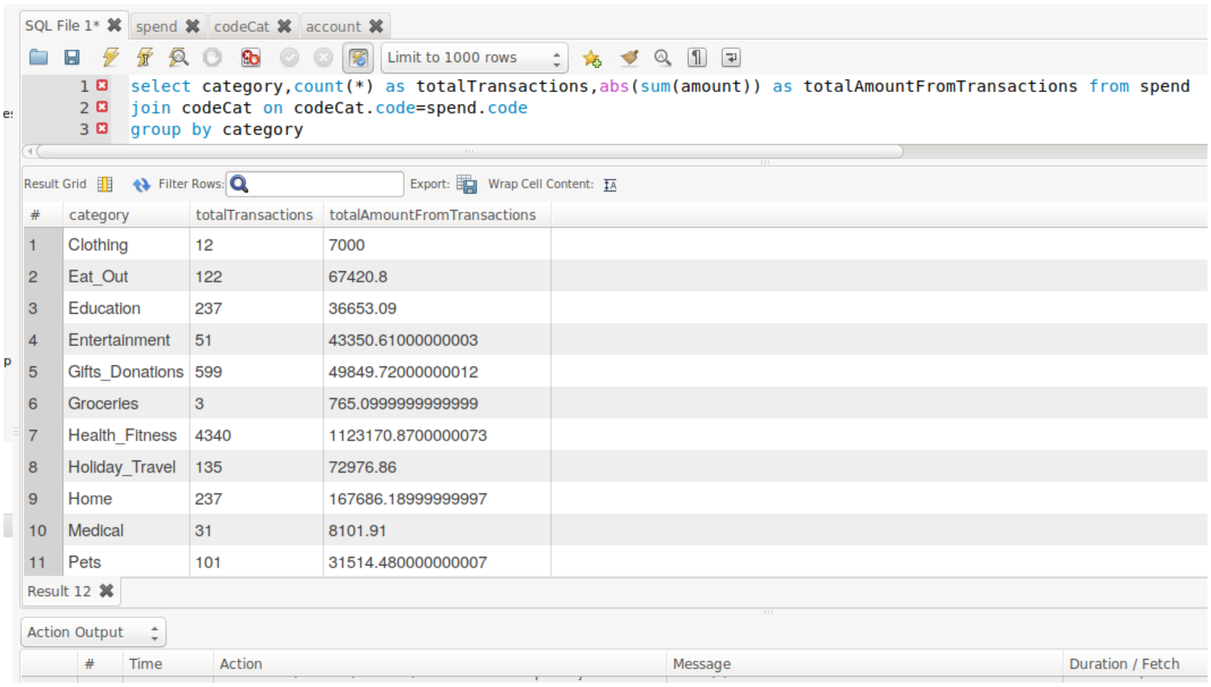
#	merchant	totalAmount
1	*WALKERVILLE FEEDS CC	5026
2	3005 CHECK STAR DURBAN	3955.89
3	3R DIEREVET	7394.4
4	9th Avenue Vet Clinic	5468.740000000001

Result 9 ✖ Read Only

Action Output

#	Time	Action	Message	Duration / Fetch
32	12:52:01	select sum(amount) as total, merchant from spend join codeC...	4 row(s) returned	0.017 sec / 0.00001...
33	12:56:24	select merchant, sum(amount) as total from spend join codeC...	4 row(s) returned	0.016 sec / 0.00000...
34	12:56:55	select merchant, sum(amount) as totalAmount from spend joi...	4 row(s) returned	0.020 sec / 0.00000...
35	12:59:30	select merchant, abs(sum(amount)) as totalAmount from spe...	4 row(s) returned	0.020 sec / 0.00002

16. For each category, show how many such transactions there are in the database along with the total Rand amount of those transactions.

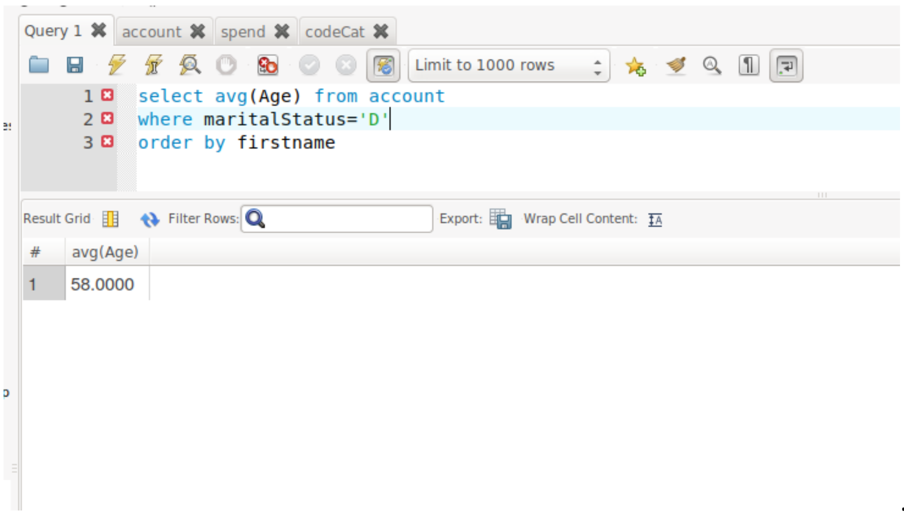

SQL File 1* | spend | codeCat | account
Limit to 1000 rows
Result Grid | Filter Rows: | Export: | Wrap Cell Content:
Result 12
Action Output
# | Time | Action | Message | Duration / Fetch

```sql
select category,count(*) as totalTransactions,abs(sum(amount)) as totalAmountFromTransactions from spend
join codeCat on codeCat.code=spend.code
group by category
```

#	category	totalTransactions	totalAmountFromTransactions
1	Clothing	12	7000
2	Eat_Out	122	67420.8
3	Education	237	36653.09
4	Entertainment	51	43350.61000000003
5	Gifts_Donations	599	49849.72000000012
6	Groceries	3	765.0999999999999
7	Health_Fitness	4340	1123170.8700000073
8	Holiday_Travel	135	72976.86
9	Home	237	167686.18999999997
10	Medical	31	8101.91
11	Pets	101	31514.480000000007

Section C

Problem: devise a query to select the average age of people who are divorced


Query 1 | account | spend | codeCat
Limit to 1000 rows
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

```sql
select avg(Age) from account
where maritalStatus='D'
order by firstname
```

#	avg(Age)
1	58.0000

My code definitely works. I made the age in my account table an INTEGER type. And the average age of the general table is different.

Section D

i. delete* from spend where merchant like ‘%3%@%1%’;

ii. insert into spend VALUES(‘12393560590’, ‘2018/12/31 0:00’, ‘-448’, ’’, ‘5251’);

iii. update spend set code = 5221 where code=5211;

update codeCat set code = 5221 where code=5211;

Iv. update account set maritalStatus=’U’ where maritalStatus != ‘M’;