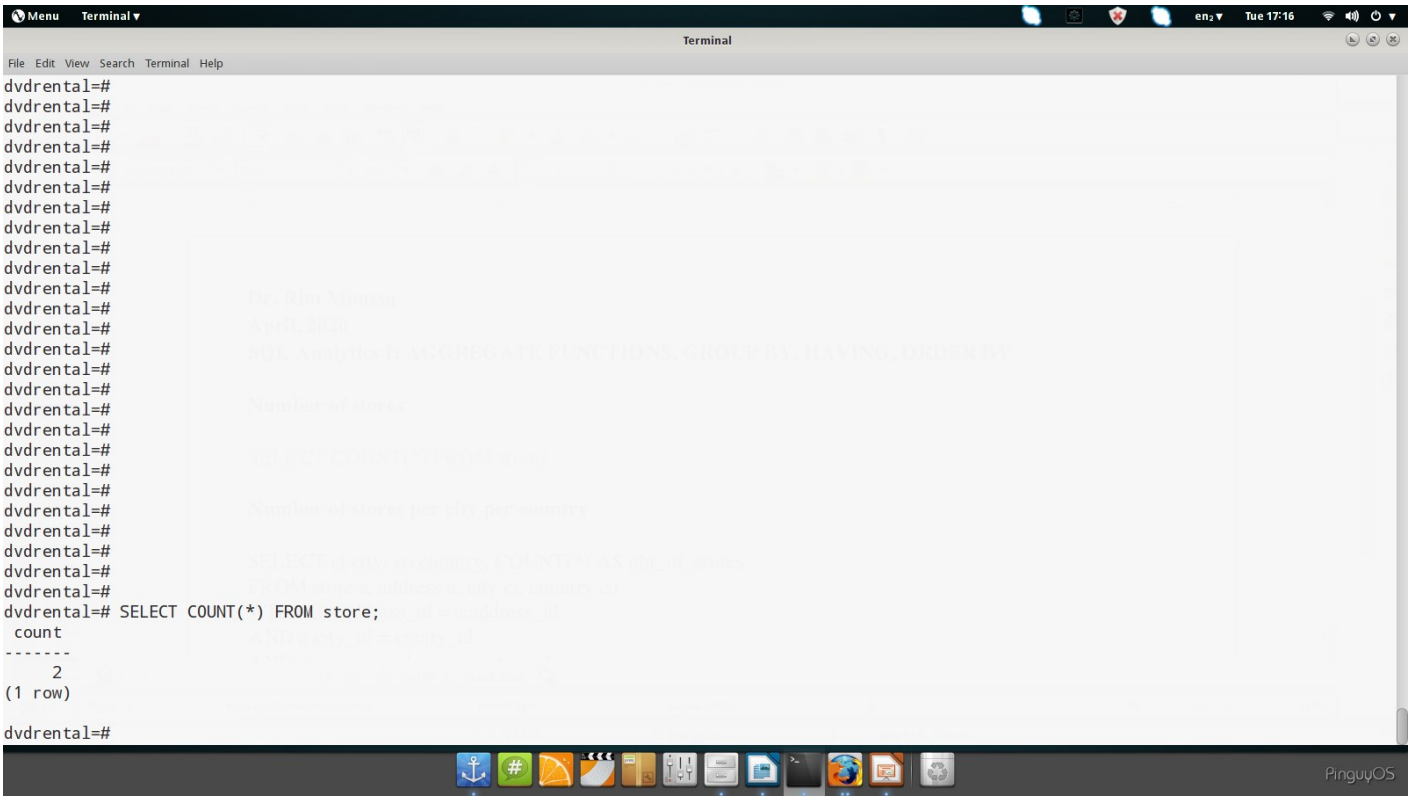
Dr. Rim Moussa
April, 2020
SQL Analytics I

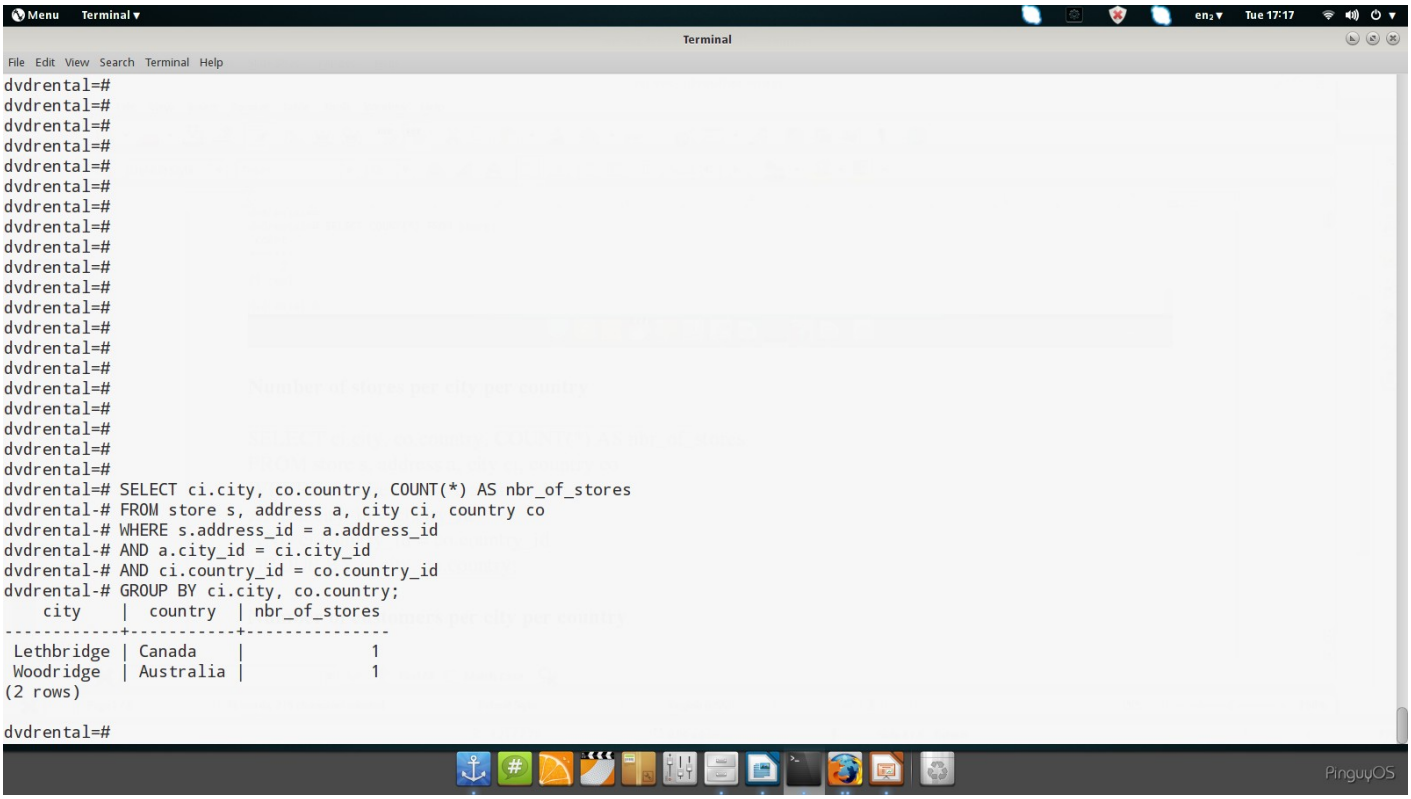
Q1: Number of stores

SELECT COUNT(*) FROM store;



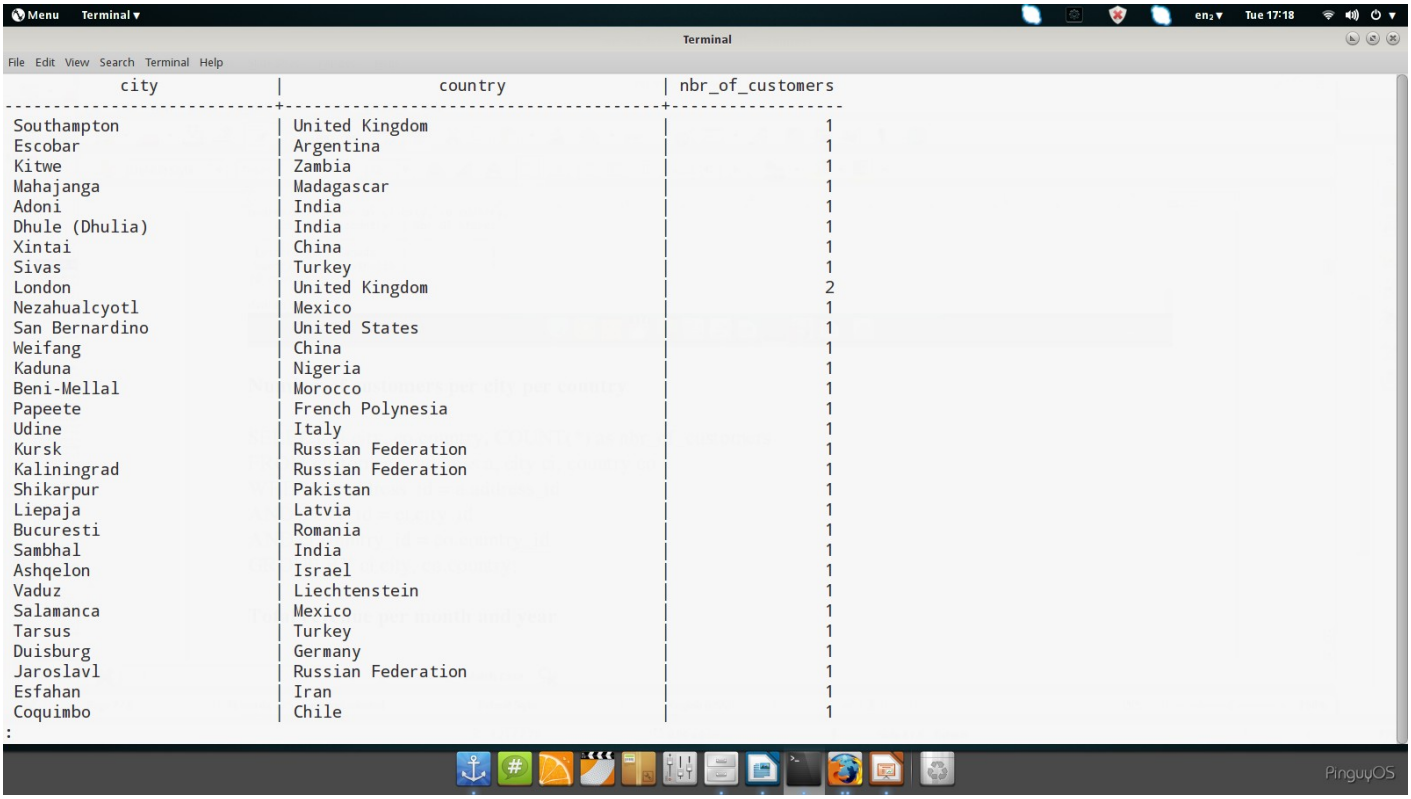
Q2: Number of stores per city per country

SELECT ci.city, co.country, COUNT(*) AS nbr_of_stores
FROM store s, address a, city ci, country co
WHERE s.address_id = a.address_id
AND a.city_id = ci.city_id
AND ci.country_id = co.country_id
GROUP BY ci.city, co.country;



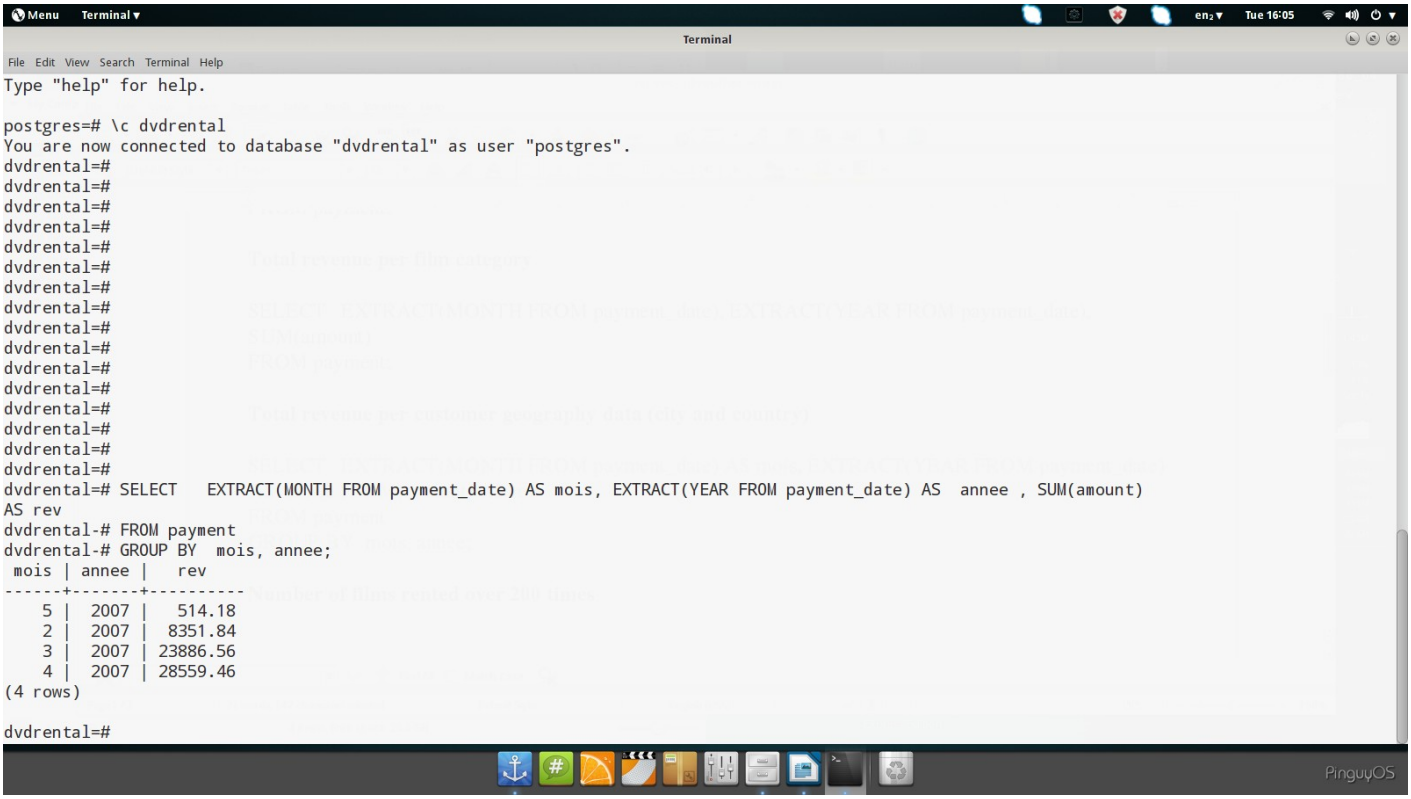
Q3: Number of customers per city per country

SELECT ci.city, co.country, COUNT(*) as nbr_of_customers
FROM customer c, address a, city ci, country co
WHERE c.address_id = a.address_id
AND a.city_id = ci.city_id
AND ci.country_id = co.country_id
GROUP BY ci.city, co.country;



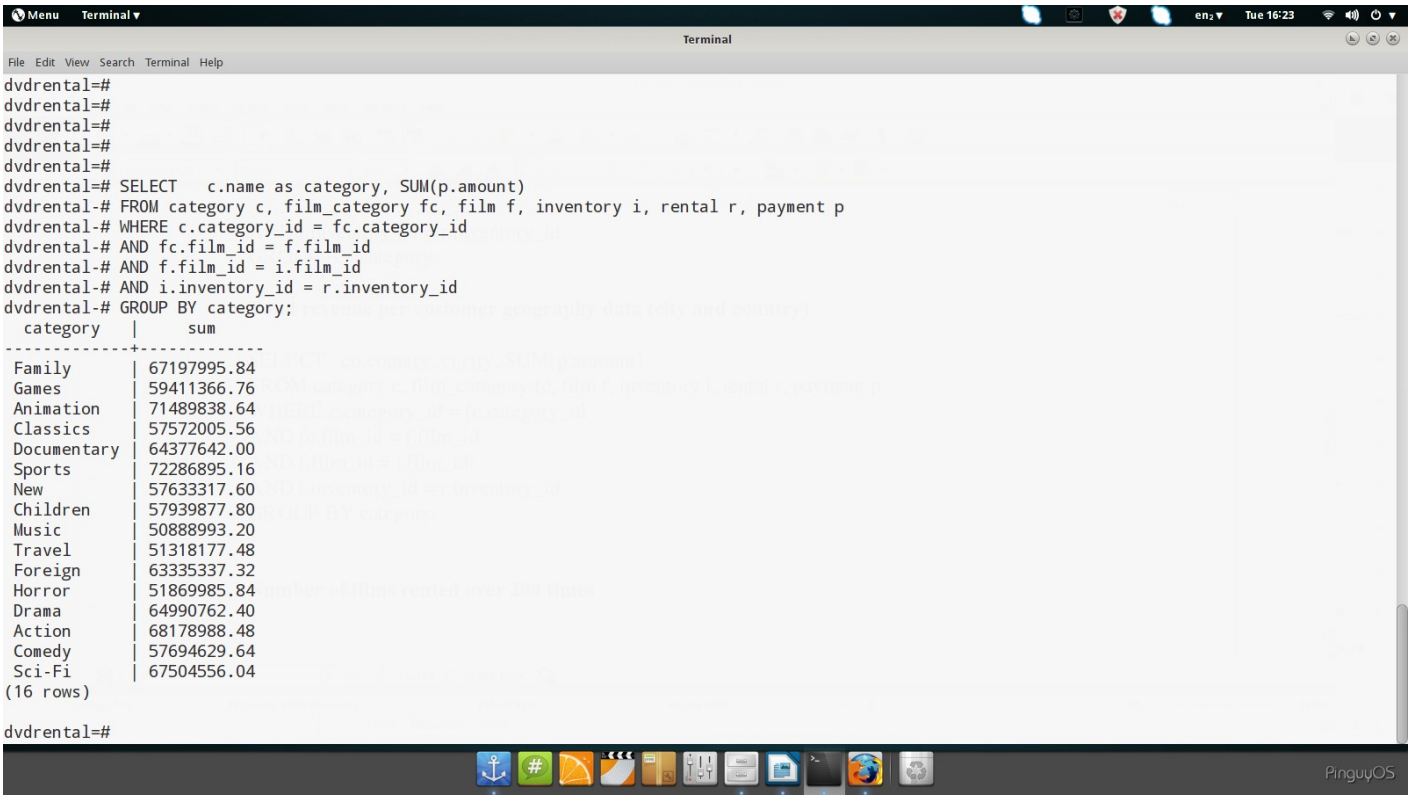
Q4: Total revenue per month and year

```
SELECT   EXTRACT(MONTH FROM payment_date) AS mois, EXTRACT(YEAR FROM payment_date)
AS  annee , SUM(amount) AS rev
FROM payment
GROUP BY  mois, annee;
```



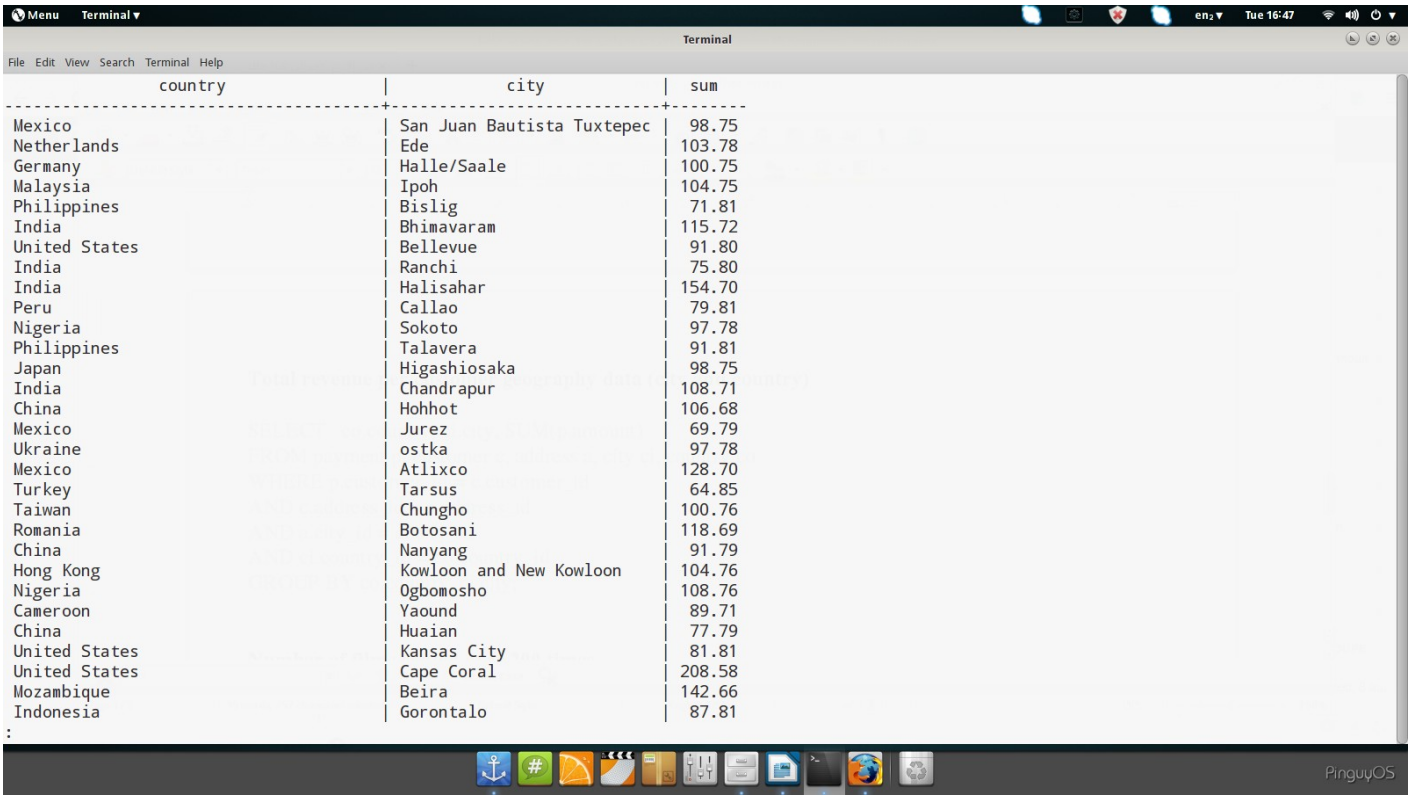
Q5: Total revenue per film category

```
SELECT   c.name as category, SUM(p.amount)
FROM category c, film_category fc, film f, inventory i, rental r, payment p
WHERE c.category_id = fc.category_id
AND fc.film_id = f.film_id
AND f.film_id = i.film_id
AND i.inventory_id = r.inventory_id
GROUP BY category;
```



Q6: Total revenue per customer geography data (city and country)

SELECT co.country, ci.city, SUM(p.amount)
FROM payment p, customer c, address a, city ci, country co
WHERE p.customer_id = c.customer_id
AND c.address_id = a.address_id
AND a.city_id = ci.city_id
AND ci.country_id = co.country_id
GROUP BY co.country, ci.city;

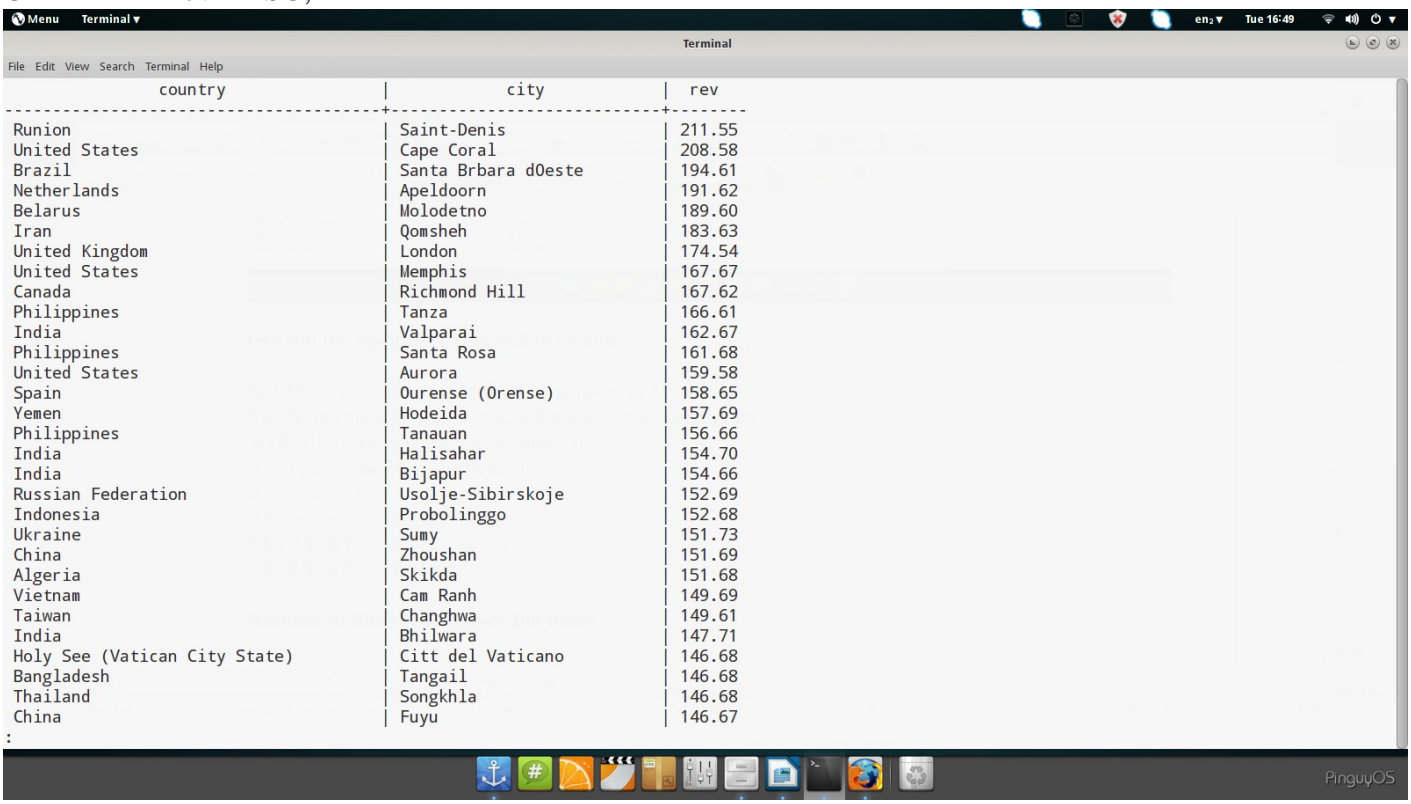


country	city	sum
Mexico	San Juan Bautista Tuxtepec	98.75
Netherlands	Ede	103.78
Germany	Halle/Saale	100.75
Malaysia	Ipoh	104.75
Philippines	Bislig	71.81
India	Bhimavaram	115.72
United States	Bellevue	91.80
India	Ranchi	75.80
India	Halisahar	154.70
Peru	Callao	79.81
Nigeria	Sokoto	97.78
Philippines	Talavera	91.81
Japan	Higashiosaka	98.75
India	Chandrapur	108.71
China	Hohhot	106.68
Mexico	Jurez	69.79
Ukraine	ostka	97.78
Mexico	Atlixco	128.70
Turkey	Tarsus	64.85
Taiwan	Chungho	100.76
Romania	Botosani	118.69
China	Nanyang	91.79
Hong Kong	Kowloon and New Kowloon	104.76
Nigeria	Ogbomosho	108.76
Cameroon	Yaound	89.71
China	Huaian	77.79
United States	Kansas City	81.81
United States	Cape Coral	208.58
Mozambique	Beira	142.66
Indonesia	Gorontalo	87.81

:

let's sort the resultset on descending revenue

```
SELECT   co.country, ci.city, SUM(p.amount) AS rev
FROM payment p, customer c, address a, city ci, country co
WHERE p.customer_id = c.customer_id
AND c.address_id = a.address_id
AND a.city_id = ci.city_id
AND ci.country_id = co.country_id
GROUP BY co.country, ci.city
ORDER BY rev DESC;
```

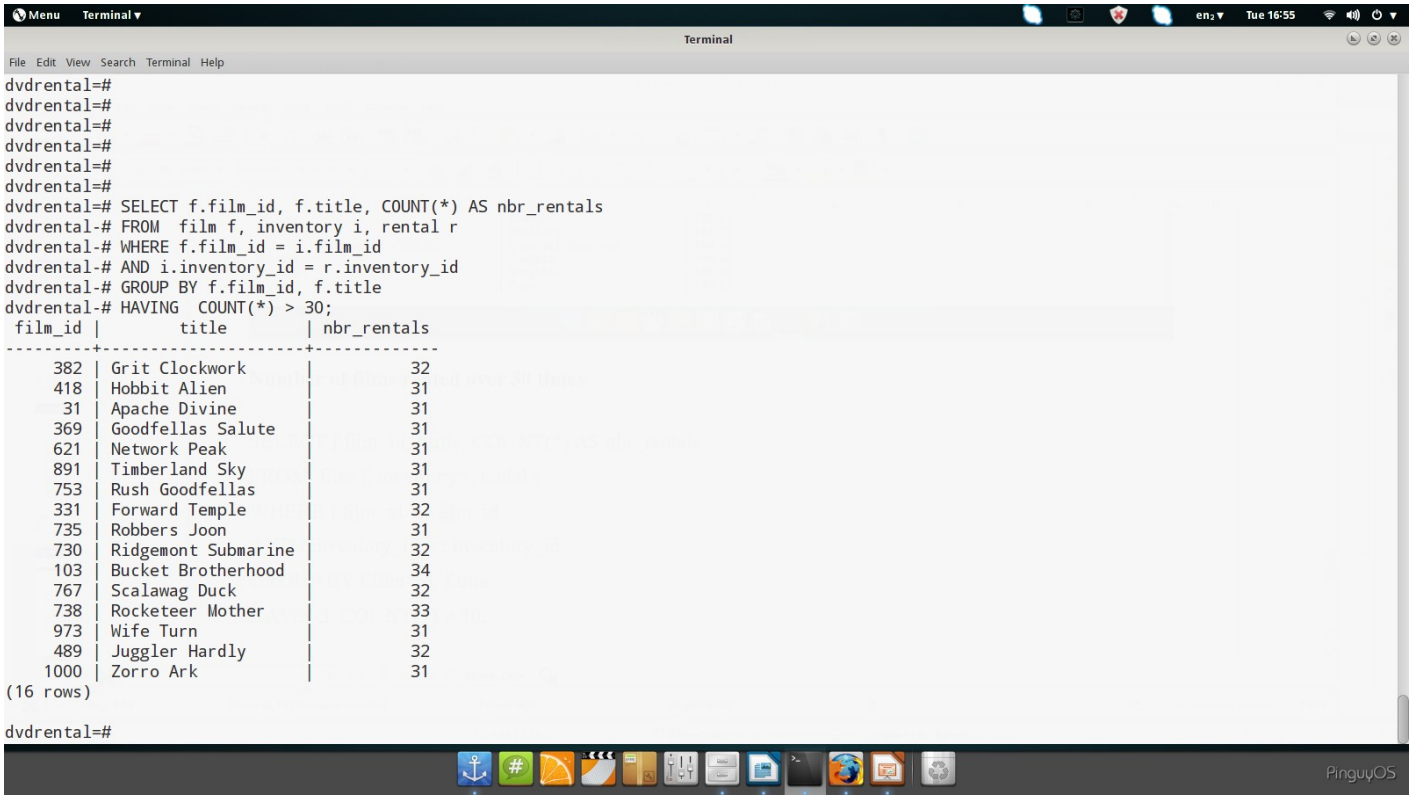


country	city	rev
Runion	Saint-Denis	211.55
United States	Cape Coral	208.58
Brazil	Santa Brbara dOeste	194.61
Netherlands	Apeldoorn	191.62
Belarus	Molodetno	189.60
Iran	Qomsheh	183.63
United Kingdom	London	174.54
United States	Memphis	167.67
Canada	Richmond Hill	167.62
Philippines	Tanza	166.61
India	Valparai	162.67
Philippines	Santa Rosa	161.68
United States	Aurora	159.58
Spain	Ourense (Orense)	158.65
Yemen	Hodeida	157.69
Philippines	Tanauan	156.66
India	Halisahar	154.70
India	Bijapur	154.66
Russian Federation	Usolje-Sibirskoje	152.69
Indonesia	Probolinggo	152.68
Ukraine	Sumy	151.73
China	Zhoushan	151.69
Algeria	Skikda	151.68
Vietnam	Cam Ranh	149.69
Taiwan	Changhwa	149.61
India	Bhilwara	147.71
Holy See (Vatican City State)	Citt del Vaticano	146.68
Bangladesh	Tangail	146.68
Thailand	Songkhla	146.68
China	Fuyu	146.67

:

Q7: Number of films rented over 30 times

```
SELECT f.film_id, f.title, COUNT(*) AS nbr_rentals
FROM  film f, inventory i, rental r
WHERE f.film_id = i.film_id
AND i.inventory_id = r.inventory_id
GROUP BY f.film_id, f.title
HAVING  COUNT(*) > 30;
```

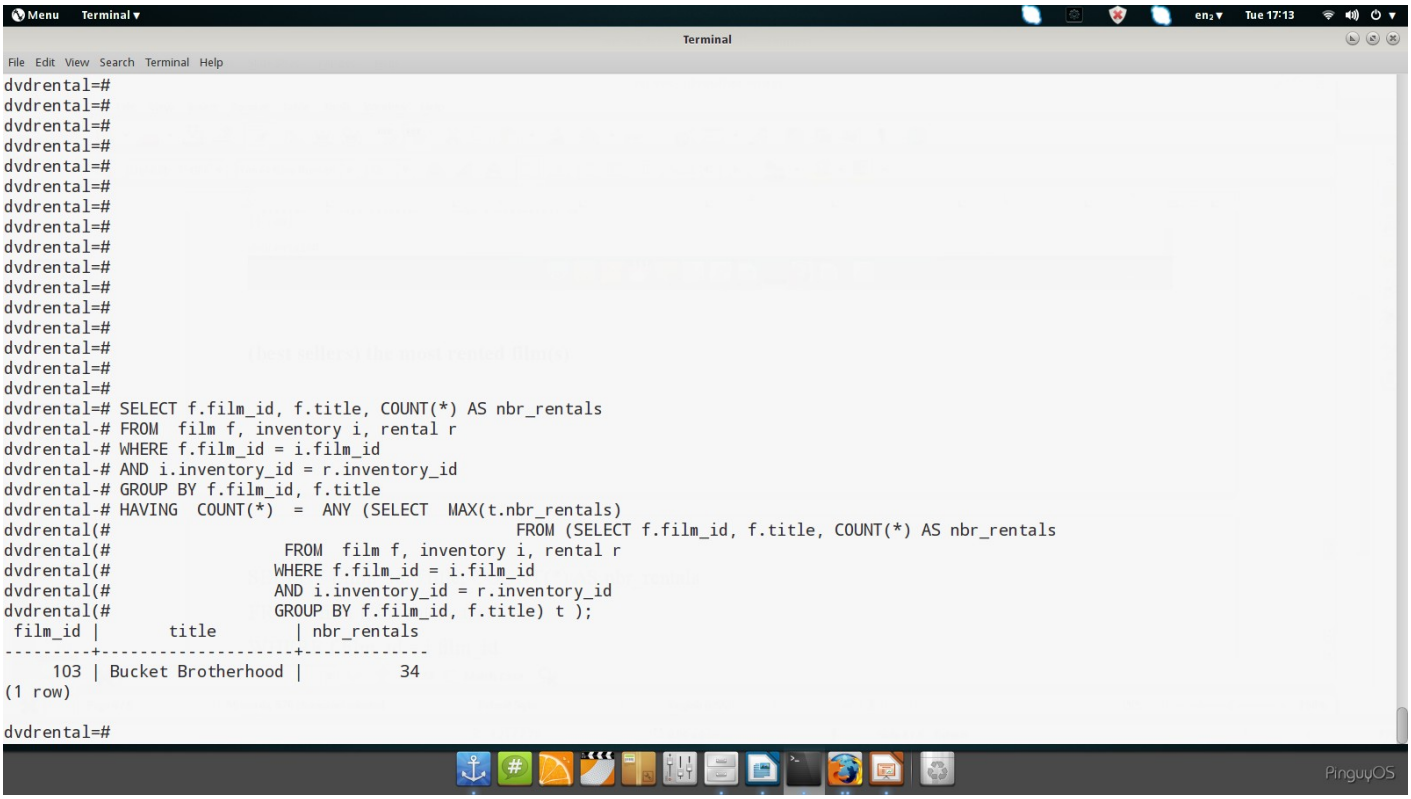


Q8: stat rent duration (number of days) per film

```
SELECT MIN(return_date - rental_date) , MAX(return_date - rental_date), AVG(return_date - rental_date)
FROM rental;
```

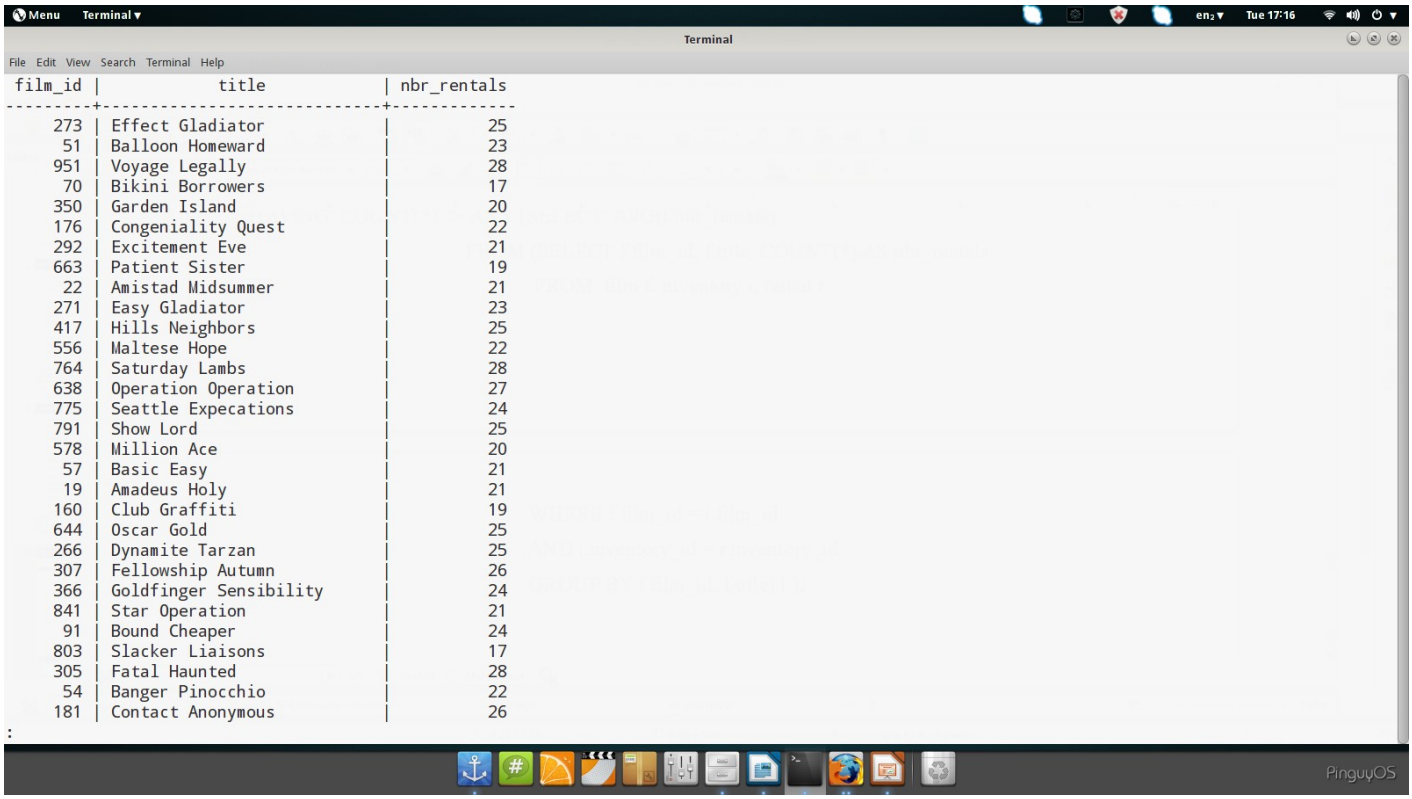
Q9: (best sellers) the most rented film(s)

```
SELECT f.film_id, f.title, COUNT(*) AS nbr_rentals
FROM  film f, inventory i, rental r
WHERE f.film_id = i.film_id
AND i.inventory_id = r.inventory_id
GROUP BY f.film_id, f.title
HAVING  COUNT(*)  =  ANY (SELECT  MAX(t.nbr_rentals)
                          FROM (SELECT f.film_id, f.title, COUNT(*) AS nbr_rentals
                                 FROM  film f, inventory i, rental r
                                WHERE f.film_id = i.film_id
                                AND i.inventory_id = r.inventory_id
                                GROUP BY f.film_id, f.title) t );
```



Q10: films rented over the average number of rentals

```
SELECT f.film_id, f.title, COUNT(*) AS nbr_rentals
FROM  film f, inventory i, rental r
WHERE f.film_id = i.film_id
AND i.inventory_id = r.inventory_id
GROUP BY f.film_id, f.title
HAVING  COUNT(*)  >  ANY (SELECT  AVG(t.nbr_rentals)
                          FROM (SELECT f.film_id, f.title, COUNT(*) AS nbr_rentals
                                FROM  film f, inventory i, rental r
                               WHERE f.film_id = i.film_id
                               AND i.inventory_id = r.inventory_id
                               GROUP BY f.film_id, f.title) t );
```



Q11: Who are the actors who played in all film categories

Hint: count the number of distinct categories of films that each actor played in, then compare to the total number of categories

SELECT a.actor_id, a.last_name, a.first_name, COUNT(DISTINCT fc.category_id)

FROM actor a, film_actor fa, film f, film_category fc

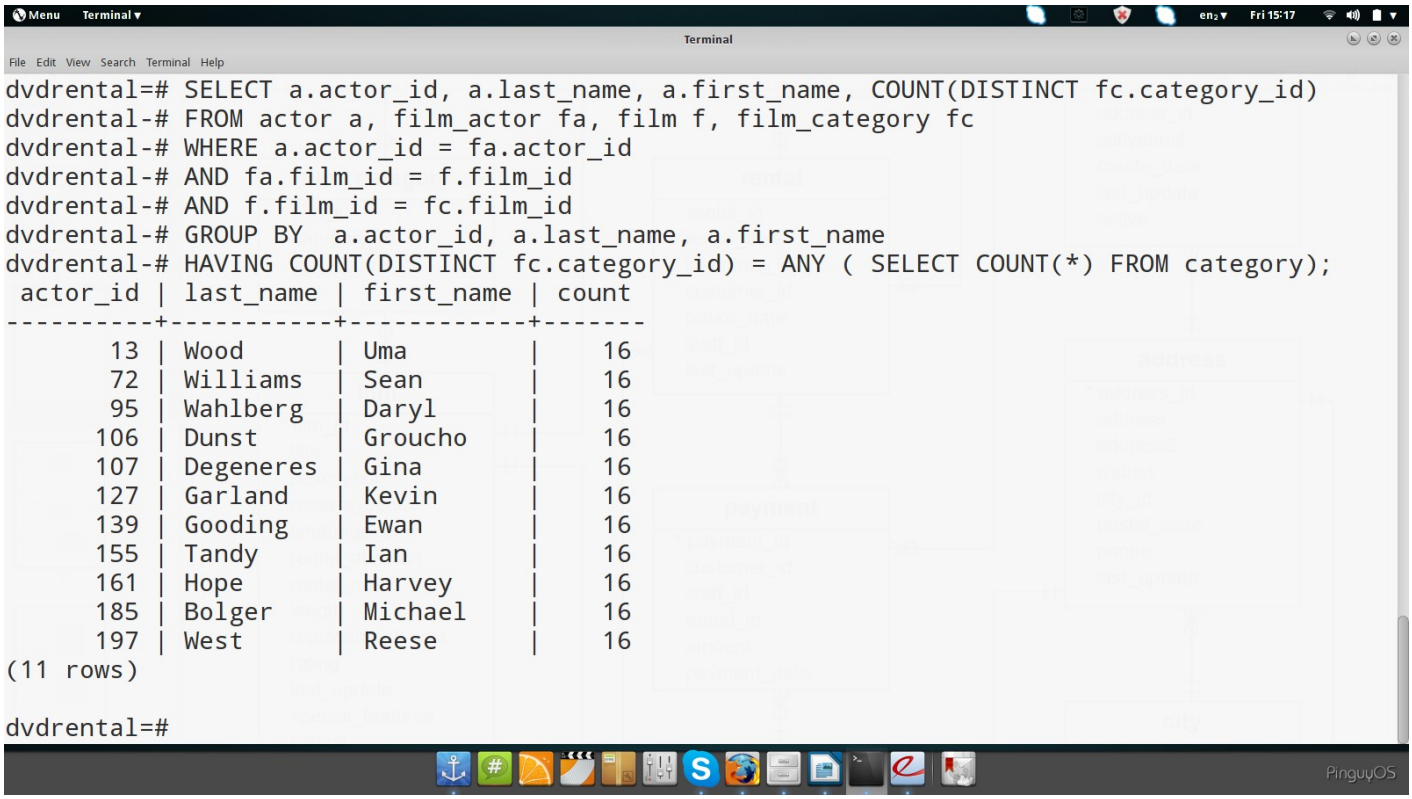
WHERE a.actor_id = fa.actor_id

AND fa.film_id = f.film_id

AND f.film_id = fc.film_id

GROUP BY a.actor_id, a.last_name, a.first_name

HAVING COUNT(DISTINCT fc.category_id) = ANY (SELECT COUNT(*) FROM category);



notice that we can replace fa.film_id = f.film_id AND f.film_id = fc.film_id by fa.film_id = fc.film_id, and we don't need to use film table.