

Actividad 17

- Liste las ciudades(id,nombre) y las tiendas que se encuentran en cada ciudad, use RIGHT JOIN para mostrar TODAS las ciudades haya o no tiendas en ellas.

```
SELECT ci.city_id,ci.city, a.address, st.store_id
```

```
FROM store st
```

```
RIGHT JOIN address a on st.address_id=a.address_id
```

```
RIGHT JOIN city ci on a.city_id=ci.city_id
```

```
1 • SELECT ci.city_id,ci.city, a.address, st.store_id
2 FROM store st
3 RIGHT JOIN address a on st.address_id=a.address_id
4 RIGHT JOIN city ci on a.city_id=ci.city_id
```

city_id	city	address	store_id
292	Lancaster	793 Cam Ranh Avenue	NULL
293	Laohekou	1557 Cape Coral Parkway	NULL
294	Lapu-Lapu	582 Papeete Loop	NULL
295	Laredo	1795 Santiago de Compo...	NULL
296	Lausanne	1331 Usak Boulevard	NULL
297	Le Mans	1740 Le Mans Loop	NULL
298	Lengshuijiang	1762 Paarl Parkway	NULL
299	Leshan	1052 Pathankot Avenue	NULL
300	Lethbridge	47 MySakla Drive	1
300	Lethbridge	23 Workhaven Lane	NULL
301	Lhokseumawe	1078 Stara Zagora Drive	NULL
302	Liaocheng	1854 Tiel Street	NULL
303	Liepaja	1780 Hino Boulevard	NULL
304	Lilongwe	710 San Felipe del Progre...	NULL
305	Lima	1175 Tanawon Wlaxu	NULL

- Liste los países (id,nombre) y las ciudades que se encuentran en cada país, use RIGHT JOIN para mostrar TODOS los países haya o no ciudades en ellos

```
SELECT co.country_id,co.country,ci.city
```

```
FROM city ci
```

```
RIGHT JOIN country co on ci.country_id=co.country_id
```

```
1 • SELECT co.country_id,co.country,ci.city
2 FROM city ci
3 RIGHT JOIN country co on ci.country_id=co.country_id
```

country_id	country	city
105	Vietnam	Halong
105	Vietnam	Hanoi
105	Vietnam	Nam Dinh
105	Vietnam	Nha Trang
105	Vietnam	Vinh
106	Virgin Islands, U.S.	Charlotte Amalie
107	Yemen	Aden
107	Yemen	Hodeida
107	Yemen	Sanaa
107	Yemen	Taizz
108	Yugoslavia	Kragujevac
108	Yugoslavia	Novi Sad
109	Zambia	Kabwe
110	Marshall Islands	Koror

- Liste las ciudades (id y nombre) y los clientes que se encuentran en cada ciudad, use RIGHT JOIN para mostrar TODAS las ciudades haya o no clientes en ellas.

```
SELECT ci.city_id,ci.city,c.customer_id
```

```
FROM customer c
```

```
RIGHT JOIN address a on c.address_id=a.address_id
```

```
RIGHT JOIN city ci on a.city_id=ci.city_id
```

```
1 • SELECT ci.city_id,ci.city,c.customer_id
2 FROM customer c
3 RIGHT JOIN address a on c.address_id=a.address_id
4 RIGHT JOIN city ci on a.city_id=ci.city_id
```

city_id	city	customer_id
308	Lipetsk	44
309	Livorno	541
310	Ljubertsy	160
311	Loja	589
312	London	252
312	London	512
313	London	653
314	Lublin	501
315	Lubumbashi	387
316	Lungtan	30
317	Luzhnia	183
318	Madiun	500
319	Mahajanga	81
320	Malikop	346
321	Malmö	66

- Listar las películas que no se han rentado ni una vez

Solución 1:

```
SELECT i.store_id,r.rental_id,i.inventory_id,f.film_id,f.title,sum(r.rental_id) as
suma
FROM rental r
RIGHT JOIN inventory i on r.inventory_id=i.inventory_id
RIGHT JOIN film f on i.film_id=f.film_id
WHERE i.store_id=1
GROUP BY f.film_id
ORDER BY film_id
```

The screenshot shows a SQL query editor with the following query:

```
1 • SELECT i.store_id,r.rental_id,i.inventory_id,f.film_id,f.title,sum(r.rental_id) as suma
2 FROM rental r
3 RIGHT JOIN inventory i on r.inventory_id=i.inventory_id
4 RIGHT JOIN film f on i.film_id=f.film_id
5 WHERE i.store_id=1
6 GROUP BY f.film_id
7 ORDER BY film_id
```

Below the query, the 'Result Grid' shows the results of the query. The columns are store_id, rental_id, inventory_id, film_id, title, and suma. The results are as follows:

store_id	rental_id	inventory_id	film_id	title	suma
1		1	1	ACADEMY DINOSAUR	972
1		16	4	AFFAIR PREJUDICE	995
1	317	26	6	AGENT TRUMAN	317
1		32	7	AIRPLANE SIERRA	
1		41	9	ALABAMA DEVIL	
1	1948	46	10	ALADDIN CALENDAR	1948
1	563	53	11	ALAMO VIDEOTAPE	563
1		60	12	ALASKA PHANTOM	3821
1	505	71	15	ALIEN CENTER	986
1		77	16	ALLEY EVOLUTION	
1		81	17	ALONE TRIP	
1	664	87	18	ALTER VICTORY	3430
1		93	19	AMADEUS HOLY	588
1		99	20	AMELIE HELFIGHTERS	
1	2017	107	21	AMERICAN CROCODILE	1704

Solución 2:

```
SELECT i.store_id,f.film_id,f.title, coalesce(count(r.rental_id),0) as suma
FROM inventory i
RIGHT JOIN film f on i.film_id=f.film_id
LEFT JOIN rental r on i.inventory_id=r.inventory_id
WHERE i.store_id=1
GROUP BY i.store_id,f.film_id,f.title
HAVING suma=0
```

The screenshot shows a SQL query editor with the following query:

```
1 • SELECT i.store_id,f.film_id,f.title, coalesce(count(r.rental_id),0) as suma
2 FROM inventory i
3 RIGHT JOIN film f on i.film_id=f.film_id
4 LEFT JOIN rental r on i.inventory_id=r.inventory_id
5 WHERE i.store_id=1
6 GROUP BY i.store_id,f.film_id,f.title
7 HAVING suma=0
```

Below the query, the 'Result Grid' shows the results of the query. The columns are store_id, film_id, title, and suma. The results are as follows:

store_id	film_id	title	suma
1	7	AIRPLANE SIERRA	0
1	9	ALABAMA DEVIL	0
1	16	ALLEY EVOLUTION	0
1	17	ALONE TRIP	0
1	20	AMELIE HELFIGHTERS	0
1	44	ATTACKS HATE	0
1	45	ATTRACTION NEWTON	0
1	49	BACKMAN DAWN	0
1	50	BAKED CLEOPATRA	0
1	66	BENEATH RUSH	0
1	67	BETTER AGENT	0
1	69	BEVERLY OUTLAW	0
1	70	BIG IN BORROWERS	0
1	74	BIRCH ANTITRUST	0
1	76	BIRDY CAMP	0

Solución 3:

```
SELECT i.store_id,f.film_id,f.title, count(r.rental_id) as rentados
FROM inventory i
RIGHT JOIN film f on i.film_id=f.film_id
LEFT JOIN rental r on i.inventory_id=r.inventory_id
WHERE i.store_id=1
GROUP BY f.film_id
```

```
1 • SELECT i.store_id,f.film_id,f.title, count(r.rental_id) as rentados
2 FROM inventory i
3 RIGHT JOIN film f on i.film_id=f.film_id
4 LEFT JOIN rental r on i.inventory_id=r.inventory_id
5 WHERE i.store_id=1
6 GROUP BY f.film_id
```

Result Grid				
Filter Rows:				
Export:				
Wrap Cell Content: F				
	store_id	film_id	title	rentados
1	1	1	ACADEMY DINOSAUR	1
1	4	4	AFFAIR PREJUDICE	1
1	6	6	AGENT TRUMAN	1
1	7	7	AIRPLANE SIERRA	0
1	9	9	ALABAMA DEVIL	0
1	10	10	ALADDIN CALENDAR	1
1	11	11	ALAMO VIDEOTAPE	1
1	12	12	ALASKA PHANTOM	3
1	15	15	ALIEN CENTER	2
1	16	16	ALLEY EVOLUTION	0
1	17	17	ALONE TRIP	0
1	18	18	ALTER VICTORY	3
1	19	19	AMADEUS HOLY	1
1	20	20	AMELIE HELFIGHTERS	0
1	21	21	AMERICAN CTRICKS	3