

UNIVERSIDAD AUTÓNOMA DE MADRID

**ESTRUCTURAS DE DATOS
(2018 – 2019)**

PRÁCTICA 2

Alba Ramos
Miguel Díaz
Grupo: 1211

Madrid, 14/11/2018

Convertir campos id a autoincrementados

Partíamos de la base de datos tweetsdb, que cargamos con el fichero que se nos adjuntaba en el material de la práctica y nos creaba las tablas users, tweets y follows.

Lo primero que hemos hecho ha sido convertir los campos id de las tablas de usuarios y tweets a auto incrementados. Para ello hemos utilizado las siguientes secuencias:

```
--convertir el id de usuario a autoincrementado
CREATE SEQUENCE userid_seq OWNED BY users.user_id;
SELECT setval('userid_seq', coalesce(max(user_id), 0)) FROM users;
ALTER TABLE users ALTER COLUMN user_id SET DEFAULT nextval
('userid_seq');

--convertir el id de tweet a autoincrementado
CREATE SEQUENCE tweetid_seq OWNED BY tweets.tweet_id;
SELECT setval('tweetid_seq', coalesce(max(tweet_id), 0)) FROM tweets;
ALTER TABLE tweets ALTER COLUMN tweet_id SET DEFAULT nextval
('tweetid_seq');
```

De esta forma conseguimos que al insertar un nuevo tweet o usuario, su id se asigne automáticamente, incrementando el último id existente. Si aplicamos estas secuencias, nuestras tablas obtienen la propiedad de id autoincrementado:

```
CREATE TABLE public.users
(
    user_id bigint NOT NULL DEFAULT nextval('userid_seq'::regclass),
    screenname character varying(50) NOT NULL,
    location character varying(100) NOT NULL,
    usercreated timestamp without time zone NOT NULL,
    CONSTRAINT users_pkey PRIMARY KEY (user_id),
    CONSTRAINT users_screenname_key UNIQUE (screenname)
)
WITH (
    OIDS=FALSE
)

CREATE TABLE public.tweets
(
    tweet_id bigint NOT NULL DEFAULT nextval('tweetid_seq'::regclass),
    tweettext text NOT NULL,
    tweettimestamp timestamp without time zone NOT NULL,
    retweet bigint,
    userwriter bigint,
    CONSTRAINT tweets_pkey PRIMARY KEY (tweet_id),
```

Si ahora insertamos dos nuevos usuarios sin especificar el campo id, este se asigna automáticamente,:

	user_id bigint	screenname character varying(50)	location character varying(100)	usercreated timestamp without time zone
1	3377671547	miguel	uam	2004-10-19 10:23:54
2	3377671546	alba	alcobendas	2004-10-19 10:23:54
3	3377671545	murcia1234568		2015-07-15 21:12:07

A continuación, implementamos una serie de programas en C para manejar la base de datos sin tener que hacer que los usuarios escriban las consultas a mano.

Programas odbc

Nuestros programas leen argumentos recibidos por la consola mediante argc y argv. En cada programa comprobamos que los argumentos de entrada sean correctos, ya que según el programa y la función que queramos realizar necesitaremos más o menos parámetros.

Las funcionalidades se implementan mediante consultas en SQL. El programa utiliza las librerías de ODBC y SQL para combinar el lenguaje C y el lenguaje SQL para poder trabajar con nuestras tablas. Primero se establece una conexión con el servidor, que se deberá cerrar al terminar el programa.

Una vez comprobado el retorno correcto, se reserva memoria para un stmt (variable que almacenará nuestra consulta) mediante AllocHandle, que se deberá liberar usando FreeHandle cada vez que queramos realizar una nueva consulta utilizando esta misma variable. La consulta permite la introducción de parámetros. Finalmente, se ejecuta con ExecDirect. Se ha ido realizando un printf de la query escrita para comprobar en cada programa si la consulta devolvía los valores esperados en el pgadmin, y una vez comprobado esto se ha comentado este printf.

Una vez se ha realizado la consulta, se debe realizar BindCol por cada columna que devuelva la misma. Con este proceso se almacenan todas las tuplas de una misma columna en una variable, y más adelante se recorren las mismas mediante un bucle utilizando Fetch para recuperar los valores de cada fila.

A continuación se muestra el proceso de implementación de los diferentes programas.

APPREQ

El programa appreq sirve para hacer diferentes consultas sobre la base de datos dependiendo del comando que introduzcamos. Tiene 5 comandos: user, tweets, retweets, maxrt y maxfw.

En el comando user se hace una consulta de la lista y el número de seguidores que tiene un usuario. Para ello, se tiene que introducir el screenname del usuario que se quiere consultar como argumento. Estas son las consultas que se hacen al escribir el comando:

```
--numero de usuarios seguidos
select count(*)
from follows
where userfollower in
(
    select user_id
    from users
    where screenname = <screenname>
);
--listado de usuarios seguidos
select user_id, screenname
from users
where user_id in
(
    select userfolloweed
    from follows
    where userfollower in
    (
        select user_id
        from users
        where screenname =<screenname>
    )
);
```

```

--numero de usuarios seguidores
select count(*)
from follows
where userfollowed in
(
    select user_id
    from users
    where screenname = <screenname>
);

--listado de usuarios seguidores
select user_id, screenname
from users
where user_id in
(
    select userfollower
    from follows
    where userfollowed in
    (
        select user_id
        from users
        where screenname = <screenname>
    )
)
);

```

Como ejemplo, hemos usado el comando user para ver los usuarios seguidores y seguidos del usuario que se llama furilo, y este es el resultado:

Numero de seguidores = 46	
Lista de seguidores:	
16777682	seguidos de JesusEncinar
19017608	ottoreuss
1562121	denegro
420882631	qiqetin
160688683	Palindron
78285530	DivisionSonora
15234407	EsperanzAguirre
234278716	zemos98
253017888	unmundolibre
362565198	Spotify
6931722	angeljmenez
63732672	medialabprado
15453259	gilbellosta
245338737	fencinar
2237912155	inurrieta
811737	abc_es
17474018	antonello
101932094	vjimenez
6157632	gb_Lucas
41786228	jlanlunez
18944456	viejomeob
14585879	Virginiaspalonso
127974009	iunida
460386348	ozutto
330886345	iescolar
224661284	rosamariaartal
15115726	PSOE
198516419	SaveChildrenEs
37883400	RevistaDon
2353384208	agallor
62237307	luisrull
6809022	ElTerrat
9379912	librosdelko
75236833	fanetin
23746926	Filmin
760839	orange_es
Alvy	juanlusanchez
231187129	AntonioRull
11713422	publico_es
7712702	fabeeoo
	radiocable
	vivitaycoleando
	Guillermolo
	CampusMadrid
	kurioso

También hemos probado las consultas en Pgadmin para asegurarnos de que salía lo mismo:

Número de usuarios seguidos:

	count bigint
1	38

Lista de usuarios seguidos:

	user_id bigint	screenname character varying(50)
1	6809022	Buenafuente
2	224661284	librosdelko
3	41786228	viejomoeb
4	9379912	juanlusanchez
5	7712702	Guillermolo
6	14585879	francisconixon
7	15234407	CERN
8	330886345	PuritoRodriguez
9	460386348	cuerda1936
10	811737	antonello
11	15453259	fencinar
12	2237912155	ElEstadoMental
13	127974009	angelcalleja
14	253017888	manueljabois
15	362565198	norcoreano
16	234278716	mrtfernandez
17	37883400	fanetin
18	63732672	gilbellostata

19	160688683	maria_gomez
20	420882631	sjpedraz
21	2353384208	HeraldodeMadrid
22	231187129	fabeeoo
23	1562121	denegro
24	245338737	ElTioCreespy
25	760839	Alvy
26	11713422	radiocable
27	15115726	mariaramirezNY
28	78285530	mondo_sonoro
29	23746926	albertocontador
30	62237307	Tyrexito
31	19017608	ottoreuss
32	17474018	vjimenez
33	1662959274	Ls_Madriles
34	6931722	angeljimenez
35	75236833	eltiodelmazo
36	18944456	Virginiapalonso
37	198516419	jorgegcastano
38	16777682	JesusEncinar

Número de seguidores:

	count bigint
1	46

Lista de seguidores:

	user_id bigint	screenname character varying(50)
1	2532962834	DivisionSonora
2	14436317	iescolar
3	17186899	SaveChildrenEs
4	7712702	Guillermolo
5	2542067612	CampusMadrid
6	169001667	orange_es
7	10532842	zemos98
8	14349895	unmundolibre
9	1410035282	RevistaDon
10	17185251	kurioso
11	37883400	fanetin
12	63732672	gilbellosta
13	6157632	jlantunez
14	16408241	qiqetin
15	6931722	angeljimenez
16	496786179	vivitaycoleando
17	17230018	Spotify
18	16777682	JesusEncinar
19	5412422	AntonioRull
20	224661284	librosdelko
21	18238467	EsperanzAguirre
22	41786228	viejomoeb

23	9379912	juanlusanchez
24	15644202	inurrieta
25	80899853	rosamariaartal
26	50982086	PSOE
27	811737	antonello
28	19923515	abc_es
29	15453259	fencinar
30	3144281	luisrull
31	17309892	Filmin
32	231187129	fabeeoo
33	1562121	denegro
34	760839	Alvy
35	11713422	radiocable
36	14260032	agallor
37	107520485	Palindron
38	17676713	publico_es
39	19017608	ottoreuss
40	17474018	vjimenez
41	14723378	ozutto
42	14824411	iunida
43	10736762	ElTerrat
44	17772404	medialabprado
45	101932094	gb_Lucas
46	18944456	Virginiapalonso

En el comando tweets se consulta el número de tweets que ha hecho un usuario y el listado de todos sus tweets, ordenados por fecha. Como argumento hay que escribir el screenname del usuario que se quiere consultar. Estas son las consultas que se hacen:

```
--total de tweets del usuario
select count(*)
from tweets
where userwriter in
(
    select user_id
    from users
    where screenname = <screenname>
);
```

```
--listado de tweets del usuario
select tweet_id, tweettimestamp, tweettext
from tweets, users
where userwriter = user_id
and screenname = <screenname>;
```

Como prueba, lo hemos comprobado con el usuario apl_1987:

```
miguel@miguel:~/Documents/EDAT P2/practicas$ ./appreq tweets apl_1987
```

Número de tweets = 69

```
Lista de tweets:
611521212028715009 2015-06-18 15:09:53 "RT @AS_Historia: El @RCDeportivo celebrando su primera Copa del Rey en 1995. #FotograFIAS http://t.co/CEdmzSPmqG"
613622547817570304 2015-06-24 10:19:50 "RT @MarcoAsande: El @RCDeportivo ficha a ALEJANDRO ARRIBAS (@_Arribass), defensa central del @SevillaFC."
614399992061657088 2015-06-26 13:49:07 "RT @RCDeportivo: #ANosaCopa095 XX Aniversario http://t.co/JAJuUs4oIx"
614400211713011712 2015-06-26 13:50:00 "RT @jes1906: Aaaaah Arsenio con la camiseta de Tino desnudo"
614400342755835904 2015-06-26 13:50:31 "RT @ondadeporcoruna: Foto de familia @RCDeportivo Copa 1995... http://t.co/4sEVdtGIaW"
614403804348465152 2015-06-26 14:04:16 "RT @Sisepuedadx: Ahora mismo los haters de Tino... http://t.co/y7sV9FcavP"
614488243973111808 2015-06-26 14:21:55 "Tino en plan... https://t.co/8T4AneSPde"
6144269799961477712 2015-06-26 15:36:22 "RT @RCDeportivo: @XanGf claro, sen problema oh! http://t.co/AQ95lDsEBU"
615681677265768448 2015-07-02 15:37:50 "RT @ondadeporcoruna: #ANosaCores , @RCDeportivo 15/16 http://t.co/9XIIpo8q"
617821418002604032 2015-07-06 00:24:37 "RT @dxtcampeon: Djalma crack de esos dos 2 kilos un buen pelizco se va para hacienda, al final nos deshacemos de Fabri y no da ni para Lucas"
620136028532305926 2015-07-12 09:42:05 "RT @sportando: Grizzlies officially re-sign Marc Gasol http://t.co/vUDtrd1x http://t.co/FEFPqyHDPH"
62077394189749248 2015-07-14 00:14:24 "RT @Anaa_IB a disfrutarlo https://t.co/guy0MW7Iq0"
62095749397248772 2015-07-14 16:46:01 "#Dmrcd da una pista al menos no"
6209774952316212 2015-07-14 17:25:47 "RT @xltcampeon: Cumpliendo con los compromisos https://t.co/09TrFuxjZP"
62099942827377288 2018-09-20 16:05:05 "RT @alvaross23: @alvaross23 puto Tino celtarra sentado al lado de Mourio"
620995149347920384 2018-09-20 16:05:05 "RT @psDavid0: Mejor endiosar al que nos dej el regalito que valorar la mejora desde hace ms de un ao, que we va al da en todo desde ah"
621658716190941185 2018-09-20 16:05:05 "#vodafone_es primero que me llamaran este lunes, luego que tengo que seguir a la espera, luego que la huéiga de tónicos de Tesa..."
62178736065728512 2018-09-20 16:05:05 "#TinoDimisin http://t.co/aQjbEMOquh"
6216409468616164544 2018-09-20 16:05:05 "RT @GALICIANGOLES: Pachi Dopico, na comida das Peas do Dpor: ""Que nos sobrou algn ano de mandato Posiblemente. Pero moi dificil deixa"
6130550881584402433 2018-09-20 16:05:05 "#Dmrcd da una pista al menos no"
6133688161447084932 2018-09-20 16:05:05 "RT @Manuelli0pez: @minitudoRC @RCDeportivo Hay que ir ms al estadio. Decir que los videomarcadores funcionaban bien..."
613643639851431936 2018-09-20 16:05:05 "RT @DeportesCOPEgal: ALEJANDRO ARRIBAS | Se compromete por tres temporadas (ms una opcional) con el @RCDeportivo. Llega libre del @Sevilla"
6136228803045199872 2018-09-20 16:05:05 "RT @tjcope: FTBOL | El Deportivo de la Corua ficha a Alejandro Arribas. http://t.co/uQaY3XzvQ"
613306457665236692 2018-09-20 16:05:05 "RT @marca: #PrimeraHora | Osasuna y Sporting, sancionados sin fichar. Informa @sergiomarca http://t.co/6Fvc5sF9CQ http://t.co/9FvRYC4wJF"
61410506004434944 2018-09-20 16:05:05 "RT @GIGANTEsBasket: Igualada, MVP por su defensa sobre LeBron. Cundo aprendi a defenderlo (Vdeo) http://t.co/DLEllSpvez http://t.co/83"
611957517653028864 2018-09-20 16:05:05 "RT @marcmarquez93: Ya en A Corua con los amigos de Repsol!!! This is Galicia :) http://t.co/4bhwr8w1mt"
61595598295546752 2018-09-20 16:05:05 "RT @GALICIANGOLES: O noso compaero de RAC1 , @gerardromero, anuncia a chegada Deportivo do canteiro blaugrana @AlexCorredera10"
616325689608945664 2018-09-20 16:05:05 "RT @RCDeportivo: David Gmez y lex Corredora, nuevos jugadores del #Fabril para las proximas dos temporadas http://t.co/0GdqDJSY2m http://"
616325888427360256 2018-09-20 16:05:05 "RT @Riera9: Es un placer formar parte de este club, de esta ciudad y de esta aficin los pximos tres aos. https://t.co/3vf66eYzll"
6163599671650287616 2018-09-20 16:05:05 "RT @protomalaga: Nuka y Yaki, cachorritas en adopcin. Nadie se fija en ellas por ser negras... acogidas10@protectoramalaga.com http://t.co/VAt"
616600416304590592 2018-09-20 16:05:05 "RT @AlexCorredera10: Feliz! Soy nuevo jugador del @RCDeportivo ! Con mucha ilusin en esta nueva etapa. Maana empezamos #ForzaDepor http://"
```

Comprobación con Pgadmin:

Número de tweets:

	count bigint
1	69

Lista de tweets:

	tweet_id bigint	tweettimestamp timestamp without time zone	tweettext text
1	620909420273377280	2018-09-20 16:05:05	"RT @alvaross23: @alvaross23 puto Tino celtarra sentado al lado de Mourio"
2	611250645149032448	2018-09-20 16:05:32	"RT @karlinhos_rcd: Nez: ""Tengo mejor recuerdo de Coruas que de Vigo. La aficin del Dpor es de las mejores que me he en"
3	611410506004434944	2018-09-20 16:05:05	"RT @GIGANTEsBasket: Igualada, MVP por su defensa sobre LeBron. Cundo aprendi a defenderlo (Vdeo) http://t.co/DLEllSpvez http://t.co/83"
4	611521212028715009	2015-06-18 15:09:53	"RT @AS_Historia: El @RCDeportivo celebrando su primera Copa del Rey en 1995. #FotograFIAS http://t.co/CEdmzSPmqG"
5	611957517653028864	2018-09-20 16:05:05	"RT @marcmarquez93: Ya en A Corua con los amigos de Repsol!!! This is Galicia :) http://t.co/4bhwr8w1mt"
6	612170736065728512	2018-09-20 16:05:05	"#TinoDimisin http://t.co/aQjbEMOquh"
7	6126409468616164544	2018-09-20 16:05:05	"RT @GALICIANGOLES: Pachi Dopico, na comida das Peas do Dpor: ""Que nos sobrou algn ano de mandato Posiblemente. Pero
8	6130550881584402433	2018-09-20 16:05:05	"@Mrcd da una pista al menos no."
9	6133688161447084032	2018-09-20 16:05:05	"RT @Manuelli0pez: @minitudoRC @RCDeportivo Hay que ir ms al estadio. Decir que los videomarcadores funcionaban bien..."
10	613622547817570304	2015-06-24 10:19:50	"RT @MarcoAsande: El @RCDeportivo ficha a ALEJANDRO ARRIBAS (@_Arribass5), defensa central del @SevillaFC."
11	613643639051431936	2018-09-20 16:05:05	"RT @DeportesCOPEgal: ALEJANDRO ARRIBAS Se compromete por tres temporadas (ms una opcional) con el @RCDeportivo. Llega libre del @Sevilla"
12	614399992061657088	2015-06-26 13:49:07	"RT @RCDeportivo: #ANosaCopa095 XX Aniversario http://t.co/JAJuUs4oIx"
13	614400111024713728	2018-09-20 16:05:05	"RT @RCDeportivo: Benvido @_Arribass5! Xa eres parte d #OSNos, d #ANosaPaixn #DPOR1516 http://t.co/mu0ajFsHCZ"
14	614400186425737217	2018-09-20 16:05:05	"RT @franhermidal: Se curs invitacin al acto a la directiva de 1995, al completo. Han asistido Luis Snchez Dporto, Pat"
15	614400211713011712	2015-06-26 13:50:00	"RT @jes1906: Aaaaah Arsenio con la camiseta de Tino desnudo"
16	614400245712203776	2018-09-20 16:05:05	"RT @RCDeportivo: La Camiseta XX Aniversario d #ANosaCopa095 xa est vendaa nas Deportiendas (44,95) #DPOR #ANosaHistor"
17	614400342755835904	2015-06-26 13:50:31	"RT @ondadeporcoruna: Foto de familia @RCDeportivo Copa 1995.... http://t.co/4sEVdtGIaW"
18	614403804348465152	2015-06-26 14:04:16	"RT @Sisepuedadx: Ahora mismo los haters de Tino... http://t.co/y7sV9FcavP"
19	61440387135665664	2018-09-20 16:05:05	"RT @Joinrazor: Lo de las camisetas es una cortina de humo para que no nos fijemos en la mala gestin de Tino. #adoptau"
20	614404498325405697	2018-09-20 16:05:05	"Vaya sacadas de Tino... Por favor, que alguien le diga que se controle que los haters ya la tienen adentrsimo "
21	614408243973111808	2015-06-26 14:21:55	"Tino en plan... https://t.co/8T4AneSPde"
22	614426979996147712	2015-06-26 15:36:22	"RT @RCDeportivo: @XanGf claro, sen problema oh! http://t.co/AQ95lDsEBU"
23	614789308382334976	2018-09-20 16:05:05	"RT @RCDeportivo: @frajarical Pues claro! Todos celebramos los tltulos al igual que todos pagamos las deudas."

No hemos mostrado todos en la captura de pantalla porque eran muchos, pero al comprobarlo se ve que salen los mismos tweets.

El siguiente comando es retweets. Hay que poner como argumento el id de un tweet, y muestra el número y la lista de tweets que son retweet de este tweet. Hace las siguientes consultas en SQL:

--El total de tweets que son retweet de <tweetid>

```
select count(*)  
from tweets  
where retweet = <tweetid>
```

--Lista de los tweets que son retweet de <tweetid> ordenados por fecha.

```
select tweet_id, screenname, tweettimestamp, tweettext  
from tweets, users  
where tweets.userwriter = users.user_id and retweet = <tweetid>  
order by tweettimestamp
```

Lo hemos probado con el tweet con el id = 610913954089451520:

```
miguel@miguel:~/Documents/EDAT P2/practica$ ./appreq retweets 610913954089451520  
Numero de retweets = 2  
Lista de retweets:  


| tweet_id           | Screenname | tweet_timestamp     | tweet_text                                                      |
|--------------------|------------|---------------------|-----------------------------------------------------------------|
| 610914354658017280 | be_trix    | 2015-06-16 22:58:27 | "RT @Daenerys: Ghost you had one job... http://t.co/3He8wLuZtT" |
| 610915255275945984 | dandel     | 2015-06-16 23:02:01 | "RT @Daenerys: Ghost you had one job... http://t.co/3He8wLuZtT" |



Lo hemos probado con el tweet con el id = 610913954089451520


```

Comprobación con Pgadmin:

Número de retweets:

	count bigint
1	2

Lista de retweets:

	tweet_id bigint	screenname character varying(50)	tweettimestamp timestamp without time zone	tweettext text
1	610914354658017280	be_trix	2015-06-16 22:58:27	"RT @Daenerys: Ghost you had one job... http://t.co/3He8wLuZtT"
2	610915255275945984	dandel	2015-06-16 23:02:01	"RT @Daenerys: Ghost you had one job... http://t.co/3He8wLuZtT"

En el comando maxrt no hay que darle ningun dato como argumento, y nos hace un ranking de los 20 tweets más retweeteados, ordenados por el orden de retweets. Hace la siguiente consulta en SQL:

```
select count(*) retweet, retweet
from tweets
where retweet is not null
group by retweet
order by count(*)
desc limit 20
```

Este es el resultado de ejecutar el comando maxrt:

```
miguel@miguel:~/Documents/EDAT P2/practica$ ./appreq maxrt
--maxrt: ranking
num_retweets      tweet_id          screenname
1296      618767639654047745      Atleti
1243      618873947912806401      Atleti
809       618499974087770116      Atleti
788       621406166179053569      Atleti
579       619567457125203969      Atleti
436       613727219022712832      Torres
411       619111238052802564      lucianovietto36
350       613602566883135488      Atleti
330       618155305835347969      FA82Oficial
327       618149169576968192      Atleti
277       615863062713794560      FA82Oficial
269       619571004390543360      Atleti
255       621419073025495041      Atleti
253       613728972711407616      AntoGriezmann
241       618886192151535616      FA82Oficial
225       616283678755307521      Atleti
219       616282370413498368      Atleti
217       613030266655907840      Atleti
217       620605927499124736      Torres
200      618774294118727680      Atleti
```

Comprobación con Pgadmin:

	retweet bigint	retweet bigint
1	1296	618767639654047745
2	1243	618873947912806401
3	809	618499974087770116
4	788	621406166179053569
5	579	619567457125203969
6	436	613727219022712832
7	411	619111238052802564
8	350	613602566883135488
9	330	618155305835347969
10	327	618149169576968192
11	277	615863062713794560
12	269	619571004390543360
13	255	621419073025495041
14	253	613728972711407616
15	241	618886192151535616
16	225	616283678755307521
17	219	616282370413498368
18	217	613030266655907840
19	217	620605927499124736
20	200	618774294118727680

En el último comando, maxfw, tampoco hay que introducir ningún dato como argumento. Hace un ranking de los 20 usuarios más seguidos ordenados por el número de seguidores. Hace la siguiente consulta en SQL:

```
--maxfw: ranking de los 20 usuarios con mas
seguidores ordenados
select user_id, screenname, count (*)
as total_seguidores
from follows, users
where userfolloweed=user_id
group by user_id
order by total_seguidores desc
limit 20
```

Y nos da el siguiente resultado:

user_id	screenname	total_seguidores
199354750	Atleti	3990
2411221668	Simeone	3234
23791197	policia	2921
394581283	Torres	2844
264120930	Koke6	2781
950341134	AntoGriezmann	2621
13426312	jordievole	2610
199013425	Juanfrantorres	2463
15095537	marca	2451
158342368	Pablo_Iglesias_	2334
2890481403	diegogodin	2289
7996082	el_pais	2226
278508885	olitorres10	2180
426738834	Guaje7Villa	2154
65369125	_anapastor_	2108
362565198	norcoreano	2089
344634424	RafaelNadal	2038
1235864750	M_A_Moya	2018
72811133	maldinisport	2011
2895016114	Ortegaelprofe	1991

Comprobación con pgadmin:

	user_id bigint	screenname character varying(50)	total_seguidores bigint
1	199354750	Atleti	3990
2	2411221668	Simeone	3234
3	23791197	policia	2921
4	394581283	Torres	2844
5	264120930	Koke6	2781
6	950341134	AntoGriezmann	2621
7	13426312	jordievole	2610
8	199013425	Juanfrantorres	2463
9	15095537	marca	2451
10	158342368	Pablo_Iglesias_	2334
11	2890481403	diegogodin	2289
12	7996082	el_pais	2226
13	278508885	olitorres10	2180
14	426738834	Guaje7Villa	2154
15	65369125	_anapastor_	2108
16	362565198	norcoreano	2089
17	344634424	RafaelNadal	2038
18	1235864750	M_A_Moya	2018
19	72811133	maldinisport	2011
20	2895016114	Ortegaelprofe	1991

APPUSER

Nuestra tabla de usuarios inicialmente se encuentra en este estado si mostramos los 3 usuarios con el id más grande:

	user_id bigint	screenname character varying(50)	location character varying(100)	usercreated timestamp without time zone
1	3377671545	murcial234568		2015-07-15 21:12:07
2	3377446216	fitevieu	"Inverness, Scotland"	2015-07-15 16:43:57
3	3377099945	Kellymilladeoro	"Estado Vicente Caldern"	2018-09-20 16:05:46

Si ahora queremos introducir a dos nuevos usuarios, debemos especificar sus screennames, sus ubicaciones y la fecha de creación de cada uno. Como fecha de creación utilizaremos la actual del sistema (localtimestamp). Además, comprobaremos que no exista previamente ese screenname en el sistema, en cuyo caso indicaremos al programa que no realice ninguna acción. Finalmente, se devolverá por pantalla el id asignado a ese usuario, información que será relevante para futuras consultas.

De esta forma, la sentencia SQL quedaría de la siguiente forma:

```
insert into users (screenname, location, usercreated)
values ('screenname', 'location', localtimestamp)
on conflict (screenname) do nothing
returning user_id
```

El screenname y la ubicación se recibirán como argumentos, por lo tanto la query final para pasarle al stmt quedaría como sigue:

```
“insert into users (screenname, location, usercreated)
values ("%s", "%s", localtimestamp)
on conflict (screenname) do nothing
returning user_id”, argv[2], argv[3]
```

Si utilizamos appuser para introducir a los usuarios Alba y Miguel, debemos hacer la llamada al programa como sigue:

```
alba@Alba:~/Descargas/practica$ ./appuser new alba eps
Id del usuario = 3377671546
alba@Alba:~/Descargas/practica$ ./appuser new miguel uam
Id del usuario = 3377671547
alba@Alba:~/Descargas/practica$ █
```

Tras la inserción, se devuelve el id asignado al nuevo usuario, que se va incrementando desde el último introducido. Si ahora mostramos la tabla de usuarios, podemos ver que la inserción se ha realizado correctamente:

	user_id bigint	screenname character varying(50)	location character varying(100)	usercreated timestamp without time zone
1	3377671547	miguel	uam	2018-11-10 13:41:11.246622
2	3377671546	alba	eps	2018-11-10 13:41:02.154237
3	3377671545	murcial234568		2015-07-15 21:12:07

A continuación, procederemos a eliminar usuarios utilizando este programa. Hay que tener en cuenta que un usuario puede tener seguidores, seguidos, tweets y tweets que han sido retuiteados por otros usuarios. Todo esto hay que eliminarlo antes de eliminar al usuario de la tabla users.

Supongamos que tenemos la tabla de tweets en este estado, mostrando los 5 tweets con mayor id. Podemos ver que Alba y Miguel han escrito un tweet cada uno y, además, Miguel ha retuiteado a Alba:

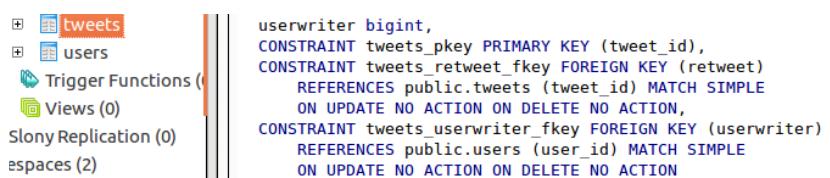
	tweet_id bignum	tweettext text	tweettimestamp timestamp without time zone	retweet bignum	userwriter bignum
1	621733529345466371	hola amigos, soy alba	2018-11-10 13:52:30.999662	621733529345466369	3377671547
2	621733529345466370	hola mundo, acabo de crear me la cuenta, soy miguel	2018-11-10 13:51:07.467436		3377671547
3	621733529345466369	hola amigos, soy alba	2018-11-10 13:50:42.501927		3377671546
4	621733529345466368	"@AndreNexus6 o/"	2015-07-16 19:29:59		492573452
5	621733526203985920	"@_SergioVidal_Casper."	2015-07-16 19:29:58		72934175

Supongamos también que, en la tabla de follows, tenemos la relación de que Alba sigue a Miguel:

	userfollower bigint	userfolloweed bigint
1	3377671546	3377671547
2	3377671545	73372670
3	3377671545	45579875

Si queremos eliminar a Alba, primero tenemos que eliminar los tweets que la hayan retuiteado, después sus tweets, también sus seguidores y seguidos y finalmente el usuario en sí.

La tabla de tweets inicialmente está definida de la siguiente forma:



Para eliminar los tweets, debemos modificar la clave externa retweets para que se borre en cascada. La sentencia SQL sería la siguiente:

```
alter table tweets  
drop constraint tweets_retweet_fkey;
```

```
alter table tweets  
add constraint tweets_retweet_fkey  
foreign key (retweet)  
references tweets (tweet_id)  
on delete cascade;
```

A continuación habría que eliminar los tweets, con la sentencia:

```
delete from tweets
where userwriter in (select user_id from users where
screenname='alba')
```

Finalmente, eliminar sus seguidores y seguidos de la tabla follows, utilizando estas dos sentencias:

```
delete from follows
where userfolloweed in (select user_id from users where
screenname='alba')
```

```
delete from follows
where userfollower in (select user_id from users where
screenname='alba')
```

Por último, se elimina Alba de la tabla users:

```
delete from users where screenname='alba'
```

Como el nombre de usuario a eliminar se va a recibir como argumento, las querys para el stmt serían las siguientes (teniendo en cuenta que cada query la ejecutaremos con un stmt y habrá que hacer un CloseCursor):

```
/*MODIFICAR LA FK RETWEETS PARA BORRAR EN CASCADA*/
"alter table tweets drop constraint tweets_retweet_fkey"
```

```
"alter table tweets add constraint tweets_retweet_fkey foreign key
(retweet) references tweets (tweet_id) on delete cascade"
```

```
/*ELIMINAR SUS TWEETS*/
"delete from tweets where userwriter in (select user_id from users
where screenname='%s')", argv[2]
```

```
/*ELIMINAR SUS SEGUIDORES*/
"delete from follows where userfolloweed in (select user_id from users
where screenname='%s')", argv[2]
```

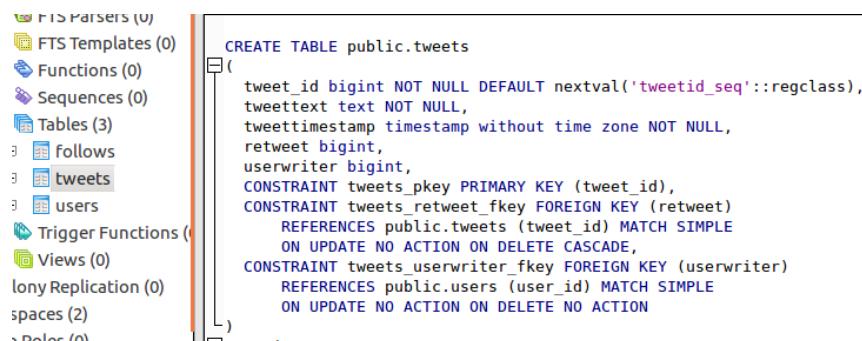
```
/*ELIMINAR SUS SEGUIDOS*/
"delete from follows where userfollower in (select user_id from users
where screenname='%s')", argv[2]
```

```
/*FINALMENTE ELIMINAR EL USUARIO DE LA TABLA USERS*/
"delete from users where screenname='%s'", argv[2]
```

Si ahora vemos lo que ocurre tras ejecutar el programa:

```
alba@Alba:~/Descargas/practica$ ./appuser remove alba
Query: alter table tweets drop constraint tweets_retweet_fkey
Se ha eliminado la FK retweets
Query: alter table tweets add constraint tweets_retweet_fkey foreign key (retweet) references tweets (tweet_id) on delete cascade
Se ha modificado la FK retweets agregando on delete cascade
Query: delete from tweets where userwriter in (select user_id from users where screenname='alba')
Eliminados los tweets del usuario de la tabla tweets
Query: delete from follows where userfolloweed in (select user_id from users where screenname='alba')
Eliminados los seguidores del usuario de la tabla follows
Query: delete from follows where userfollower in (select user_id from users where screenname='alba')
Eliminados los seguidos del usuario de la tabla follows
Query: delete from users where screenname='alba'
Eliminado el usuario de la tabla users
alba@Alba:~/Descargas/practica$
```

Vemos que la tabla de tweets ahora tiene la clave externa retweets modificada para borrado en cascada:

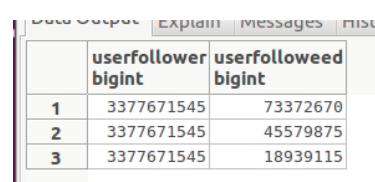


```
CREATE TABLE public.tweets
(
    tweet_id bigint NOT NULL DEFAULT nextval('tweetid_seq'::regclass),
    tweettext text NOT NULL,
    tweettimestamp timestamp without time zone NOT NULL,
    retweet bigint,
    userwriter bigint,
    CONSTRAINT tweets_pkey PRIMARY KEY (tweet_id),
    CONSTRAINT tweets_retweet_fkey FOREIGN KEY (retweet)
        REFERENCES public.tweets (tweet_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE CASCADE,
    CONSTRAINT tweets_userwriter_fkey FOREIGN KEY (userwriter)
        REFERENCES public.users (user_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)
```

Podemos comprobar que la tabla tweets ahora no tiene los tweets de Alba, ni los que Miguel retuiteó de Alba:

	tweet_id bigint	tweettext text	tweettimestamp timestamp without time zone	retweet bigint	userwriter bigint
1	621733529345466370	hola mundo, acabo de crear la cuenta, soy miguel	2018-11-10 14:42:46.071418		3377671547
2	621398691807657984	"Publicidad engaosa lo de #XperiaZ3 la bateria no me dura nada y el mvil se calienta como	2018-09-20 16:05:05		3377671545
3	621405846304653312	"@sonyxperiaeas si, pero se apagar despues de Escribir esto. Como dato; la bateria del LG f	2018-09-20 16:05:32		3377671545

También se ha eliminado la relación de la tabla follows:



	userfollower bigint	userfolloweed bigint
1	3377671545	73372670
2	3377671545	45579875
3	3377671545	18939115

Y, por último, Alba se ha eliminado de la tabla users:

	user_id bigint	screenname character varying(50)	location character varying(100)	usercreated timestamp without time zone
1	3377671547	miguel	uam	2018-11-10 14:40:31.12415
2	3377671545	murcia1234568		2015-07-15 21:12:07
3	3377446216	fitevio	"Inverness, Scotland"	2015-07-15 16:43:57

APPTWEET

Si mostramos los 3 tweets con el id más grande nos sale los siguiente:

	tweet_id bigint	tweettext text	tweettimestamp timestamp without time zone	retweet bigint	userwriter bigint
1	621733529345466368	"@AndreNexus6 o/"	2015-07-16 19:29:59		492573452
2	621733526203985920	"@_SergioVidal_ Casper."	2015-07-16 19:29:58		72934175
3	621733524001959939	"@isanseba Vete abriendo la boca que te la lleno de ser	2018-09-20 16:05:05		171452630

A continuación vamos a añadir un nuevo tweet con el comando new. Para ello tenemos que introducir el screenname del autor y el texto del tweet. Como timestamp utilizaremos la hora que tiene el sistema (localtimestamp). El campo tweet_id se generará solo con la secuencia que hicimos al principio y el campo userwriter será el id del autor. El campo retweet está a null, porque no es un retweet. Esta es la sentencia que se ejecutará en SQL:

```
insert into tweets(tweettext, tweettimestamp, retweet, userwriter)
values(<tweettext>, localtimestamp,null, (select user_id
from users
where screenname = <screenname>))
```

Los argumentos son tweettext y screenname, por lo que la query que le pasamos al stmt es la siguiente:

```
insert into tweets(tweettext, tweettimestamp, retweet, userwriter) values("%s",
localtimestamp,null, (select user_id from users where screenname = '%s'))",
tweettext, argv[2]
```

Para tener todo el texto del tweet en una sola variable (tweettext) hemos ido concatenando todos los argumentos a partir del tercero con un bucle, poniendo espacios entre cada uno, ya que no se podía coger un texto entero con espacios como un solo argumento.

Vamos a insertar un tweet cuyo autor será furilo y su texto será “Buenos días”:

```
miguel@miguel-Aspire-V5-571G ~/Documents/EDAT P2/practica $ ./apptweet new furilo Buenos días
Tweet insertado
```

Y ahora refrescamos la consulta de los 3 tweets con el id más grande en Pgadmin y vemos que ha sido insertado adecuadamente:

	tweet_id bigint	tweettext text	tweettimestamp timestamp without time zone	retweet bigint	userwriter bigint
1	621733529345466380	Buenos días	2018-11-14 11:49:13.691828		13139
2	621733529345466368	"@AndreNexus6 o/"	2015-07-16 19:29:59		492573452
3	621733526203985920	" @_SergioVidal_ Casper . "	2015-07-16 19:29:58		72934175

Si queremos insertar un retweet con el comando retweet, tenemos que introducir como argumento el screenname del usuario que queremos que haga el retweet y el id del tweet que queremos que sea retweeteado. Se creará un nuevo tweet que tendrá como id el siguiente en la secuencia, como texto tendrá el mismo que el tweet original, como timestamp la hora del sistema (localtimestamp), como retweet el id del tweet original y como userwriter el id del usuario que ha hecho el retweet. Esta es la sentencia sql que se hace para insertarlo:

```

insert into tweets(tweettext, tweettimestamp, retweet, userwriter)
values((select tweettext
       from tweets
      where tweet_id = <tweetid>
     ), localtimestamp,<tweetid>, (select user_id
                                         from users
                                        where screenname = <screenname>))

```

Los argumentos son el screenname y el id del tweet, por lo que la query que le pasamos al stmt es la siguiente:

```

"insert into tweets(tweettext, tweettimestamp, retweet, userwriter) values((select
tweettext      from tweets where tweet_id = %s), localtimestamp,%s, (select
user_id from users where screenname = '%s'))", argv[3], argv[3], argv[2]

```

Vamos a insertar un retweet del tweet que hemos creado antes. El retweet lo va a hacer el usuario apl_1987:

Ahora al comprobar en Pgadmin vemos que el retweet se ha insertado correctamente:

	tweet_id bigint	tweettext text	tweettimestamp timestamp without time zone	retweet bigint	userwriter bigint
1	621733529345466382	Buenos días	2018-11-14 11:58:35.139932	621733529345466380	243584581
2	621733529345466380	Buenos días	2018-11-14 11:49:13.691828		13139
3	621733529345466368	"@AndreNexus6 o/"	2015-07-16 19:29:59		492573452

Para borrar un tweet hemos hecho el comando remove. Como argumento solo hay que introducir el id del tweet que se quiere eliminar. Se eliminarán los retweets de este tweet en caso de que los haya, y luego se eliminará el tweet original. Estas son las sentencias en SQL que hacemos:

```

--eliminar los retweets
delete from tweets where retweet = <tweetid>
--eliminar el tweet
delete from tweets where tweet_id = <tweetid>

```

Como el argumento es el id del tweet, este es el query que le pasamos al stmt:

```

/*ELIMINAR LOS RETWEETS*/
"delete from tweets where retweet = %s", argv[2]
/*ELIMINAR EL TWEET*/
"delete from tweets where tweet_id = %s", argv[2]

```

Para comprobarlo vamos a borrar el tweet de furilo que hemos creado antes. Al borrarlo también debería borrarse el retweet que hemos creado de apl_1987:

```
miguel@miguel-Aspire-V5-571G ~/Documents/EDAT P2/practica $ ./apptweet remove 621733529345466380
Rweets eliminados
Tweet eliminado
```

Al comprobarlo en Pgadmin vemos que sí que se han borrado:

	tweet_id bigint	tweettext text	tweettimestamp timestamp without time zone	retweet bigint	userwriter bigint
1	621733529345466368	"@AndreNexus6 o/"	2015-07-16 19:29:59	492573452	
2	621733526203985920	"@ SergioVidal_Casper."	2015-07-16 19:29:58	72934175	
3	621733524001959939	"@isanseba Vete abriendo la boca que te la lleno de seres vivos. Te dejo que los abrace despues."	2018-09-20 16:05:05	171452630	

APPFOLLOW

Si mostramos los 3 primeros resultados de la tabla follows, obtenemos este resultado:

	userfollower bigint	userfolloweed bigint
1	3377671545	73372670
2	3377671545	45579875
3	3377671545	18939115

A continuación vamos a utilizar appfollow para insertar una nueva relación seguidor-seguido conocidos los screennames de los usuarios. La consulta en SQL para realizar esta función es:

```
insert into follows
select u1.user_id, u2.user_id
from users as u1, users as u2
where u1.screenname = 'screenname1' and
u2.screenname='screenname2'.
```

Los screennames de los usuarios son argumentos que recibe el programa, por lo tanto la query final que le pasaremos al stmt es la siguiente:

```
"insert into follows select u1.user_id, u2.user_id from users
as u1, users as u2 where u1.screenname = '%s' and
u2.screenname=%s", argv[2], argv[3]
```

Vamos a insertar la relación de que alba (con id 3377671546) siga a miguel (con id 3377671547). Ejecutamos el programa utilizando el parámetro new:

```
gcc -o appfollow.o appfollow.c -lcdb -o appfollow
alba@Alba:~/Descargas/practica$ ./appfollow new alba miguel
Se ha introducido la relacion de seguidor-seguido
alba@Alba:~/Descargas/practica$
```

Si ahora miramos en la tabla de follows, veremos que se ha creado correctamente la relación solicitada:

	userfollower bigint	userfolloweed bigint
1	3377671546	3377671547
2	3377671545	73372670
3	3377671545	45579875

La query para eliminar una relación de la tabla follows conociendo los screennames (que, en este caso, también se reciben por consola) sería la siguiente:

```
"delete from follows where
userfollower = (
select user_id
from users
where screenname='%os')
and
userfolloweed = (
select user_id
from users
where screenname='%os')", argv[2], argv[3]
```

Si ahora decidimos eliminar la relación previamente introducida usando el mismo programa, introducimos remove:

```
gcc appfollow.o odbc.o -lodbc -o appfollow
alba@Alba:~/Descargas/practica$ ./appfollow new alba miguel
Se ha introducido la relacion de seguidor-seguido
alba@Alba:~/Descargas/practica$ ./appfollow remove alba miguel
Se ha eliminado la relacion de seguidor-seguido
alba@Alba:~/Descargas/practica$
```

Y, efectivamente, vemos que la relación se elimina de la tabla:

	userfollower bigint	userfolloweed bigint
1	3377671545	73372670
2	3377671545	45579875
3	3377671545	18939115