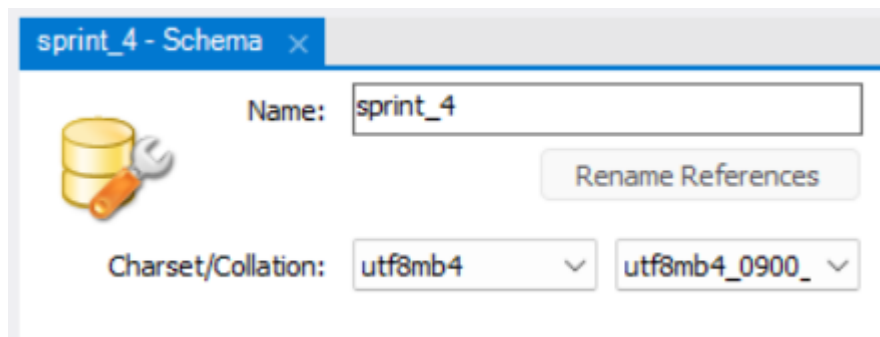


# Nivell 1

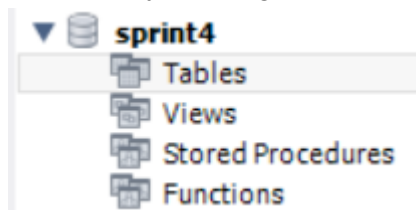
Descàrrega els arxius CSV, estudia'ls i dissenya una base de dades amb un esquema d'estrella que contingui, almenys 4 taules de les quals puguis realitzar les següents consultes:

Antes que nada he bajado los archivos y he revisado la información que tiene cada uno. He realizado un esquema en mi cuaderno en el cual he tomado nota de los datos que incluye cada tabla.

Luego, he creado la base sprint4 (la primera imagen dice sprint\_4 porque fue la que primero había creado y luego la borré, pero de la nueva no hice captura de pantalla por eso he dejado esta).



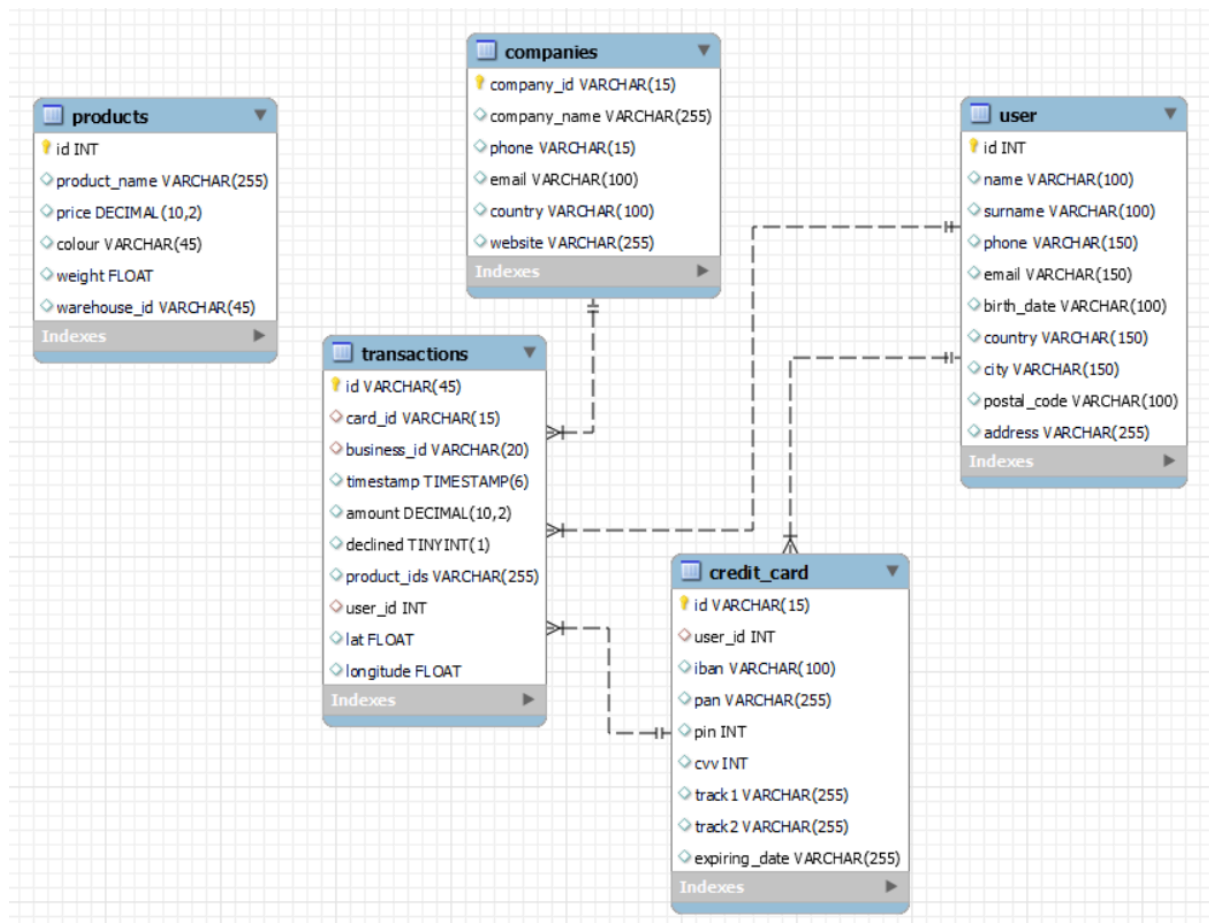
Por lo que ya le tengo



Ahora toca crear las tablas y relaciones. Puede hacerse usando el mysql o sino escribiendo código.

He creado cada tabla indicando las conexiones entre las variables.

Luego de toda la creación (lo hice con crear tabla del sistema y no tengo captura de pantallas de la query de cada tabla, pero de ser necesario haré captura de pantalla del detalle de cada tabla - ya me dirán), he conseguido esto:



He identificado que también hay una relación entre la tabla user (id) y la tabla credit\_card (user\_id).

También he dejado la tabla products sin conectar con transactions pues en una transacción puede haber más de un producto. Originariamente había conectado ambas tablas pero al intentar cargar los datos como venía el archivo noté que importaba solamente 110 registros de los 587 que figuraban en el archivo y eso me llevó a pensar que algo no funcionaba correctamente. Di muchas vueltas hasta entender que en una misma transacción podía haber más de un producto.

La cardinalidad de las relaciones está bien reflejada en el diagrama.

Luego de crear las tablas, procedí a cargar la información que viene en los archivos adjuntos del sprint.

En el caso de la tabla products, tuve que quitarle el símbolo de pesos a la variable price para que lo pudiera tomar correctamente. Eso lo hice con el visual studio.

He definido que debía hacer una única tabla para los usuarios, cargando los tres archivos en la misma tabla y comprobando que no quedara ningún registro fuera.

Finalmente las tablas han quedado completas:

companies

Limit to 50000 rows

1 • SELECT \* FROM sprint4.companies;

	company_id	company_name	phone	email	country	website
▶	b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.porttitor.tellus@yahoo.net	Germany	https://instagram.com/site
	b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia	https://whatsapp.com/group/9
	b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States	https://pinterest.com/sub/cars
	b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.couk	Germany	https://cnn.com/user/110
	b-2238	Ante Taculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand	https://netflix.com/settings
	b-2242	Donec Ltd	01 25 51 37 37	at.iaculis@hotmail.couk	Norway	https://nytimes.com/user/110

Con 100 líneas

credit\_card

Limit to 50000 rows

1 • SELECT \* FROM sprint4.credit\_card;

	id	user_id	iban	pan	pin	cvv	track1	track2	expiring_
▶	CdU-2938	275	TR301950312213576817638661	5424465566813633	3257	984	%B8383712448554646^WovsxeJOpwiev^8604...	%B7653863056044187=800716333673	10/30/22
	CdU-2945	274	DO26854763748537475216568689	5142423821948828	9080	887	%B4621311609958661^UftuyfsSeimxn^06106...	%B4149568437843501=510714033071	08/24/23
	CdU-2952	273	BG451VQL52710525608255	4556 453 55 5287	4598	438	%B2183285104307501^CddyttUxwfdq^5907...	%B6778580257827162=6906859740077	06/29/21
	CdU-2959	272	CR7242477244335841535	372461377349375	3583	667	%B87281111956795320^XocddjBdkecd^09016...	%B4246154489281853=280522391678	02/24/23
	CdU-2966	271	BG72LKTQ15627628377363	448566 886747 7265	4900	130	%B4728932322756223^JhlgvsuFbmwgj^7202...	%B2318571115599881=890821578475	10/29/24

Con 275 líneas

products

Limit to 50000 rows

1 • SELECT \* FROM sprint4.products;

Result Grid Filter Rows: Edit: Export

	id	product_name	price	colour	weight	warehouse_id
▶	1	Direwolf Stannis	161.11	#7c7c7c	1	WH-4
	2	Tarly Stark	9.24	#919191	2	WH-3
	3	duel tourney Lannister	171.13	#d8d8d8	1.5	WH-2
	4	warden south duel	71.89	#111111	3	WH-1
	5	skywalker ewok	171.22	#dbdbdb	3.2	WH-0
	6	dooku solo	136.60	#c4c4c4	0.8	WH--1

Con 100 líneas

transactions

Limit to 50000 rows

1 • SELECT \* FROM sprint4.transactions;

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content: F5

	id	card_id	business_id	timestamp	amount	declined	product_ids	user_id	lat	longitude
▶	02C6201E-D90A-1859-84EE-88D2986D3B02	CcU-2938	b-2362	2021-08-28 23:42:24.000000	466.92	0	71, 1, 19	92	81.9185	-12.5276
	0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	2021-07-26 07:29:18.000000	49.53	0	47, 97, 43	170	-43.9695	-117.525
	063FBA79-99EC-66FB-29F7-25726D1764A5	CcU-2987	b-2250	2022-01-06 21:25:27.000000	92.61	0	47, 67, 31, 5	275	-81.2227	-129.05
	0668296C-CDB9-A883-76BC-2E4C4F8C8AE	CcU-3743	b-2618	2022-01-26 02:07:14.000000	394.18	0	89, 83, 79	265	-34.3593	-100.556
	06CD9AA5-9B42-D684-DDDD-A5E394FEBA99	CcU-2959	b-2346	2021-10-26 23:00:01.000000	279.93	0	43, 31	92	33.7381	158.298
	07A46D48-31A3-7E87-65B9-0DA902AD109F	CcU-3225	b-2386	2021-06-28 21:11:42.000000	340.87	1	47, 23	272	38.8342	92.1905

Con 587 líneas

user

Limit to 50000 rows

1 • SELECT \* FROM sprint4.user;

Result Grid

	id	name	surname	phone	email	birth_date	country	city	postal_code	address
▶	1	Zeus	Gamble	1-282-581-0551	interdum.enim@protonmail.edu	Nov 17, 1985	United States	Lowell	73544	348-7818 Sagittis St.
	2	Garrett	Mcconnell	(718) 257-2412	integer.vitae.nibh@protonmail.org	Aug 23, 1992	United States	Des Moines	59464	903 Sit Ave
	3	Ciaran	Harrison	(522) 598-1365	interdum.feugiat@aol.org	Apr 29, 1998	United States	Columbus	56518	736-2063 Tellus St.
	4	Howard	Stafford	1-411-740-3269	ornare.egestas@icloud.edu	Feb 18, 1989	United States	Kailua	77417	Ap #545-2244 Erat. Rd.
	5	Hayfa	Pierce	1-554-541-2077	et.malesuada.fames@hotmail.org	Sep 26, 1998	United States	Sandy	31564	341-2821 Ultrices Av.
	6	Joel	Tyson	(718) 288-8020	gravida.nunc.sed@yahoo.ca	Oct 15, 1989	United States	Nashville	96838	888-2799 Amet Street

Con 275 líneas

Ahora sí, a resolver ejercicios:

## - Exercici 1

Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.

### #ejercicio 1:

```
SELECT user.id, user.name, user.surname, COUNT(transactions.id)
FROM sprint4.user
JOIN sprint4.transactions
ON user.id = transactions.user_id
GROUP BY user.id, user.name, user.surname
HAVING COUNT(transactions.id) > 30;
```

Result Grid

Filter Rows:

Exp

	id	name	surname	COUNT(transactions.id)
▶	92	Lynn	Riddle	39
	267	Ocean	Nelson	52
	272	Hedwig	Gilbert	76
	275	Kenyon	Hartman	48

Este es el listado de los usuarios con más de 30 transacciones.

## - Exercici 2

Mostra la mitjana de la suma de transaccions per IBAN de les targetes de crèdit en la companyia Donec Ltd. utilitzant almenys 2 taules..

EJERCICIO MODIFICADO POR LAS TUTORAS

## - Exercici 2

Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.

```
11  #ejercicio 2:
12  •  SELECT
13      AVG(amount) AS average_amount, iban
14  FROM sprint4.transactions
15  JOIN sprint4.companies
16  ON transactions.business_id = companies.company_id
17  JOIN sprint4.credit_card
18  ON credit_card.id = transactions.card_id
19  WHERE companies.company_name = 'Donec Ltd'
20  GROUP BY credit_card.iban;
21
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
average_amount	iban			
203.715000	PT87806228135092429456346			

La media de la suma de las transacciones por IBAN de las tarjetas de crédito en la compañía Donec Ltd. es de 203.715.-

## Nivell 2

Crea una nova taula que reflecteixi l'estat de les targetes de crèdit basat en si les últimes tres transaccions van ser declinades i genera la següent consulta:

Para crear la tabla he aplicado la siguiente query:

```

CREATE TABLE IF NOT EXISTS estado_tarj AS (
  WITH trans_card AS
  #ES COMO LLAMAR A LA TABLA QUE ESTOY CREANDO (buscar video de COMMON TABLE EXPRESSION)
  (SELECT transactions.card_id, transactions.timestamp, transactions.declined,
    ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY timestamp DESC) AS row_num
   FROM transactions)
  SELECT trans_card.card_id,
    CASE
      WHEN SUM(trans_card.declined)<=2 THEN 'ACTIVA'
      ELSE 'INACTIVA'
    END AS estado_tarjeta
   FROM trans_card
  WHERE row_num <= 3
  GROUP BY card_id
)
;

```

Ha creado la tabla con 275 líneas (que equivale a la cantidad de tarjetas activas)

Result Grid		
	card_id	estado_tarjeta
▶	CcU-2938	ACTIVA
	CcU-2945	ACTIVA
	CcU-2952	ACTIVA
	CcU-2959	ACTIVA
	CcU-2966	ACTIVA

## Exercici 1

Quantes targetes estan actives?

Si quisiera saber con exactitud únicamente sobre las tarjetas que tienen al menos tres operaciones, y de estas las últimas tres que no tengan las tres declinadas, la query a aplicar sería:

```

WITH trans_card AS
(
  SELECT transactions.card_id, transactions.timestamp, transactions.declined,
  ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY timestamp DESC) AS row_num
  FROM transactions)
SELECT trans_card.card_id,
CASE
  WHEN SUM(trans_card.declined)<=2 THEN 'ACTIVA'
  ELSE 'INACTIVA'
END AS estado_tarjeta
FROM trans_card
WHERE row_num <= 3
GROUP BY card_id
HAVING COUNT(trans_card.card_id) = 3
;

```

Result Grid		Filter Rows:
	card_id	estado_tarjeta
▶	CcU-2938	ACTIVA
	CcU-2959	ACTIVA
	CcU-2994	ACTIVA
	CcU-3393	ACTIVA
	CcU-3981	ACTIVA

Esta consulta arroja un resultado de 9 líneas. Serían 9 las tarjetas que cumplen con las condiciones de tener tres operaciones o más y de esas operaciones en las últimas tres al menos una con una operación aceptada, por lo que la tarjeta no se considera inactiva. Pero siguiendo el razonamiento, todas las tarjetas se encuentran activas.

## Nivell 3

Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada, tenint en compte que des de transaction tens product\_ids. Genera la següent consulta:

Este ejercicio lo dejo para realizar más adelante.

## Exercici 1

Necessitem conèixer el nombre de vegades que s'ha venut cada producte.