

Bases de dades (2019-2020)

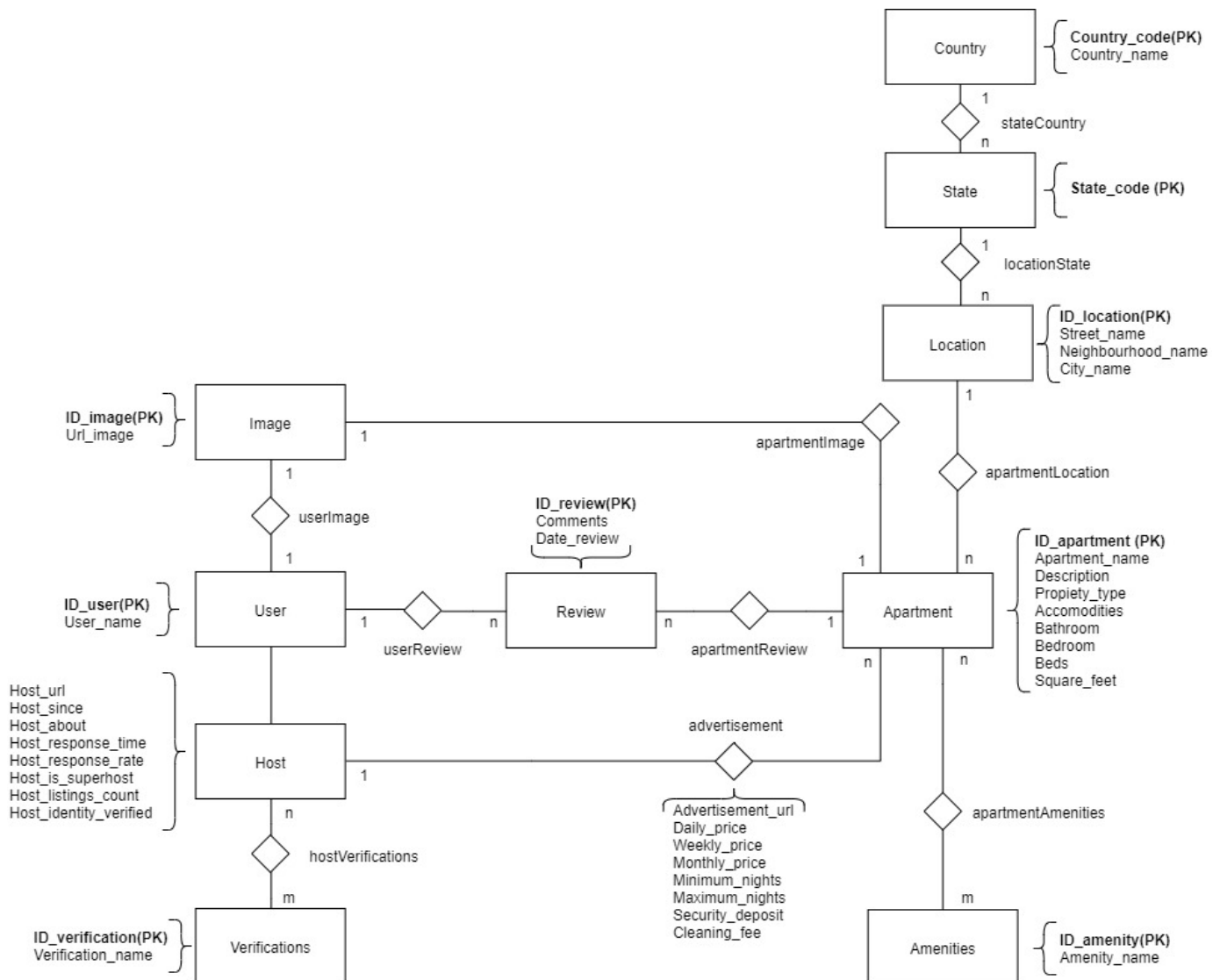
PRÀCTICA 1

Alumnes		
Nom	Cognoms	Login
Curial	Iglesias Clua	curialjosep.iglesias
Pau	Guirao Castells	pau.guirao

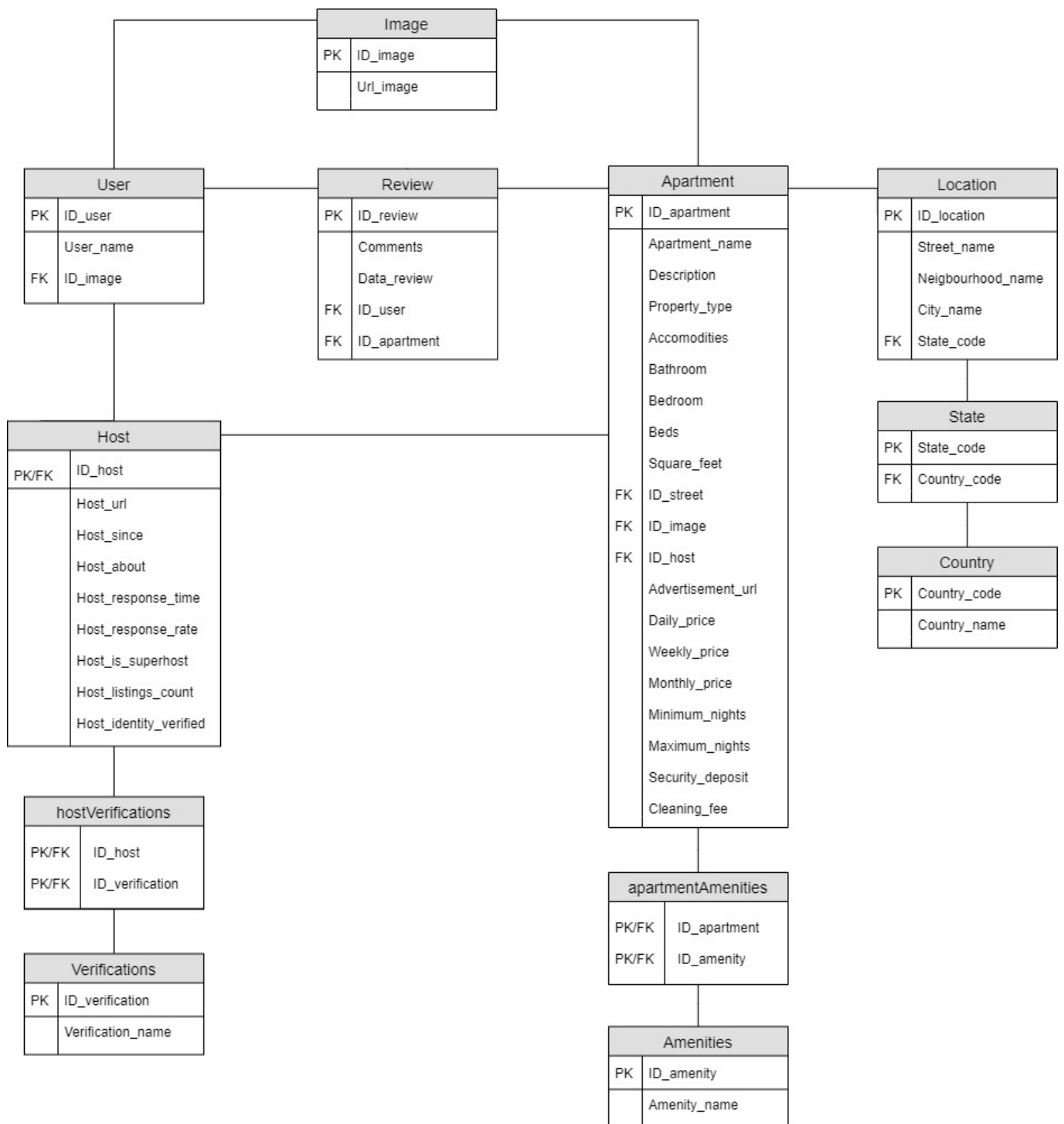
ÍNDEX

1. MODEL CONCEPTUAL.....	2
2. MODEL RELACIONAL.....	3
3. MODEL FISIC	4
4. EXPLICACIÓ DE CADA TAULA.....	7
5. VERIFICACIÓ DE LA IMPORTACIÓ.....	9
6. QUERIES	13
7. CONCLUSIONS.....	25

1. MODEL CONCEPTUAL



2. MODEL RELACIONAL



3. MODEL FISIC

```
DROP TABLE IF EXISTS Country CASCADE;
CREATE TABLE Country (
  Country_code VARCHAR(255),
  Country_name VARCHAR(255),
  PRIMARY KEY (Country_code)
);

DROP TABLE IF EXISTS State CASCADE;
CREATE TABLE State (
  State_code VARCHAR(255),
  Country_code VARCHAR(255),
  PRIMARY KEY (State_code),
  FOREIGN KEY (Country_code) REFERENCES Country (Country_code)
);

DROP TABLE IF EXISTS Location CASCADE;
CREATE TABLE Location (
  ID_location SERIAL,
  Street_name VARCHAR(255),
  Neighbourhood_name VARCHAR(255),
  City_name VARCHAR(255),
  State_code VARCHAR(255),
  PRIMARY KEY (ID_location),
  FOREIGN KEY (State_code) REFERENCES State (State_code)
);

DROP TABLE IF EXISTS Image CASCADE;
CREATE TABLE Image (
  ID_image SERIAL,
  Url_image VARCHAR(255),
  PRIMARY KEY (ID_image)
);

DROP TABLE IF EXISTS "User" CASCADE;
CREATE TABLE "User" (
  ID_user INT,
  User_name VARCHAR(255),
  ID_image INT,
  PRIMARY KEY (ID_user),
  FOREIGN KEY (ID_image) REFERENCES Image (ID_image)
);

DROP TABLE IF EXISTS Host CASCADE;
CREATE TABLE Host (
  ID_host INT,
  Host_url VARCHAR(255),
  Host_since VARCHAR(255),
  Host_about TEXT,
  Host_response_time VARCHAR(255),
  Host_response_rate VARCHAR(255),
  Host_is_superhost CHAR(1),
  Host_listings_count INT,
  Host_identity_verified VARCHAR(255),
  PRIMARY KEY (ID_host),
  FOREIGN KEY (ID_host) REFERENCES "User" (ID_user)
);
```

```
DROP TABLE IF EXISTS Verifications CASCADE;
CREATE TABLE Verifications (
  ID_verification SERIAL,
  Verification_name VARCHAR(255),
  PRIMARY KEY (ID_verification)
);

DROP TABLE IF EXISTS hostVerifications CASCADE;
CREATE TABLE hostVerifications (
  ID_host INT,
  ID_verification INT,
  PRIMARY KEY (ID_host, ID_verification),
  FOREIGN KEY (ID_host) REFERENCES Host (ID_host),
  FOREIGN KEY (ID_verification) REFERENCES Verifications
(ID_verification)
);

DROP TABLE IF EXISTS Apartment CASCADE;
CREATE TABLE Apartment (
  ID_apartment INT,
  Apartment_name VARCHAR(255),
  Description TEXT,
  Property_type VARCHAR(255),
  Accomodities VARCHAR(255),
  Bathroom FLOAT,
  Bedroom INT,
  Beds INT,
  Square_feet INT,
  ID_location INT,
  ID_image INT,
  ID_host INT,
  Advertisement_url VARCHAR(255),
  Daily_price FLOAT,
  Weekly_price FLOAT,
  Monthly_price FLOAT,
  Minimum_nights INT,
  Maximum_nights INT,
  Security_deposit FLOAT,
  Cleaning_fee FLOAT,
  PRIMARY KEY (ID_apartment),
  FOREIGN KEY (ID_location) REFERENCES Location (ID_location),
  FOREIGN KEY (ID_image) REFERENCES Image (ID_image),
  FOREIGN KEY (ID_host) REFERENCES Host (ID_host)
);

DROP TABLE IF EXISTS Amenities CASCADE;
CREATE TABLE Amenities (
  ID_amenity SERIAL,
  Amenity_name VARCHAR(255),
  PRIMARY KEY (ID_amenity)
);

DROP TABLE IF EXISTS apartmentAmenities CASCADE;
CREATE TABLE apartmentAmenities (
  ID_apartment INT,
  ID_amenity INT,
  PRIMARY KEY (ID_apartment, ID_amenity),
  FOREIGN KEY (ID_apartment) REFERENCES Apartment (ID_apartment),
  FOREIGN KEY (ID_amenity) REFERENCES Amenities (ID_amenity)
);
```

```
DROP TABLE IF EXISTS Review CASCADE;
CREATE TABLE Review (
  ID_review SERIAL,
  ID_user INT,
  ID_apartment INT,
  Comments TEXT,
  Data_review DATE,
  PRIMARY KEY (ID_review),
  FOREIGN KEY (ID_user) REFERENCES "User" (ID_user),
  FOREIGN KEY (ID_apartment) REFERENCES Apartment (ID_apartment)
);
```

4. EXPLICACIÓ DE CADA TAULA

L'entitat **Country** guarda el codi d'un país com a PK i el nom del país.

L'entitat **State** guarda el codi d'un estat com a PK i rep el codi d'un país com a FK.

L'entitat **Location** guarda el nom d'una ciutat, el nom d'un barri i el nom d'un carrer, a més rep el codi d'un estat com a FK

L'entitat **Apartment** guarda tota la informació relacionada amb un apartament (nom, descripció, tipus de propietat, comoditats, habitacions etc...). Rep els identificadors de Localització, Imatge i Host com a FK ja que cada apartament només en pot tenir un de cada. De la relació amb Host obté els atributs relacionats amb l'anunci com poden ser el preu o el enllaç.

L'entitat **Amenities** es una llista de tots els amenities que pot contenir un apartament, corresponent cada amenity a una fila diferent.

L'entitat **amenitiesApartment** relaciona les entitats Amenities i Apartment i ens diu tots els possibles amenities que pot tenir cada apartament.

L'entitat **Image** conte els urls de totes les imatges, ja siguin de apartaments o de usuaris, per tant, esta relacionada amb les entitats User i Apartment de forma que el identificador ID_image de cada usuari/apartament correspon a la url de la seva imatge en la taula Image.

L'entitat **User** guarda el nom d'usuari i rep el identificador de la seva imatge com a FK, també rep el seu identificador PK de la taula importació hosts.

Host es una subentitat de User que guarda tots els atributs propis del Host (enllaç, verificació d'identitat, el temps de resposta etc...). Hem decidit en aquest cas fer una generalització ja que Host necessita els atributs que té User però User no necessita els atributs propis del Host.

L'entitat **Review** guarda el comentari, la data en que es va fer i rep el usuari i apartament pel que s'ha escrit ambdós com a FK, se li assigna un identificador com a PK. D'aquesta forma un mateix usuari pot deixar diversos comentaris a un mateix vídeo.

L'entitat **Verificacions** es una llista de totes les verificacions que pot tenir un Host, corresponent cada verificació a una fila diferent.

L'entitat **hostVerifications** relaciona les entitats Host i Verificacions i ens diu totes les possibles verificacions que pot tenir cada host.

5. VERIFICACIÓ DE LA IMPORTACIÓ

Les taules mes importants d'aquest model son Apartment, Host i Review. Contenen la majoria d'informació de les taules d'importació imp_apartments, imp_hosts i imp_reviews. Comprovem que s'hagin inserit totes les files:

```
SELECT COUNT(ia.id) AS num_imp_apartments
FROM imp_apartments AS ia;

SELECT COUNT(a.id_apartment) AS num_Apartment
FROM Apartment AS a;

SELECT COUNT(DISTINCT ih.host_id) AS num_imp_hosts
FROM imp_hosts AS ih;

SELECT COUNT(h.id_host) AS num_Host
FROM Host AS h;

SELECT COUNT(ir.comments) AS num_imp_reviews
FROM imp_reviews AS ir;

SELECT COUNT(r.id_review) AS num_Review
FROM Review AS r;
```

num_imp_apartments	num_imp_hosts	num_imp_reviews
1 22895	1 14988	1 486920
num_apartment	num_host	num_review
1 22895	1 14988	1 486920

Per comprovar que funcioni correctament utilitzarem l'apartament d'ID 10022934 com a exemple.

```
SELECT * FROM Apartment
WHERE id_apartment = '10022934';
```

id_apartment	apartment_name	description	property_type
1	10022934	quiet stylish spacious inner north Melbourne's inner...	Apartment

accomodities	bathroom	bedroom	beds	square_foot
2	1	1	1	<null>

daily_price	weekly_price	monthly_price	minimum_nights	maximum_nights	security_deposit	cleaning_fee
118	<null>	<null>	1	28	150	65

La taula Apartment rep els següents següents FK:

id_location	id_image	id_host
352	11122	15785687

Amb el id_host podem accedir a la informació del propietari de l'apartament.

```
SELECT * FROM Host
WHERE id_host = '15785687';
```

id_host	host_url	host_since	host_about
15785687	https://www.airbnb.com/users/show/15785687	2014-05-20	My partner Andy and I are retired and enjoy all the good things in life.

host_response_time	host_response_rate	host_is_superhost	host_listings_count	host_identity_verified
within an hour	100%	t	1	t

La taula host es una subentitat de User, podem accedir a la informació de User amb el mateix id_host ja que es un FK de id_user.

```
SELECT * FROM "User"
WHERE id_user = '15785687';
```

	id_user	user_name	id_image
1	15785687	Melanie	35060

Tant els usuaris com els apartaments tenen imatges, podem accedir al seu respectiu enllaç amb les següents queris:

```
SELECT i.Url_image
FROM Image AS i, "User" AS u
WHERE i.ID_image = u.ID_image AND u.ID_user = '15785687';
```

url_image
1 https://a0.muscache.com/im/pictures/user/a9abe2d5-c257-40cb-a466-af1443d4b742.jpg?aki_policy=profile_x_medium

```
SELECT i.Url_image
FROM Image AS i, Apartment AS a
WHERE i.ID_image = a.ID_image AND a.ID_apartment = '10022934';
```

url_image
1 https://a0.muscache.com/im/pictures/9ac153d0-7379-45ee-baf4-d555496c11b8.jpg?aki_policy=large

També podem accedir a totes les reviews que han deixat altres usuaris sobre el nostre apartament, en aquest cas han deixat 88 reviews:

```
SELECT * FROM Review
WHERE id_apartment = '10022934';
```

88 rows				
	id_review	id_user	id_apartment	comments
1	92015	25918507	10022934	Melanie was a lovely host and the we felt welcome from the start. The spac. 2016-09-23
2	91995	32495153	10022934	Couldn't have wished for better hosts or better accommodation. I was Melan. 2016-01-24
3	91996	35110905	10022934	This studio apartment was beautiful, clean and comfortable. It was a fanta. 2016-01-28
4	91997	1904937	10022934	We loved Melanie's 1 bedroom apartment, perfect size, location and Melanie. 2016-02-02
5	91998	30180393	10022934	Melanie's studio apartment is just what I've been looking for in Melbourne. 2016-02-29
6	91999	42725802	10022934	Mel's apartment was perfect for my purposes. It is modern, clean, light a. 2016-03-12
7	92000	5869521	10022934	Melanie made us very welcome and was a thoughtful host. The accommodation . 2016-03-15
8	92001	59069122	10022934	We had a very pleasant stay at Melanie's! She was welcoming and we got alo. 2016-06-02
9	92002	29412623	10022934	Mel and Andy are great hosts. They made us feel very welcome and the apart. 2016-06-11
10	92004	64153536	10022934	Had such an enjoyable stay at Mel and Andy's pad. It was clean, spacious a. 2016-06-26

Cada apartament també té una FK ID_location que guarda el carrer, el barri i la ciutat on està situat l'apartament, alhora, la taula Location rep un FK de a quin estat pertany i aquest també pertany a un país, podem accedir a tota aquesta informació amb la següent query:

```
SELECT l.street_name, l.neighbourhood_name, l.city_name, s.state_code,
c.country_name
FROM Location AS l, Apartment AS a, State AS s, Country AS c
WHERE l.id_location = a.id_location AND l.state_code = s.state_code AND
s.country_code = c.country_code AND a.id_apartment = '10022934';
```

	street_name	neighbourhood_name	city_name	state_code	country_name
1	Clifton Hill, VIC, Australia	Yarra	Clifton Hill	VIC	Australia

Cada apartament també té unes amenitats (comoditats). Tots els possibles amenitats que pot tenir un apartament estan en una taula separada, podem accedir als amenitats amb la següent query:

```
SELECT am.amenity_name
FROM Amenities AS am, apartmentAmenities AS aa, Apartment AS a
WHERE am.id_amenity = aa.id_amenity AND a.id_apartment =
aa.id_apartment AND a.id_apartment = '10022934';
```

	amenity_name
1	Dog(s)
2	Hangers
3	TV
4	Iron
5	Host greets you
6	Kitchen
7	Smoke detector
8	Air conditioning
9	Hair dryer
10	Washer
11	Free parking on premises
12	Pets live on this property
13	Wifi
14	translation missing: en.hosting_amenity_49
15	Indoor fireplace
16	Essentials
17	Fire extinguisher
18	Heating
19	First aid kit
20	Dryer
21	Laptop friendly workspace
22	Shampoo

I finalment també podem veure quines verificacions té el host del apartament.

```
SELECT v.Verification_name
FROM verifications AS v, hostverifications AS hv, Host AS h
WHERE v.id_verification = hv.id_verification AND h.id_host = hv.id_host
AND h.id_host = '15785687';
```

	verification_name
1	email
2	jumio
3	government_id
4	phone
5	reviews

6. QUERIES

Query 1

```
SELECT l.city_name AS name, 100 - AVG(a.weekly_price / 7 /  
a.daily_price * 100) AS savings_percentage  
FROM Apartment AS a, Location AS l, Host AS h  
WHERE a.id_location = l.id_location AND a.id_host = h.id_host AND  
h.host_identity_verified LIKE 't'  
GROUP BY l.city_name  
HAVING 100 - AVG(a.weekly_price / 7 / a.daily_price * 100) IS NOT NULL  
ORDER BY savings_percentage DESC  
LIMIT 3;
```

	name	savings_percentage
1	Cranbourne South	79.22077922077922
2	Wattle Glen	64.28571428571428
3	Pakenham	56.04395604395604

Verificació Query 1

La query no dona el mateix resultat que al PDF de referència, però si realitzem la query utilitzant les taules d'importació de Apartments i Hosts sense alterar dona el nostre mateix resultat.

```
SELECT ia.city AS name, 100 - AVG(ia.weekly_price / 7 / ia.price * 100)  
AS savings_percentage  
FROM imp_apartments AS ia, imp_hosts AS ih  
WHERE ia.listing_url = ih.listing_url AND ih.host_identity_verified  
LIKE 't'  
GROUP BY ia.city  
HAVING 100 - AVG(ia.weekly_price / 7 / ia.price * 100) IS NOT NULL  
ORDER BY savings_percentage DESC  
LIMIT 3;
```

	name	savings_percentage
1	Cranbourne South	79.22077922077922
2	Wattle Glen	64.28571428571428
3	Pakenham	56.04395604395604

Query 2

```
SELECT a.Apartment_name AS name, (a.Daily_price/a.Square_feet) AS
price_m2, COUNT(r.id_review) AS reviews
FROM Apartment AS a, Review AS r
WHERE a.id_apartment = r.id_apartment AND a.square_feet <> '0'
GROUP BY a.id_apartment
HAVING COUNT(r.id_review) >= 200
ORDER BY (a.Daily_price/a.Square_feet) DESC
LIMIT 2;
```

	name	price_m2	reviews
1	Bright apartment in vibrant Fitzroy	0.4024767801857585	362
2	The Stables, Island of Richmond	0.37174721189591076	538

Verificació Query 2

La query no dona el mateix resultat que al PDF de referencia que es “The Stables, Island of Richmond”, a nosaltres ens surt com a primer resultat “Bright apartment in vibrant Fitzroy”

Per verificar que hem fet la query correctament, hem localitzat els 2 apartaments obrint el csv en excel i hem realitzat la operació manualment, efectivament “Bright apartment in vibrant Fitzroy” surt mes car que “The Stables, Island of Richmond”.

id	listing_url	name	square_feet	price
4965	1093015	https://www.airbnb.com/rooms/1093015	Bright apartment in vibrant Fitzroy	323 \$130.00

“Bright apartment in vibrant Fitzroy”: $130 / 323 = 0,40$

id	listing_url	name	square_feet	price
21529	999067	https://www.airbnb.com/rooms/999067	The Stables, Island of Richmond	269 \$100.00

“The Stables, Island of Richmond”: $100 / 269 = 0,37$

Query 3

```
SELECT a.Apartment_name AS name, a.Advertisement_url AS url,
a.Daily_price * '5' * '6' + a.Cleaning_fee + a.Security_deposit * '0.1'
AS price
FROM Apartment AS a, Location AS l, Host AS h, Amenities AS am,
apartmentAmenities AS aa
WHERE a.id_location = l.id_location AND a.id_host = h.id_host AND
aa.id_apartment = a.id_apartment AND aa.id_amenity = am.id_amenity
AND h.host_response_rate <> 'N/A' AND a.accomodities = '6' AND
l.neighbourhood_name LIKE 'Port Phillip' AND am.amenity_name LIKE
'Balcony' AND a.bathroom > '1.5' AND CAST(LEFT(h.host_response_rate,-1)
AS INT) > '90'
ORDER BY price ASC
LIMIT 1;
```

	name	url	price
1	Spacious Designer Apartment near Albert Park	https://www.airbnb.com/rooms/12528535	10050

Verificació Query 3

La query dona el mateix resultat que al PDF de referencia.

Hem realitzat la query utilitzant les taules d'importació de Apartments i Hosts sense alterar i dona el nostre mateix resultat.

```
SELECT ia.name AS name, ia.listing_url AS url, ia.price * '5' * '6' +
ia.Cleaning_fee + ia.Security_deposit * '0.1' AS price
FROM imp_apartments AS ia, imp_hosts AS ih
WHERE ia.listing_url = ih.listing_url AND ih.host_response_rate <>
'N/A' AND ia.accommodates = '6' AND ia.neighbourhood_cleansed LIKE
'Port Phillip' AND ia.amenities LIKE '%Balcony%' AND ia.bathrooms >
'1.5' AND CAST(LEFT(ih.host_response_rate,-1) AS INT) > '90'
ORDER BY price ASC
LIMIT 1;
```

	name	url	price
1	Spacious Designer Apartment near Albert Park	https://www.airbnb.com/rooms/12528535	10050

Query 4

```

UPDATE Host
SET Host_is_superhost = 'f'
WHERE host_is_superhost = 't';

UPDATE Host
SET Host_is_superhost = 't'
WHERE host_since LIKE '2014%' OR host_since LIKE '2013%' OR host_since
LIKE '2012%' OR host_since LIKE '2011%' OR host_since LIKE '2010%' OR
host_since LIKE '2009%';

SELECT COUNT(id_host) AS superhosts
FROM Host
WHERE Host_is_superhost = 't'
GROUP BY Host_is_superhost;

SELECT COUNT(id_host) AS normal_hosts
FROM Host
WHERE Host_is_superhost = 'f'
GROUP BY Host_is_superhost;

```

	superhosts
1	4957

	normal_hosts
1	10028

Verificació Query 4

La query dona el mateix resultat que al PDF de referencia.

Captura de verificació de que s'han realitzat els updates correctament.

```

sql> UPDATE Host
      SET Host_is_superhost = 'f'
      WHERE host_is_superhost = 't'
[2020-01-10 18:01:44] 3119 rows affected in 76 ms
sql> UPDATE Host
      SET Host_is_superhost = 't'
      WHERE host_since LIKE '2014%' OR host_since LIKE '2013%' OR host_since LIKE '2012%'
      OR host_since LIKE '2011%' OR host_since LIKE '2010%' OR host_since LIKE '2009%'
[2020-01-10 18:01:44] 4957 rows affected in 83 ms
sql> SELECT COUNT(id_host) AS superhosts
      FROM Host
      WHERE Host_is_superhost = 't'
      GROUP BY Host_is_superhost
[2020-01-10 18:01:44] 1 row retrieved starting from 1 in 11 ms (execution: 6 ms, fetching: 5 ms)
sql> SELECT COUNT(id_host) AS normal_hosts
      FROM Host
      WHERE Host_is_superhost = 'f'
      GROUP BY Host_is_superhost
[2020-01-10 18:01:44] 1 row retrieved starting from 1 in 22 ms (execution: 6 ms, fetching: 16 ms)

```

Query 5

```
SELECT l.Street_name AS street, COUNT(a.id_apartment) AS num,  
AVG(a.daily_price) AS price  
FROM Apartment AS a, Location AS l  
WHERE a.id_location = l.id_location  
GROUP BY l.street_name  
HAVING AVG(a.daily_price) < '100'  
ORDER BY COUNT(a.id_apartment) DESC  
LIMIT 3;
```

	street	num	price
1	Brunswick, VIC, Australia	437	94.80091533180779
2	Footscray, VIC, Australia	161	90.85714285714286
3	Preston, VIC, Australia	142	94.36619718309859

Verificació Query 5

La query dona el mateix resultat que al PDF de referència, però amb una petita variació del nombre de apartaments i en conseqüència també una mica de variació en el preu.

Si realitzem la query utilitzant la taula d'importació de Apartments sense alterar dona el nostre mateix resultat.

```
SELECT Street AS street, COUNT(id) AS num, AVG(price) AS price  
FROM imp_apartments  
GROUP BY street  
HAVING AVG(price) < '100'  
ORDER BY COUNT(id) DESC  
LIMIT 3;
```

	street	num	price
1	Brunswick, VIC, Australia	437	94.80091533180779
2	Footscray, VIC, Australia	161	90.85714285714286
3	Preston, VIC, Australia	142	94.36619718309859

Query 6

```
SELECT u.User_name AS name_reviewer, a.Advertisement_url AS url,
COUNT(r.id_review) AS num_reviews
FROM "User" AS u, Apartment AS a, Review AS r
WHERE u.id_user = r.id_user AND r.id_apartment = a.id_apartment
GROUP BY u.id user, a.id_apartment
ORDER BY COUNT(r.id_review) DESC
LIMIT 3;
```

	name_reviewer	url	num_reviews
1	Cameron	https://www.airbnb.com/rooms/7842455	81
2	Therese	https://www.airbnb.com/rooms/1995746	24
3	Laurie	https://www.airbnb.com/rooms/9239787	23

Creiem que Cameron es qui esta creant reviews falses degut a que deixa moltes reviews fins i tot en un mateix més i totes son molt repetitives.

Verificació Query 6

La query no dona el mateix resultat que al PDF de referencia degut a que la tercera i quarta persona que han fet mes reviews tenen el mateix nombre de reviews fetes, 23. Al limitar per 3 a nosaltres en selecciona la Laurie en comptes del Michael. Això ho podem comprovar limitant per 4 en comptes de 3:

	name_reviewer	url	num_reviews
1	Cameron	https://www.airbnb.com/rooms/7842455	81
2	Therese	https://www.airbnb.com/rooms/1995746	24
3	Michael	https://www.airbnb.com/rooms/9263414	23
4	Laurie	https://www.airbnb.com/rooms/9239787	23

Si realitzem la query utilitzant la taula d'importació de Reviews sense alterar dona el nostre mateix resultat.

```
SELECT u.user_name AS name_reviewer, ir.listing_url AS url,
COUNT(ir.comments) AS num_reviews
FROM imp_reviews AS ir, "User" AS u
WHERE ir.reviewer_id = u.id_user
GROUP BY u.id user, ir.listing_url
ORDER BY COUNT(comments) DESC
LIMIT 4;
```

	name_reviewer	url	num_reviews
1	Cameron	https://www.airbnb.com/rooms/7842455	81
2	Therese	https://www.airbnb.com/rooms/1995746	24
3	Michael	https://www.airbnb.com/rooms/9263414	23
4	Laurie	https://www.airbnb.com/rooms/9239787	23

Query 7

```

SELECT a.id_apartment AS id, a.apartment_name AS name, a.Daily_price *
'2' * '2' + a.Cleaning_fee + a.Security_deposit * '0.1' AS price
FROM Apartment AS a, Location AS l, Host AS h, Amenities AS am,
apartmentAmenities AS aa, hostverifications AS hv, Verifications AS v
WHERE a.id_location = l.id_location AND a.id_host = h.id_host AND
aa.id_apartment = a.id_apartment AND aa.id_amenity = am.id_amenity AND
hv.id_host = h.id_host AND hv.id_verification = v.id_verification
AND a.accomodities >= '2' AND a.beds >= 2 AND l.city_name LIKE 'Saint
Kilda' AND am.amenity_name LIKE 'Kitchen' AND v.verification_name LIKE
'phone' AND (a.Daily_price * '2' * '2' + a.Cleaning_fee +
a.Security_deposit * '0.1') < '5000'
ORDER BY price DESC;

```

	id ↕	name	price ↕
1	30479523	★ Beach at your Doorstep: Enjoy Summer in St Kilda	4166
2	28871882	Marc 102 - 2 Princes St - St Kilda	4046
3	28871774	Marc 108 - 2 Princes St - Adam Klein	4046
4	15901949	Grand Design St kilda house THE BIG ONE	3284
5	21504911	LUXICO - St Kilda Terrace	3139
6	8512458	Designer apartment	2904
7	20241292	ST KILDA CHURCH HOUSE as featured on TV!	2208
8	21982062	Beautiful St Kilda home	2030
9	15737004	Premium property in great location	2000
10	20236151	Superb Victorian House in prime St Kilda location	1976
11	8943497	Victorian Family Terrace - St Kilda	1836
12	28227216	Luxury Vintage Melbourne Loft	1807
13	5932204	St Kilda Townhouse - Stunning & Entire 3 BRM house	1770
14	4672396	Luxurious inner-city haven	1752
15	19738183	Luxurious Spacious St.Kilda Esplanade Apartment	1745
16	16483654	Byron Bay meets Bali (Resort style house)	1740
17	21084320	St Kilda Perfect Location	1685
18	10575312	Modern, Spacious 4BDR St Kilda Home	1650
19	12528535	Spacious Designer Apartment near Albert Park	1600
20	4238287	Design House near Saint Kilda	1550
21	16416097	Huge St Kilda Beach PentHouse Beauty	1540
22	26338530	Stylish and Spacious Contemporary Apartment in St Kilda	1500
23	15044738	4 Bedroom family home in the heart of St Kilda	1480
24	22392766	Family and Large Group Parking WIFI Exc Location	1436
25	13031881	Luxury Modern Home St Kilda	1386
26	23125153	Modern Apart Opposite Albert Park with Breakfast	1360

Verificació Query 7

La query no dona el mateix resultat que al PDF de referencia. Les files que apareixen al exemple també ens apareixen a nosaltres amb el preu correcte però tenim varies files entre mig.

Hem fet la comprovació de dos dels apartaments que ens apareixen entre mig (el segon i el tercer que surten mes cars) i sembla que compleixen amb tots els requeriments que es demana a l'enunciat.

id	name	city	accommodates	beds	amenities	price	security_deposit	cleaning_fee
28871774	Marc 108 - 2 Princes St - Adam Klein	Saint Kilda	4	2	{TV,Wifi,"Air conditioning",Kitchen,"Free f	\$999.00	\$500.00	\$0.00
28871882	Marc 102 - 2 Princes St - St Kilda	Saint Kilda	6	2	{TV,Wifi,"Air conditioning",Kitchen,"Free f	\$999.00	\$500.00	\$0.00

listing_url	name	host_id	host_name	host_verifications
https://www.airbnb.com/rooms/28871774	Marc 108 - 2 Princes St - Adam Klein	217542156	Adam	['email', 'phone']
https://www.airbnb.com/rooms/28871882	Marc 102 - 2 Princes St - St Kilda	217542156	Adam	['email', 'phone']

Tenen mes de 2 accommodates, tenen 2 llits, esta a Saint Kilda, tenen cuina i el host té verificació per telèfon.

Si realitzem el càlcul del preu:

Preu diària x 2 dies x 2 persones + Taxa de neteja + dipòsit de seguretat x 0,1

$\$999 \times 2 \times 2 + \$0 + \$500 \times 0,1 = \4.046 No arriba al límit de \$5.000

Query 8

```
SELECT u.user_name AS name, SUM(1/a.daily_price) * (1 + CASE WHEN
h.host_is_superhost = 't' THEN 1 ELSE 0 END) * COUNT(DISTINCT
hv.ID_verification) * COUNT(DISTINCT a.ID_apartment) AS score
FROM "User" AS u, apartment AS a, Host AS h, hostverifications AS hv
WHERE u.id_user = h.id_host AND h.id_host = a.id_host AND h.id_host =
hv.id_host AND daily_price <> 0
GROUP BY u.id_user, h.id_host
ORDER BY score DESC;
```

	name	score
1	Valeria	9168.473838336065
2	Paul	4250.482573458238
3	Jared SSP	3870.90454545939
4	Adji	3789.1057212146948

Verificació Query 8

```
SELECT ih.host_id AS name, SUM(1/ia.price) * (1 + CASE WHEN
ih.host_is_superhost = 't' THEN 1 ELSE 0 END) * COUNT(DISTINCT
hs.id_verification) * COUNT(DISTINCT ia.id) AS score
FROM imp_apartments as ia, imp_hosts AS ih, hostverifications AS hs
WHERE ia.listing_url = ih.listing_url AND hs.id_host = ih.host_id AND
ia.price <> 0
GROUP BY ih.host_id, ih.host_is_superhost
ORDER BY score DESC;
```

	name	score
1	90729398	9168.473838336073
2	1739996	4250.482573458251
3	8530753	3870.9045454593916
4	22860147	3789.1057212146898

La query no dona el mateix resultat que al pdf però fent la comprovació amb les taules d'importació si que ens dona el mateix resultat. Com que verifications en la importació es una llista no ho podem comprovar directament i hem fet servir la nostre pròpia taula, tot i així hem comprovat que el numero de verificacions de cada host son correctes comparant amb l'excel original.

	E	G	N	O
1	host_id	host_name	host_listings_count	host_verifications
15363	90729398	Valeria	89	['email', 'phone', 'facebook', 'reviews', 'jumio', 'offline_government_id', 'government_id', 'work_email']

	E	G	N	O
1	host_id	host_name	host_listings_count	host_verifications
3551	8530753	Jared SSP	182	['email', 'phone', 'reviews', 'jumio', 'selfie', 'government_id', 'identity_manual']

	E	G	N	O
1	host_id	host_name	host_listings_count	host_verifications
6865	22860147	Adji	90	['email', 'phone', 'reviews', 'manual_offline', 'jumio', 'offline_government_id', 'government_id', 'work_email']

Query 9

```

DROP TABLE IF EXISTS Points10 CASCADE;
CREATE TABLE Points10 (
  ID INT,
  Points INT
);

DROP TABLE IF EXISTS Points15 CASCADE;
CREATE TABLE Points15 (
  ID INT,
  Points INT
);

INSERT INTO Points10 (ID, Points)
SELECT r.ID_user, COUNT(r.ID_review) * '10'
FROM Review AS r
WHERE LENGTH(r.comments) < '100'
GROUP BY r.ID_user;

INSERT INTO Points15 (ID, Points)
SELECT r.ID_user, COUNT(r.ID_review) * '15'
FROM Review AS r
WHERE LENGTH(r.comments) >= '100'
GROUP BY r.ID_user;

SELECT u.User_name AS name, (p10.Points + p15.Points) AS points
FROM Points10 AS p10, Points15 AS p15, "User" AS u
WHERE u.id_user = p10.id AND u.id_user = p15.id
ORDER BY points DESC
LIMIT 10;

```

L'enunciat demana sumar 15 punts si la review té 100 o mes caràcters, però llavors Laurie obté 1520 punts i no els 1505 que es demana en el PDF. En cas de que no es sumin punts quan te just 100 caracters si que s'obté el valor de 1505 que es demana.

Amb `LENGTH(r.comments) >= '100'` | Amb `LENGTH(r.comments) > '100'`

	name	points
1	Laurie	1520
2	Michael	715
3	Thanh	540
4	Andrew	520
5	Suzanne	475
6	Anders	475
7	John	435
8	Jon	425
9	Andrew	425
10	Richard	410

	name	points
1	Laurie	1505
2	Michael	715
3	Thanh	540
4	Andrew	520
5	Suzanne	475
6	Anders	475
7	John	435
8	Jon	425
9	Andrew	425
10	Richard	410

Verificació Query 9

Si realitzem la query utilitzant la taula d'importació de Reviews sense alterar dona el nostre mateix resultat.

```
DROP TABLE IF EXISTS Points10 CASCADE;
CREATE TABLE Points10 (
  ID INT,
  Points INT
);

DROP TABLE IF EXISTS Points15 CASCADE;
CREATE TABLE Points15 (
  ID INT,
  Points INT
);

INSERT INTO Points10 (ID, Points)
SELECT ir.reviewer_id, COUNT(ir.comments) * '10'
FROM imp_reviews AS ir
WHERE LENGTH(ir.comments) < '100'
GROUP BY ir.reviewer_id;

INSERT INTO Points15 (ID, Points)
SELECT ir.reviewer_id, COUNT(ir.comments) * '15'
FROM imp_reviews AS ir
WHERE LENGTH(ir.comments) >= '100'
GROUP BY ir.reviewer_id;

SELECT distinct ir.reviewer_name AS name, (p10.Points + p15.Points) AS
points
FROM Points10 AS p10, Points15 AS p15, imp_reviews AS ir
WHERE ir.reviewer_id = p10.id AND ir.reviewer_id = p15.id
ORDER BY points DESC
LIMIT 10;
```

	name	points
1	Laurie	1520
2	Michael	715
3	Thanh	540
4	Andrew	520
5	Anders	475
6	Suzanne	475
7	John	435
8	Andrew	425
9	Jon	425
10	Richard	410

Query 10

Selecciona el nom del host, del seu apartament i el nombre de reviews obtingudes de tots els hosts que no siguin super-host, que el seu apartament estigui a la ciutat de Melbourne amb 2.5 lavabos o mes i amb wifi, i que el seu nom no sigui mes llarg que el nombre de reviews obtingudes, ordenats per llargada de nom limitat per 5.

```
SELECT u.user_name AS name, a.apartment_name AS Apartment,
COUNT(r.ID_review) AS num_reviews
FROM Apartment AS a, Location AS l, Host AS h, Amenities AS am,
apartmentAmenities AS aa, "User" AS u, review as r
WHERE u.id_user = h.id_host
AND a.id_host = h.id_host
AND a.id_location = l.id_location
AND a.id_apartment = aa.id_apartment
AND am.id_amenity = aa.id_amenity
AND r.id_apartment = a.id_apartment
AND h.host_is_superhost = 'f'
AND l.city_name = 'Melbourne'
AND a.bathroom >= '2.5'
AND am.amenity_name LIKE 'Wifi'
GROUP BY u.id_user, a.apartment_name
HAVING COUNT(r.ID_review) > LENGTH(u.user_name)
ORDER BY LENGTH(u.user_name) DESC
LIMIT 5;
```

	name	apartment	num_reviews
1	Pinnacle Luxury	Pinnacle Luxury Serviced Apartment on LaTrobe St	86
2	Catherine	Mackenzie St Melb luxury apartment	11
3	Charlotte	Colourful Penthouse Apartment with Incredible Skyline Views	25
4	Ming Yu	Comfort Zone in Melbourne City 1 & 5	22
5	Vanessa	CBD Absolute Centre Niagara Lane 3/br	134

7. CONCLUSIONS

Aquesta pràctica posa a prova la capacitat des alumnes per aplicar els conceptes i la teoria explicades tant a classe com a les sessions de laboratori en un cas pràctic com aquest relacionat amb el tractament de les dades d'una empresa on la gent pot llogar, alquilar i deixar valoracions.

En la fase 1 no hem tingut gaires dificultats amb el plantejament del model conceptual, el model relacional i el model físic trient el tractament de les dades dels atributs “amenities” i “host_verifications” que son llistes.

Hem tingut molts problemes en la part d'importació de dades a les taules normalitzades del model físic degut a que hem tingut molts errors relacionats amb les claus foranes de la taula Apartments així com errors acumulats d'altres taules com User o Image, hem pogut solucionar aquests errors preguntant a becaris i dedicant moltes hores a solucionar-los.

En la fase 2 hem tingut alguns problemes com que hem trobat coses que no funcionaven i hem agut de corregir els models de la fase 1, no hem aconseguit realitzar la query 8.

Estem contents amb el resultat d'aquesta pràctica i creiem que ens ha servir per reforçar els coneixements de classe.