

Co-working Space Management System

Data Management for Analytics

Milestone: Implementation in MySQL

Group 24

Aditi Chadha

Harsh Shingala

857-391-4247

781-985-1057

chadha.ad@northeastern.edu

shingala.h@northeastern.edu

Percentage of Effort Contributed by Student1: 50%

Percentage of Effort Contributed by Student2: 50%

Signature of Student 1: Aditi Chadha

Signature of Student 2: Harsh Shingala

Submission Date: 11/05/22

Project Coworking Spaces

Saturday, November 5, 2022 1:55 PM

Implementation Summary:

We have followed the following steps for MySQL implementation for the project:

1. **Data Generation & Storage:**

The data has been generated using [mockaroo](#) website. Additionally the data was manually refined in excel to fine tune the format required by MySQL. The generated data was stored in csv file.

2. **Create Schema and tables:**

Using DDL commands, we created the database named 'coworking' and their respective tables. Datatypes for each column, Foreign key, Primary Key as well as Not Null constraints were specified while creating a table.

3. **Database Population:**

Used Import csv option to upload the data into respective tables.

1. Creating schema named `coworking`

```
CREATE SCHEMA `coworking` ;
```

2. Creating 'Building' table

```
CREATE TABLE Building (
  Id int NOT NULL,
  BuildingNumber int NOT NULL,
  StreetName varchar(255) NOT NULL,
  StreetSuffix varchar(255) NOT NULL,
  Neighbourhood varchar(255),
  ZipCode int NOT NULL,
  Latitude float NOT NULL,
  Longitude float NOT NULL,
  Status char(255) NOT NULL,
  PRIMARY KEY (Id)
);
```

3. Creating 'Building_floor' table

```
CREATE TABLE Building_floor (
  BuildingId int NOT NULL,
  FloorNumber int NOT NULL,
  Area int,
  shared_floor tinyint(3) NOT NULL,
  PRIMARY KEY (BuildingId, FloorNumber),
  FOREIGN KEY (BuildingId) REFERENCES Building (Id)
);
```

4. Creating 'Spaces' table

```
CREATE TABLE Spaces (
  Id int NOT NULL AUTO_INCREMENT,
  SpaceType varchar(255) NOT NULL,
  MaxCapacity int NOT NULL,
  SpaceDescription varchar(255),
  SpaceName varchar(255) NOT NULL,
  PerSeatCost numeric(5, 2) NOT NULL,
  Availability varchar(255) NOT NULL,
  BuildingId int NOT NULL,
  FloorNumber int NOT NULL,
```

```

    PRIMARY KEY (Id),
    FOREIGN KEY (BuildingId, FloorNumber) REFERENCES Building_floor
(BuildingId, FloorNumber)
);

```

5. Creating 'Users' table

```

CREATE TABLE Users (
    Id int NOT NULL AUTO_INCREMENT,
    FirstName varchar(255) NOT NULL,
    LastName varchar(255),
    ContactNumber varchar (255) NOT NULL UNIQUE,
    PRIMARY KEY (Id)
);

```

6. Creating 'CorporatePoc' table

```

CREATE TABLE CorporatePoc (
    UsersId int NOT NULL,
    CorporateEmail varchar(255) NOT NULL,
    SSN varchar(255) NOT NULL UNIQUE,
    PRIMARY KEY (UsersId),
    FOREIGN KEY (UsersId) REFERENCES Users (Id)
);

```

7. Creating 'Freelancer' table

```

CREATE TABLE Freelancer (
    UsersId int NOT NULL,
    Email varchar(255) NOT NULL UNIQUE,
    IdentificationType varchar(255) NOT NULL,
    IdentificationNum varchar(35) NOT NULL,
    HouseNumber int NOT NULL,
    StreetName varchar(255) NOT NULL,
    StreetSuffix varchar(255) NOT NULL,
    City varchar(255) NOT NULL,
    State varchar(255) NOT NULL,
    ZipCode int NOT NULL,
    PRIMARY KEY (UsersId),
    FOREIGN KEY (UsersId) REFERENCES Users (Id)
);

```

8. Creating 'Corporate' table

```

CREATE TABLE Corporate (
    Id int NOT NULL AUTO_INCREMENT,
    CorporateName varchar(255) NOT NULL,
    EIN varchar(35) NOT NULL UNIQUE,
    UnitNum int NOT NULL,
    BuildingName varchar(255) NOT NULL,
    StreetName varchar(255) NOT NULL,
    StreetSuffix varchar(255) NOT NULL,
    City varchar(255) NOT NULL,
    State varchar(255) NOT NULL,
    ZipCode int NOT NULL,
    Website varchar(255) NOT NULL,
    CorporatePocUsersId int NOT NULL UNIQUE,
    FOREIGN KEY (CorporatePocUsersId) REFERENCES CorporatePoc (UsersId),
    PRIMARY KEY (Id)
);

```

9. Creating 'Employees' table

```

CREATE TABLE Employees (
    Id int NOT NULL AUTO_INCREMENT,
    FirstName varchar(255) NOT NULL,
    LastName varchar(255),

```

```
Email varchar(255),
CorporateId int NOT NULL,
PRIMARY KEY (Id),
FOREIGN KEY (CorporateId) REFERENCES Corporate (Id)
);
```

10. Creating 'SpaceBooking' table

```
CREATE TABLE SpaceBooking (
  Id int NOT NULL AUTO_INCREMENT,
  NumSeatsBooked int NOT NULL,
  BookingTime timestamp NOT NULL,
  StartDate date NOT NULL,
  EndDate date NOT NULL,
  SpaceId int NOT NULL,
  UsersId int NOT NULL,
  PRIMARY KEY (Id),
  FOREIGN KEY (UsersId) REFERENCES Users (Id),
  FOREIGN KEY (SpaceId) REFERENCES Spaces (Id)
);
```

11. Creating 'Service' table

```
CREATE TABLE Service (
  Id int NOT NULL AUTO_INCREMENT,
  Type varchar(255) NOT NULL,
  ServiceCost numeric(3, 2) NOT NULL,
  PRIMARY KEY (Id)
);
```

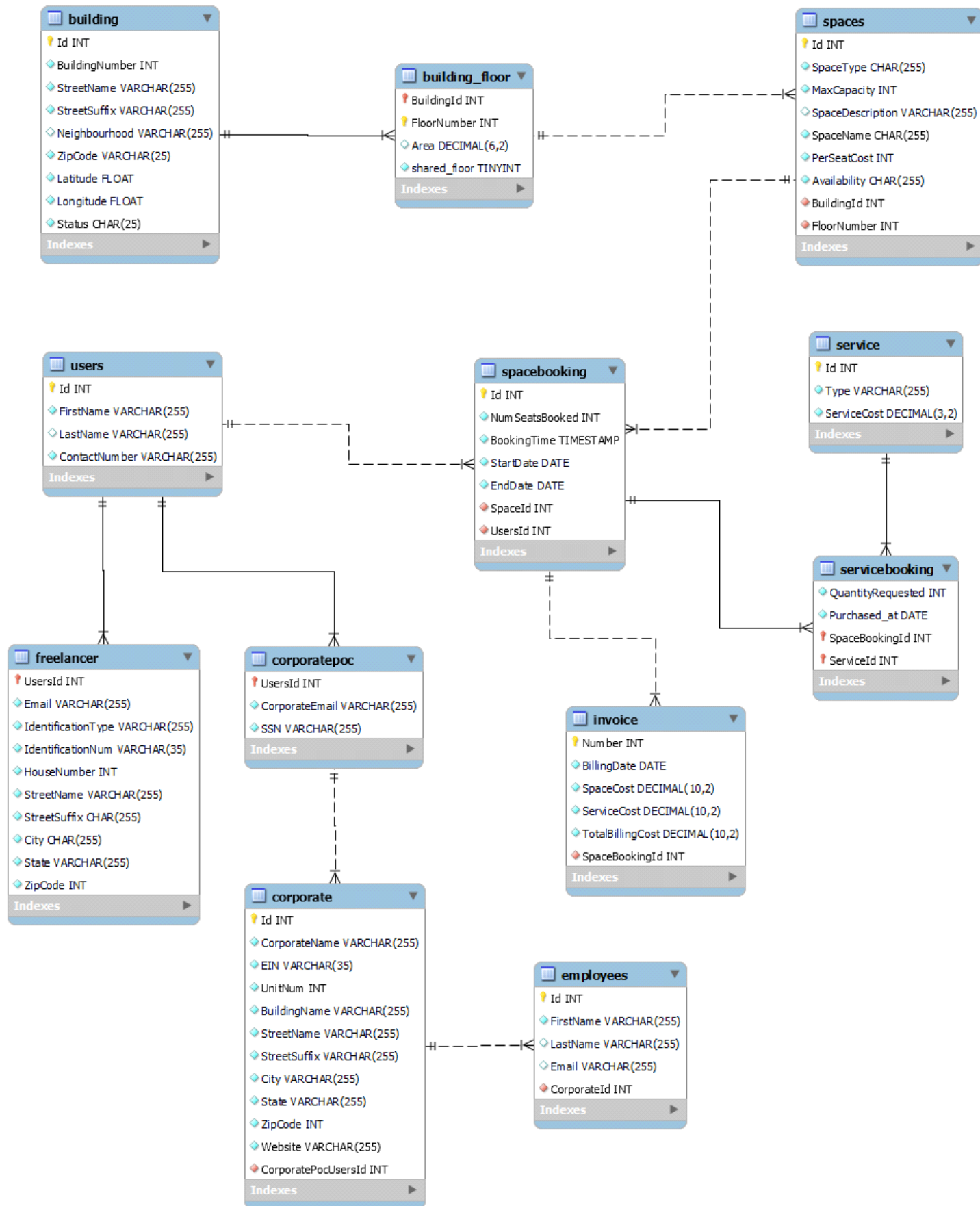
12. Creating 'ServiceBooking' table

```
CREATE TABLE ServiceBooking (
  QuantityRequested int NOT NULL,
  Purchased_at date NOT NULL,
  SpaceBookingId int NOT NULL,
  ServiceId int NOT NULL,
  PRIMARY KEY (SpaceBookingId, ServiceId),
  FOREIGN KEY (SpaceBookingId) REFERENCES SpaceBooking (Id),
  FOREIGN KEY (ServiceId) REFERENCES Service (Id)
);
```

13. Creating 'Invoice' table

```
CREATE TABLE Invoice (
  Number int NOT NULL AUTO_INCREMENT,
  BillingDate date NOT NULL,
  SpaceCost numeric(10, 2) NOT NULL,
  ServiceCost numeric(10, 2) NOT NULL,
  TotalBillingCost numeric(10, 2) NOT NULL,
  SpaceBookingId int NOT NULL UNIQUE,
  PRIMARY KEY (Number),
  FOREIGN KEY (SpaceBookingId) REFERENCES SpaceBooking (Id)
);
```

EER diagram using reverse engineering from MySQL workbench



Importing Data into Tables

1. Building

	Id	BuildingNumber	StreetName	StreetSuffix	Neighbourhood	ZipCode	Latitude	Longitude	Status
▶	121	130	Brickson Park	Junction	North End	02113	137.013	35.0553	TRUE
	169	268	Westridge	Trail	Charlestown	02129	2.49439	48.9319	TRUE
	256	238	Kenwood	Alley	Allston	02134	113.406	-8.09896	TRUE
	285	211	Hovde	Center	Roxbury Crossing	02120	120.167	30.2733	TRUE
	308	206	Kennedy	Terrace	Roxbury Crossing	02120	14.7218	51.9785	TRUE
	350	225	Eggendart	Trail	Fenway-Kenmore	02115	-57.5897	-37.9642	TRUE
	395	1411	Tennessee	Place	Bay Village	02116	13.6137	46.0531	TRUE
	399	114	5th	Center	Chinatown	02111	70.3361	35.2303	TRUE
	457	107	Evergreen	Center	East Boston	02128	6.15257	5.84671	TRUE
	503	153	Ronald Regan	Place	South End	02118	18.0442	59.3476	TRUE
	508	238	Surrey	Alley	Roslindale	02131	120.125	30.2827	TRUE

building 3 × Apply

2. Building_Floor

Result Grid					Filter Rows:	Edit:
	BuildingId	FloorNumber	Area	shared_floor		
▶	121	3	174.00	1		
	121	6	114.00	1		
	169	2	105.00	1		
	169	6	181.00	1		
	256	1	116.00	1		
	256	3	166.00	1		
	256	6	182.00	1		
	285	3	100.00	1		
	285	4	400.00	1		
	285	8	336.00	1		
	308	1	144.00	1		
	350	4	266.00	1		

building_floor4 ×

3. Spaces

Id	SpaceType	MaxCap	SpaceDescription	SpaceName	PerSeatCost	Availability	BuildingId	FloorNumber
1	Common_Area	80	A common area provid...	Lacoste	16	1	350	3
2	Common_Area	35	A common area provid...	Burchett	17	1	508	4
3	Meeting_Room	10	A meeting room compri...	Grenshields	14	1	121	3
4	Meeting_Room	25	A meeting room compri...	De Laspee	16	1	855	2
5	Meeting_Room	10	A meeting room compri...	Winkworth	10	1	921	2
6	Meeting_Room	10	A meeting room compri...	Cobbled	19	1	915	2
7	Common_Area	45	A common area provid...	Burnep	16	1	395	1
8	Common_Area	80	A common area provid...	Saxton	18	1	169	2
9	Meeting_Room	25	A meeting room compri...	Mappleth...	17	1	711	2
10	Common_Area	80	A common area provid...	Franchyo...	17	1	932	3
11	Meeting_Room	10	A meeting room compri...	Ohm	15	1	285	3

4. Users

	Id	FirstName	LastName	ContactNumber
▶	1	Shandeigh	Hunt	894-354-5261
	2	Emmalee	Kingdom	263-978-2384
	3	Justino	Redhead	281-555-9277
	4	Sam	Huntington	256-252-6214
	5	Joane	Fresson	372-451-7855
	6	Junie	Capper	915-309-3381
	7	Caria	Felten	341-294-9960
	8	Cherry	Bischoff	776-843-1290
	9	Fin	Ginity	467-263-6265
	10	Sonja	Morforth	143-839-5344
	11	Care	Yushachkov	860-770-0734
	12	Alida	Chisney	203-173-3004

users 6 ×

5. Freelancers

	UsersId	Email	IdentificationType	Identification	House	StreetName	StreetSuffi	City	State	ZipCode
▶	1	vhyrons0@...	Driving_License	129-12-4822	63	Valley Edge	Lane	Paden City	West Virginia	84552
	2	klambart1...	SSN	238-92-7920	46	Bluejay	Drive	West Boylston	Massachusetts	53356
	3	dtiddy2@w...	SSN	272-79-6282	34	Portage	Park	Chenango F...	New York	87578
	4	schoppen3...	Passport_num	708-17-6341	24	Fair Oaks	Center	Lone Wolf	Oklahoma	24484
	5	crodgier4@...	Driving_License	407-27-6111	81	Talmadge	Trail	Grafton	Massachusetts	26355
	6	ldax5@sha...	SSN	117-46-1736	21	Swallow	Center	Gillett	Pennsylvania	18504
	7	dtabary6@...	SSN	332-30-6029	77	Eastwood	Court	Malvern	Pennsylvania	67784
	8	jhospital7@...	SSN	509-08-4535	82	Scoville	Crossing	Kurtz	Indiana	16296
	9	cdeuley8@...	Passport_num	240-27-9476	70	Helena	Point	Hubbell	Michigan	58322
	10	cphaup9@...	Driving_License	760-55-8826	79	Sheridan	Trail	Hendrix	Oklahoma	99489
	11	mmaharga...	Passport_num	331-90-6617	69	Butterfield	Junction	Grosvenor ...	Connecticut	23199
	12	demark0@...	SSN	428-88-5863	64	Underbank	Hill	Hallidaysburg	Pennsylvania	60356

freelancer 7 ×

Apply

6. CorporatePoc

	UsersId	CorporateEmail	SSN
▶	19	rtodhunter9@quantcast.com	888-04-9718
	20	lshuttlewoodt@vk.com	626-42-3399
	23	fconryr@vkontakte.ru	892-61-5992
	29	rcourcey3@ftc.gov	535-49-7317
	30	jvaughten6@java.com	233-53-0421
	42	sbachnic0@edublogs.org	777-06-3038
	51	athorndalei@alexa.com	606-86-4014
	56	dstringmang@clickbank.net	294-18-5322
	60	meverss@dropbox.com	567-39-0267
	65	hleglise13@omniture.com	150-15-4549
	67	neverse@posterous.com	782-54-3835
	72	brunichanal@aimmedia.com	608-31-6751

corporatepoc 8 ×

7. Corporate

	Id	CorporateName	EIN	UnitNum	BuildingName	StreetName	StreetSt	City	State	ZipCode	Website	CorporatePocUsersId
▶	1	Brainverse	70-239...	42	Faulkner	Buell	Drive	Ma...	New Jersey	19905	www...	42
	2	Meembee	11-912...	262	Heinle	Longview	Avenue	On...	Connecticut	46837	www...	174
	3	Meevee	66-608...	51	Rikkard	Aberg	Park	Ell...	Wisconsin	17517	www...	234
	4	Zoomzone	96-786...	303	Scheu	Cottonw...	Junction	Mo...	New York	11683	www...	29
	5	Jazzy	90-455...	235	Mozzetti	Evergreen	Cross...	Wi...	Texas	48488	www...	182
	6	Skaboo	27-685...	293	Amthor	Cody	Street	Hu...	Pennsylv...	93345	www...	168
	7	Rhynyx	45-978...	156	Greatba...	Ridge Oak	Way	Sk...	Illinois	87040	www...	30
	8	Yodo	13-780...	150	Lacaze	Loftsgordon	Way	Bri...	Pennsylv...	76688	www...	165
	9	Quinu	70-704...	163	Maling	Summit	Drive	Wa...	Illinois	92262	www...	263
	10	Edgetag	88-299...	210	Bendix	Tennyson	Place	Da...	New Jersey	49551	www...	19
	11	Camido	92-387...	300	Stockney	Hoepker	Hill	Wa...	Michigan	81796	www...	76
	12	Udder...	95-040...	404	Udder...	Red Cloud	Way	Ne...	Idaho	25025	www...	170

corporate 9 × Apply

8. Employees

	Id	FirstName	LastName	Email	CorporateId
▶	1	Jonell	Siemon	jsiemon0@ning.com	3
	2	Raven	Haythorne	rhaythorne1@columbia.edu	3
	3	Sosanna	Tuohy	stuohy2@networkadvertising.org	3
	4	Kermit	Baudesson	kbaudesson3@devhub.com	3
	5	Claudette	Quenell	cquenell4@seattletimes.com	3
	6	Kienan	Yerrill	kyerrill5@addthis.com	3
	7	Stanislaw	de Copeman	sdecopeman6@intel.com	3
	8	Myrle	Blaskett	mblaskett7@hexun.com	3
	9	Mignon	Nickols	mnickols8@examiner.com	3
	10	Sherilyn	Moreland	smoreland9@rakuten.co.jp	3
	11	Rodney	Lambot	rlambota@lycos.com	3
	12	Shad	Udder...	shaduck@comcast.net	3

employees 10 ×

9. Services

	Id	Type	ServiceCost
▶	1	Printing	0.10
	2	Gym	2.00
	3	Scanning	0.10
	4	8-hour-Parking	4.00
	5	3-hour-Parking	2.00
	6	Pantry-Food	1.50
	7	Pantry-Beverage	1.00

10. ServiceBooking

	QuantityRequested	Purchased_at	SpaceBookingId	ServiceId
▶	9	2021-12-09	5	4
	8	2022-10-19	8	6
	8	2022-08-24	10	3
	8	2022-10-23	10	5
	5	2021-12-10	11	1
	6	2022-04-07	15	2
	8	2022-04-05	21	2
	5	2021-12-10	21	5
	8	2022-01-20	23	2
	5	2022-01-01	24	1
	5	2022-03-13	26	5
	6	2022-04-02	27	6

servicebooking 12 ×

11. SpaceBooking

	Id	NumSeatsBooked	BookingTime	StartDate	EndDate	SpaceId	UsersId
▶	1	3	2022-01-01 15:05:41	2022-01-24	2022-02-02	56	127
	2	16	2022-01-02 03:46:25	2022-01-18	2022-01-31	13	129
	3	61	2022-01-03 21:07:44	2022-01-23	2022-02-10	74	126
	4	11	2022-01-16 14:19:48	2022-02-01	2022-02-20	166	71
	5	94	2022-01-28 00:13:58	2022-02-27	2022-03-24	95	179
	6	31	2022-01-29 08:14:38	2022-01-31	2022-02-16	145	297
	7	85	2022-01-30 08:10:21	2022-02-09	2022-02-10	196	167
	8	52	2022-02-01 08:22:26	2022-02-28	2022-03-01	163	135
	9	62	2022-02-02 19:33:48	2022-02-17	2022-02-27	117	240
	10	34	2022-02-08 00:16:05	2022-03-10	2022-04-04	55	208
	11	89	2022-02-08 09:18:39	2022-03-01	2022-03-26	56	175
	12	87	2022-02-11 01:58:03	2022-02-27	2022-03-16	170	220

spacebooking13 x

12. Invoice

	Number	BillingDate	SpaceCost	ServiceCost	TotalBillingCost	SpaceBookingId
▶	1	2022-07-04	96.17	12.75	108.92	1
	2	2022-03-27	89.92	17.52	107.44	2
	3	2022-02-05	30.01	10.08	40.09	3
	4	2022-04-03	56.39	15.66	72.05	4
	5	2021-12-25	25.77	18.60	44.37	5
	6	2022-01-04	34.12	16.55	50.67	6
	7	2022-02-12	20.50	13.16	33.66	7
	8	2022-10-09	72.84	10.26	83.10	8
	9	2022-09-29	13.09	15.80	28.89	9
	10	2022-10-06	26.28	19.58	45.86	10
	11	2022-02-09	17.94	17.05	34.99	11
	12	2022-02-15	20.01	10.41	30.42	12

invoice 14 x

Queries:

1. Number of floors in buildings which are currently accepting bookings

```
SELECT b.Id, COUNT(DISTINCT bf.FloorNumber) num_floors
FROM building b
INNER JOIN building_floor bf ON bf.BuildingId = b.Id
WHERE b.Status = 'True'
GROUP BY b.Id
```

	Id	num_floors
▶	121	2
	169	2
	256	3
	285	3
	308	3
	350	2
	395	2
	399	4
	457	2
	503	4
	508	2
	511	2

Result 15 x

2. Total number of users and number of users segregated between Freelancers and CorporatePoc

```

SELECT COUNT(DISTINCT u.Id) Total_users, COUNT(DISTINCT f.UsersId)
freelancers,
COUNT(DISTINCT c.UsersId) corporate_poc
FROM users u
LEFT JOIN freelancer f ON f.UsersId = u.Id
LEFT JOIN corporatepoc c ON c.UsersId = u.Id

```

	Total_users	freelancers	corporate_poc
▶	300	250	50

3. Total number of bookings booked between 1st February and 1st July 2022 w.r.t SpaceTypes

```

SELECT s.SpaceType, COUNT(DISTINCT sb.Id) bookings
FROM spacebooking sb
INNER JOIN spaces s ON s.Id = sb.SpaceId
WHERE DATE(sb.bookingtime) >='2022-02-01' AND DATE(sb.bookingtime)
<'2022-07-01'
GROUP BY s.SpaceType

```

	SpaceType	bookings
▶	Common_Area	26
	Meeting_Room	31
	Seminar_Hall	3

4. Which user is generating maximum revenue for the Coworking Spaces ?

```

SELECT u.Id User_id, SUM(i.TotalBillingCost) total_cost
FROM invoice i
JOIN spacebooking sp ON sp.id = i.SpaceBookingId
JOIN users u ON u.Id = sp.UsersId
GROUP BY u.Id
ORDER BY total_cost DESC
LIMIT 1

```

	User_id	total_cost
▶	180	287.16

5. Which corporate has the maximum number of employees ?

```

SELECT cm.CorporateName
FROM corporate cm
WHERE cm.Id IN
(SELECT t.Id FROM
    (SELECT c.Id , count(DISTINCT e.Id) num_employees
    FROM corporate c
    JOIN employees e ON e.CorporateId = c.Id
    GROUP BY 1
    ORDER BY num_employees DESC
    LIMIT 1)
t)

```

	CorporateName
▶	Quinu