

פרוייקט גמר

יסודות התוכנה SQL

בית מלון

מגיש: רועי אפל

ת"ז: 329220511

כיתה: יג 13

מכללה: אורט רחובות

תאריך הגשה: 14.4.2025



2.....	עולם תוכן:
2.....	אתגרים:
3.....	מטרות:
3.....	משתמשים בבסיס הנתונים:
3.....	נתונים שיאוכסנו בבסיס הנתונים:
4.....	ישויות:
4.....	טבלת קשרים:
6.....	תרשים ERD
7.....	בסיס הנתונים
8.....	שאלות פעולות וטריגרים
8.....	בדיקת תפוסת חדרים לפי תאריכים
9.....	פירוט ההכנסות של המלון
10.....	מדד שביעות רצון האורחים
11.....	דוח שימוש בשירותים
12.....	מעקב אחרי עזיבות קרובות
13.....	עובדים רווחיים
14.....	היסטוריית הזמנות של אורחים
15.....	הזמנות לפי חודש
16.....	נתונים שימושיים על שירותי המלון
17.....	מחיקת ביקורות עם אורח - Trigger
20.....	קוד ליצירת הטבלות:
24.....	הכנסת נתונים לטבלות:

עולם תוכן:

עולם התוכן שלי הינו מערכת לניהול בית מלון. נשמר בה מידע על העובדי המלון, על השירותים שהמלון מספק ועל פרטי ההזמנות והאורחים במלון. בנוסף גם ניתן לשמור את הביקורות שניתנו על ידי האורחים במלון כדי לזהות מקומות לשיפור ושימור במלון.

אתגרים:

1. אין דרך קלה לעקוב אחר מצב זמינות החדרים (אילו פנויים ואילו תפוסים ובאילו תאריכים).
2. לא קל לזהות את העונות החזקות יותר והעונות החלשות ולדעת מתי לעשות מבצעים או לפרסם יותר.
3. לא תמיד ברור איזה שירותים האורחים אוהבים (שירותים פופולריים) ואיזה נתונים לשיפור.
4. אתגר בזיהוי אורחים קבועים ועידוד שלהם לחזור שוב.
5. בעיות בהתארגנות בזמן ובצורה יעילה לקראת עזיבת אורחים (ניקיון, הכנות).

מטרות:

1. קביעת מחירים נכונה לפי עונה וביקוש.
2. להבין אילו תחומים במלון הכי רווחיים ואיפה צריך להשתפר.
3. לשפר את השירותים ולהבין מה האורחים אוהבים.
4. להחליט אילו שירותים לשפר ואילו להפסיק להציע.
5. להתכונן לניקיון והכנת חדרים בזמן.
6. לזהות עובדים מצטיינים ולעודד מכירות נוספות.
7. להעניק הנחות לאורחים שחוזרים הרבה.
8. לזהות עונות חזקות וחלשות לצורך תכנון כוח אדם ושיווק.
9. להבין אילו שירותים הכי פופולריים במלון.

משתמשים בבסיס הנתונים:

מנהל: יש לו גישה להכל, הוא יכול להוסיף, לשנות ולמחוק כל דבר בבסיס הנתונים.

עובד: יכול להוסיף ולעדכן הזמנות ושירותים.

אורח: יכול לראות ולעדכן רק את המידע האישי שלו, לא יכול לראות או לשנות נתונים של אחרים. בנוסף, יכול להוסיף ביקורת ולמחוק את הביקורות שכתב בעבר.

נתונים שיאוכסנו בבסיס הנתונים:

מידע על אורחי המלון
מידע על עובדי המלון
מידע על סוגי החדרים השונים במלון
מידע על השירותים שהמלון מציע
פרטי הזמנות המלון
ביקורות על המלון ואת סוגי הביקורות השונים

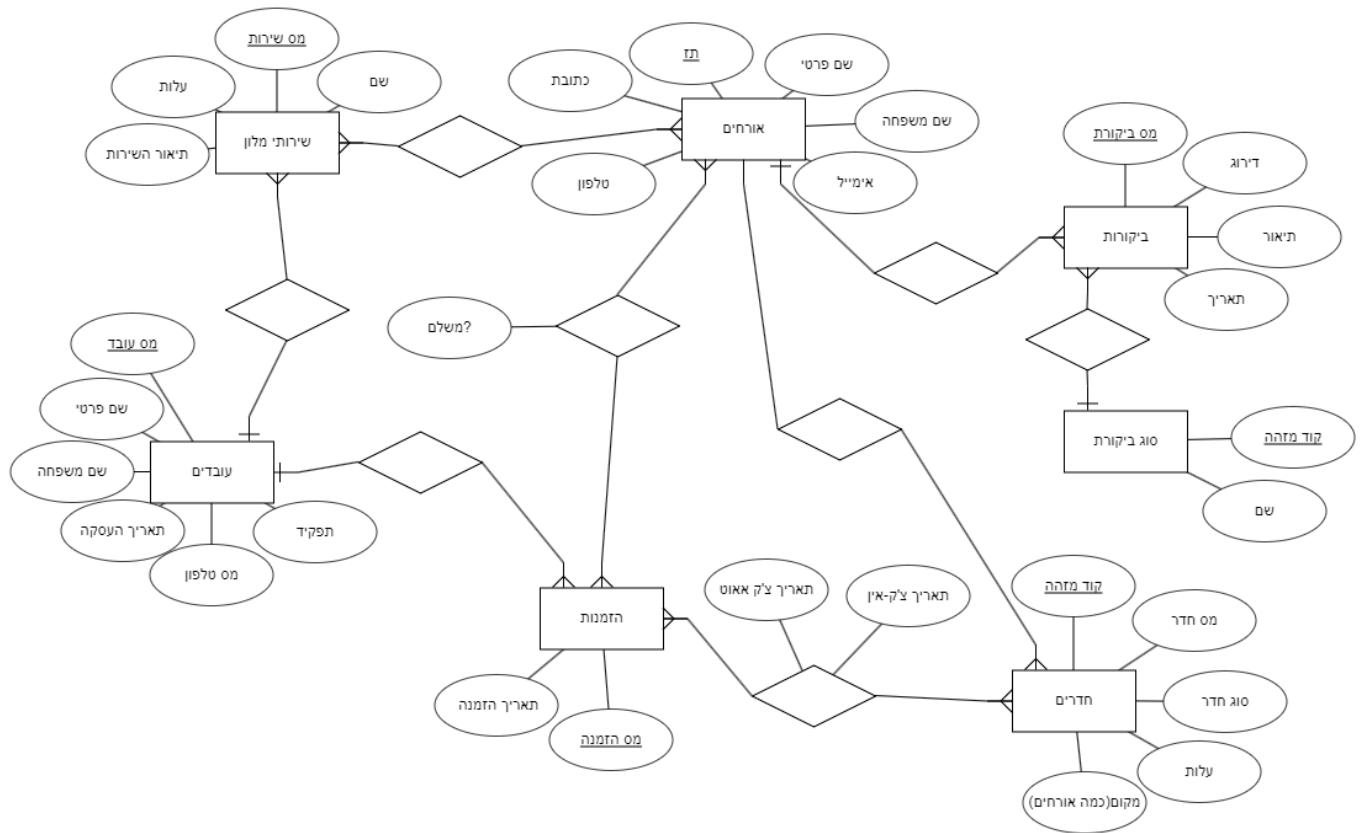
ישויות:

אורחים
הזמנות
ביקורות
סוגי ביקורות
עובדים
שירותי מלון
חדרים

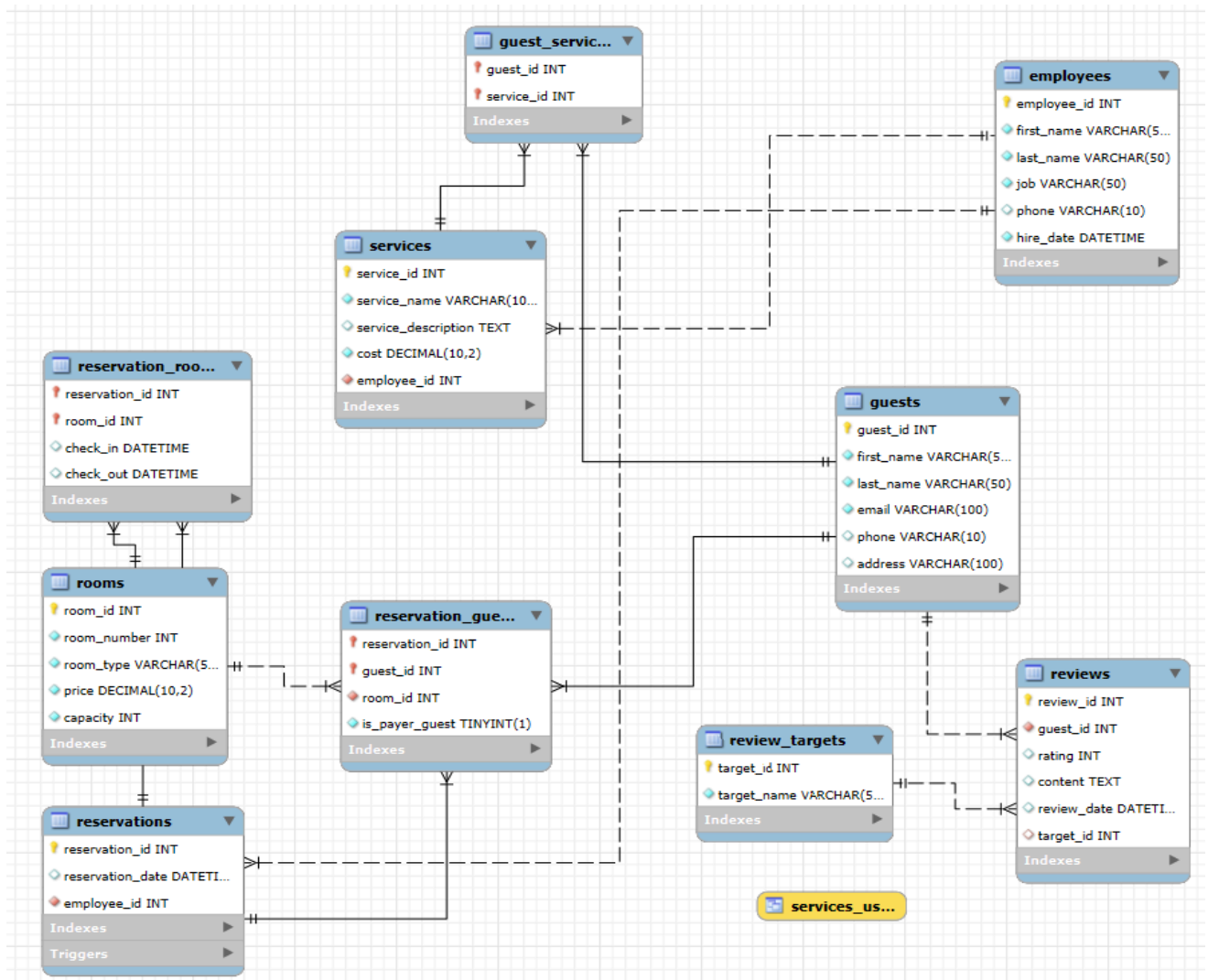
טבלת קשרים:

	אורחים	ביקורות	סוגי ביקורות	חדרים	הזמנות	עובדים	שירותי מלון
אורחים	x	יחיד לרבים	אין	אין	רבים לרבים	אין	רבים לרבים
ביקורות	x	x	יחיד לרבים	אין	אין	אין	אין
סוגי ביקורות	x	x	x	אין	אין	אין	אין
חדרים	x	x	x	x	רבים לרבים	אין	אין
הזמנות	x	x	x	x	x	יחיד לרבים	אין
עובדים	x	x	x	x	x	x	יחיד לרבים
שירותי מלון	x	x	x	x	x	x	x

ERD תרשים



בסיס הנתונים



שאלות פועולות וטריגרים

בדיקת תפוסת חדרים לפי תאריכים

- **רעיון:** פעולה שמסכמת כמה חדרים תפוסים בתאריך מסוים.
- **מטרה:** לדעת כמה חדרים תפוסים לעומת כמה פנויים בכל יום.
- **הסבר עסקי:** זה עוזר לקבוע מחירים טובים יותר לפי עונה וביקוש.

```
CREATE DEFINER='root'@'localhost' PROCEDURE `get_occupied_rooms`(in p_date date)
BEGIN

select r.room_id, r.room_number, r.room_type, r.price
from rooms r
inner join reservation_rooms rr on r.room_id = rr.room_id
where p_date >= date(rr.check_in) and p_date < date(rr.check_out);

END
```

```
#1
call get_occupied_rooms("2025-4-4");
```

room_id	room_number	room_type	price
1	101	Single	75.00
3	103	Suite	250.00
5	105	Double	130.00
8	108	Double	125.00

פירוט ההכנסות של המלון

רעיון: לשלב נתונים מהזמנות חדרים (לשלב **reservation_rooms** עם **rooms**) ומהשירותים הנוספים (עם **guest_services** ו-**services**) ולחשב את סך ההכנסות.

מטרה: להשיג בצורה נוחה פירוט של הכנסות החדרים ושל ההכנסות מהשירותים הנוספים.

הסבר עסקי: עוזר להשוות בין השירותים במלון מבחינת הרווח שלהם, איפה צריך לשפר ואיפה לשמר.

```
#2
select concat(room_type, " room type") as product, sum(price) as profit
from rooms as r
inner join reservation_rooms as rr on rr.room_id = r.room_id
group by room_type

union all

select service_name as product, sum(cost) as profit
from services as s
inner join guest_services as gs on s.service_id = gs.service_id
group by s.service_id
having profit > 0

union all

select "Total", get_total_profit() as profit;
```

```

CREATE DEFINER=`root`@`localhost` FUNCTION `get_total_profit`() RETURNS decimal(10,2)
    DETERMINISTIC
BEGIN

declare total_rooms_profit decimal(10,2);
declare total_services_profit decimal(10,2);

select sum(price) into total_rooms_profit
from rooms as r
inner join reservation_rooms as rr on r.room_id = rr.room_id;

select sum(cost) into total_services_profit
from services as s
inner join guest_services as gs on s.service_id = gs.service_id;

RETURN total_rooms_profit + total_services_profit;
END

```

product	profit
Single room type	315.00
Double room type	510.00
Suite room type	1065.00
Room Cleaning	50.00
Laundry	30.00
Breakfast	60.00
Spa	100.00
Airport Shuttle	80.00
Concierge	60.00
Extra Towels	10.00
Mini Bar	20.00
Room Service	70.00
Total	2370.00

מדד שביעות רצון האורחים

- **רעיון:** לחשב ממוצע ציונים ולספור את מספר הביקורות הטובות (יותר מ3) והפחות טובות (פחות מ4) מתוך טבלת **reviews**.
- **מטרה:** לדעת כמה האורחים מרוצים מזמנם במלון.
- **הסבר עסקי:** מאפשר לשפר שירותים ולהבין מה האורחים אוהבים ומה צריך שינוי.

```
#3
select floor(avg(rating)) as average_rating,
(select count(*) from reviews where rating > 3) as good_reviews,
(select count(*) from reviews where rating <= 3) as bad_reviews
from reviews;
```

average_rating	good_reviews	bad_reviews
4	9	3

דוח שימוש בשירותים

- רעיון: לספור כמה פעמים כל שירות נבחר.
- מטרה: לדעת איזה שירותים הכי מבוקשים ואילו פחות.
- הסבר עסקי: מאפשר להחליט אילו שירותים לשפר או אילו להפסיק להציע שירותים שפחות נחשבים.

*נעשה שימוש בview בשאלתה בהמשך

```
#4
create view services_uses as
select service_name, count(gs.service_id) as users_count
from services as s
inner join guest_services as gs on s.service_id = gs.service_id
group by gs.service_id
order by users_count;
```

service_name	users_count
WiFi Setup	1
Room Cleaning	2
Laundry	2
Spa	2
Airport Shuttle	2
Concierge	2
Extra Towels	2
Mini Bar	2
Room Service	2
Breakfast	3

מעקב אחרי עזיבות קרובות

- **רעיון:** להציג רשימת הזמנות שיש להן צ'ק-אאוט ב-24 השעות הקרובות.

- **מטרה:** להציג רשימת הזמנות שיש להן צ'ק-אאוט ב-24 השעות הקרובות.

- **הסבר עסקי:** להתכונן לניקיון, תחזוקה והכנת החדרים לאורחים הבאים. דואג שהחדרים מוכנים בזמן ושומר על רמת שירות גבוהה.

```
#5
select reservation_id, timestampdiff(hour, curdate(), check_out) as hours_til_check_out
from reservation_rooms
having hours_til_check_out between 0 and 24
order by hours_til_check_out;
```

reservation_id	hours_til_check_out
2	10

עובדים רווחיים

- **רעיון:** לבדוק כמה הכנסות הגיעו משירותים שכל עובד אחראי עליהם.
- **מטרה:** לזהות את העובדים שעשו את העבודה הכי טובה.
- **הסבר עסקי:** לתגמל עובדים טובים ולעודד מכירות נוספות.

```
#6
select concat(e.first_name, " ", e.last_name) as employee_name, e.job as job, sum(cost) as profit
from services as s inner join employees as e on e.employee_id = s.employee_id
group by s.employee_id
order by profit desc;
```

employee_name	job	profit
Patricia Allen	Concierge	70.00
Michael Clark	Receptionist	55.00
Samantha Lee	Manager	50.00
Linda Walker	Housekeeping	40.00
William Young	Maintenance	35.00
Robert Hall	Chef	15.00
Barbara Hernandez	Security	5.00

היסטוריית הזמנות של אורחים

- רעיון: למצוא את כמות ההזמנות שעשה כל אורח במשך חייו.
- מטרה: למצוא אורחים בעלי כמות ההזמנות משמעותית.
- הסבר עסקי: הענקת הנחות לאורחים שהזמינו הרבה כדי לעודד אותם לבוא שוב.

```
#7
select concat(g.first_name , " ", g.last_name) as guest_name,
count(distinct rg.reservation_id) as total_reservations
from guests as g
inner join reservation_guests as rg on g.guest_id = rg.guest_id
group by g.guest_id
order by total_reservations desc;
```

guest_name	total_reservations
John Doe	2
Jane Smith	2
Alice Johnson	1
Bob Williams	1
Carol Brown	1
David Jones	1
Eve Davis	1
Frank Miller	1
Grace Wilson	1
Henry Moore	1

הזמנות לפי חודש

- רעיון: לבדוק כמה הזמנות היו בכל חודש בשנה.
- מטרה: לזהות עונות חזקות וחלשות.
- הסבר עסקי: תכנון נכון של כוח אדם, שיווק והנחות עונתיות.

```
#8
select month(reservation_date) as month, count(month(reservation_date)) as total_reservations
from reservations
group by month(reservation_date);
```

month	total_reservations
4	12

נתונים שימושיים על שירותי המלון

- **רעיון:** לבדוק כמה שירותים בממוצע ובסך הכל השתמשו אורחים.
- **מטרה:** להבין אילו שירותים הכי פופולריים.
- **הסבר עסקי:** לשפר שירותים אהובים ולוותר על שירותים פחות טובים.

```
#9
select round(avg(users_count)) as average_services_uses,
max(users_count) as max_service_uses,
count(users_count) as total_uses
from services_uses;
```

average_services_uses	max_service_uses	total_uses
2	3	10

מחיקת ביקורות עם אורח - Trigger

- רעיון: למחוק אוטומטית ביקורות של אורח שמחקנו.
- מטרה: שלא יישארו ביקורות בלי אורח אמיתי.
- הסבר עסקי: לשמור על מערכת ביקורות נקייה ואמינה ללא תלות בבני אדם.

```
CREATE DEFINER='root'@'localhost' TRIGGER `guests_AFTER_DELETE` AFTER DELETE ON `guests` FOR EACH ROW BEGIN
    delete from reviews
    where guest_id = old.guest_id;
END
```

```
delete from guests
where guest_id = 1;
```

לפני הטריגר (טבלת הביקורות):

review_id	guest_id	rating	content	review_date	target_id
1	1	5	Excellent service and clean rooms.	2023-04-02 10:00:00	0
2	2	4	Great location and friendly staff.	2023-04-03 11:00:00	0
3	3	3	Room was average; could be better.	2023-04-04 12:00:00	1
4	4	5	Wonderful experience overall.	2023-04-05 13:00:00	0
5	5	4	Comfortable stay with minor issues.	2023-04-06 14:00:00	0
6	6	2	Not very clean, disappointed.	2023-04-07 15:00:00	1
7	7	5	Amazing service and cozy rooms.	2023-04-08 16:00:00	0
8	8	3	Okay stay, nothing exceptional.	2023-04-09 17:00:00	3
9	9	4	Very good, would come back again.	2023-04-10 18:00:00	0
10	10	5	Perfect experience and top-notch amenities.	2023-04-11 19:00:00	0
11	11	4	Nice but some issues with noise.	2023-04-12 20:00:00	2
12	12	5	Fantastic experience, highly recommended.	2023-04-13 21:00:00	0
NULL	NULL	NULL	NULL	NULL	NULL

לאחרי הטריגר (טבלת הביקורות):

review_id	guest_id	rating	content	review_date	target_id
2	2	4	Great location and friendly staff.	2023-04-03 11:00:00	0
3	3	3	Room was average; could be better.	2023-04-04 12:00:00	1
4	4	5	Wonderful experience overall.	2023-04-05 13:00:00	0
5	5	4	Comfortable stay with minor issues.	2023-04-06 14:00:00	0
6	6	2	Not very clean, disappointed.	2023-04-07 15:00:00	1
7	7	5	Amazing service and cozy rooms.	2023-04-08 16:00:00	0
8	8	3	Okay stay, nothing exceptional.	2023-04-09 17:00:00	3
9	9	4	Very good, would come back again.	2023-04-10 18:00:00	0
10	10	5	Perfect experience and top-notch amenities.	2023-04-11 19:00:00	0
11	11	4	Nice but some issues with noise.	2023-04-12 20:00:00	2
12	12	5	Fantastic experience, highly recommended.	2023-04-13 21:00:00	0
NULL	NULL	NULL	NULL	NULL	NULL

קוד ליצירת הטבלות:

```
CREATE TABLE guests (  
    guest_id INT AUTO_INCREMENT PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    email VARCHAR(100) UNIQUE NOT NULL,  
    phone VARCHAR(10),  
    address VARCHAR(100)  
);
```

```
CREATE TABLE employees (  
    employee_id INT AUTO_INCREMENT PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    job VARCHAR(50) NOT NULL,  
    phone VARCHAR(10),  
    hire_date DATETIME NOT NULL  
);
```

```
CREATE TABLE rooms (  
    room_id INT AUTO_INCREMENT PRIMARY KEY,  
    room_number INT UNIQUE NOT NULL,  
    room_type VARCHAR(50) NOT NULL,  
    price DECIMAL(10,2) NOT NULL,  
    capacity INT NOT NULL  
);
```

```
CREATE TABLE reservations (  
    reservation_id INT AUTO_INCREMENT PRIMARY  
KEY,  
    reservation_date DATETIME,  
    employee_id INT NOT NULL,  
    FOREIGN KEY (employee_id) REFERENCES  
employees(employee_id)  
);
```

```
CREATE TABLE services (  
    service_id INT AUTO_INCREMENT PRIMARY KEY,  
    service_name VARCHAR(100) NOT NULL,  
    service_description TEXT,  
    cost DECIMAL(10,2) NOT NULL,  
    employee_id INT NOT NULL,  
    FOREIGN KEY (employee_id) REFERENCES  
employees(employee_id)  
);
```

```
CREATE TABLE review_targets (  
    target_id INT PRIMARY KEY,  
    target_name VARCHAR(50) NOT NULL  
);
```

```
CREATE TABLE reviews (  
    review_id INT AUTO_INCREMENT PRIMARY KEY,  
    guest_id INT NOT NULL,  
    rating INT,  
    content TEXT,  
    review_date DATETIME,  
    target_id INT,  
    FOREIGN KEY (guest_id) REFERENCES  
guests(guest_id),  
    FOREIGN KEY (target_id) REFERENCES  
review_targets(target_id)  
);
```

```
CREATE TABLE reservation_guests (  
    reservation_id INT NOT NULL,  
    guest_id INT NOT NULL,  
    room_id INT NOT NULL,  
    is_payer_guest BOOLEAN NOT NULL,  
    PRIMARY KEY (reservation_id, guest_id),  
    FOREIGN KEY (reservation_id) REFERENCES  
reservations(reservation_id),  
    FOREIGN KEY (guest_id) REFERENCES  
guests(guest_id),  
    FOREIGN KEY (room_id) REFERENCES rooms(room_id)  
);
```

```
CREATE TABLE reservation_rooms (  
    reservation_id INT NOT NULL,  
    room_id INT NOT NULL,  
    check_in DATETIME,  
    check_out DATETIME,  
    PRIMARY KEY (reservation_id, room_id),  
    FOREIGN KEY (reservation_id) REFERENCES  
reservations(reservation_id),  
    FOREIGN KEY (room_id) REFERENCES rooms(room_id)  
);
```

```
CREATE TABLE guest_services (  
    guest_id INT NOT NULL,  
    service_id INT NOT NULL,  
    PRIMARY KEY (guest_id, service_id),  
    FOREIGN KEY (guest_id) REFERENCES  
guests(guest_id),  
    FOREIGN KEY (service_id) REFERENCES  
services(service_id)  
);
```

הכנסת נתונים לטבלות:

```
INSERT INTO guests (first_name, last_name, email,  
phone, address) VALUES  
( 'John', 'Doe', 'john.doe@example.com', '5551234567', '123  
Main St'),  
( 'Jane', 'Smith', 'jane.smith@example.com', '5552345678',  
'456 Oak Ave'),  
( 'Alice', 'Johnson', 'alice.johnson@example.com',  
'5553456789', '789 Pine Rd'),  
( 'Bob', 'Williams', 'bob.williams@example.com',  
'5554567890', '321 Maple Dr'),  
( 'Carol', 'Brown', 'carol.brown@example.com',  
'5555678901', '654 Cedar Ln'),  
( 'David', 'Jones', 'david.jones@example.com',  
'5556789012', '987 Birch Blvd'),  
( 'Eve', 'Davis', 'eve.davis@example.com', '5557890123',  
'159 Spruce St'),  
( 'Frank', 'Miller', 'frank.miller@example.com',  
'5558901234', '753 Elm St'),  
( 'Grace', 'Wilson', 'grace.wilson@example.com',  
'5559012345', '852 Walnut Ave'),  
( 'Henry', 'Moore', 'henry.moore@example.com',  
'5550123456', '951 Chestnut Rd'),  
( 'Ivy', 'Taylor', 'ivy.taylor@example.com', '5552233445',  
'204 Ash Dr'),  
( 'Jack', 'Anderson', 'jack.anderson@example.com',  
'5553344556', '301 Pine Rd'),
```


('Lily', 'King', 'lily.king@example.com', '5551122334', '456 Birch Ave'),
('Oscar', 'Martin', 'oscar.martin@example.com', '5552233446', '672 Elm St');

INSERT INTO employees (first_name, last_name, job, phone, hire_date) VALUES
('Samantha', 'Lee', 'Manager', '5551112222', '2022-01-10 09:00:00'),
('Michael', 'Clark', 'Receptionist', '5552223333', '2021-05-15 08:30:00'),
('Linda', 'Walker', 'Housekeeping', '5553334444', '2020-07-20 07:45:00'),
('Robert', 'Hall', 'Chef', '5554445555', '2019-03-12 10:00:00'),
('Patricia', 'Allen', 'Concierge', '5555556666', '2023-02-01 11:15:00'),
('William', 'Young', 'Maintenance', '5556667777', '2018-11-05 06:30:00'),
('Barbara', 'Hernandez', 'Security', '5557778888', '2020-09-17 08:00:00'),
('James', 'King', 'Receptionist', '5558889999', '2021-12-25 09:15:00'),
('Elizabeth', 'Wright', 'Manager', '5559990000', '2017-08-08 08:45:00'),
('Charles', 'Lopez', 'Housekeeping', '5550001111', '2022-06-30 07:00:00');

```
INSERT INTO rooms (room_number, room_type, price, capacity) VALUES
```

```
(101, 'Single', 75.00, 1),  
(102, 'Double', 120.00, 2),  
(103, 'Suite', 250.00, 4),  
(104, 'Single', 80.00, 1),  
(105, 'Double', 130.00, 2),  
(106, 'Suite', 260.00, 4),  
(107, 'Single', 70.00, 1),  
(108, 'Double', 125.00, 2),  
(109, 'Suite', 255.00, 4),  
(110, 'Double', 135.00, 2),  
(111, 'Single', 90.00, 1),  
(112, 'Suite', 300.00, 4);
```

```
INSERT INTO reservations (reservation_date, employee_id) VALUES
```

```
('2023-04-01 14:00:00', 1),  
( '2023-04-02 15:30:00', 2),  
( '2023-04-03 12:00:00', 3),  
( '2023-04-04 16:45:00', 4),  
( '2023-04-05 10:15:00', 5),  
( '2023-04-06 11:00:00', 6),  
( '2023-04-07 09:30:00', 7),  
( '2023-04-08 13:20:00', 8),  
( '2023-04-09 17:10:00', 9),  
( '2023-04-10 08:50:00', 10),  
( '2023-04-11 16:00:00', 1),  
( '2023-04-12 10:30:00', 2);
```

```
INSERT INTO services (service_name,  
service_description, cost, employee_id) VALUES  
( 'Room Cleaning', 'Daily room cleaning service', 25.00, 3),  
( 'Laundry', 'Clothes washing and ironing', 15.00, 3),  
( 'Breakfast', 'In-room breakfast service', 20.00, 2),  
( 'Spa', 'Relaxing spa treatments', 50.00, 1),  
( 'Airport Shuttle', 'Transport to/from the airport', 40.00,  
5),  
( 'Concierge', 'Personal assistance with bookings', 30.00,  
5),  
( 'Extra Towels', 'Provision of additional towels', 5.00, 7),  
( 'Mini Bar', 'Restocking mini bar items', 10.00, 6),  
( 'Room Service', 'In-room dining service', 35.00, 2),  
( 'WiFi Setup', 'High-speed internet access setup', 0.00, 8),  
( 'Fitness Center', 'Access to gym facilities', 25.00, 6),  
( 'Pool Access', 'Swimming pool usage', 15.00, 4);
```

```
INSERT INTO review_targets (target_id, target_name)  
VALUES  
(0, 'Good Review'),  
(1, 'Room'),  
(2, 'Service'),  
(3, 'Facilities');
```

```
INSERT INTO reviews (guest_id, rating, content,
review_date, target_id) VALUES
(1, 5, 'Excellent service and clean rooms.', '2023-04-02
10:00:00', 0),
(2, 4, 'Great location and friendly staff.', '2023-04-03
11:00:00', 0),
(3, 3, 'Room was average; could be better.', '2023-04-04
12:00:00', 1),
(4, 5, 'Wonderful experience overall.', '2023-04-05
13:00:00', 0),
(5, 4, 'Comfortable stay with minor issues.', '2023-04-06
14:00:00', 0),
(6, 2, 'Not very clean, disappointed.', '2023-04-07
15:00:00', 1),
(7, 5, 'Amazing service and cozy rooms.', '2023-04-08
16:00:00', 0),
(8, 3, 'Okay stay, nothing exceptional.', '2023-04-09
17:00:00', 3),
(9, 4, 'Very good, would come back again.', '2023-04-10
18:00:00', 0),
(10, 5, 'Perfect experience and top-notch amenities.',
'2023-04-11 19:00:00', 0),
(11, 4, 'Nice but some issues with noise.', '2023-04-12
20:00:00', 2),
(12, 5, 'Fantastic experience, highly recommended.',
'2023-04-13 21:00:00', 0);
```

```
INSERT INTO reservation_guests (reservation_id,
guest_id, room_id, is_payer_guest) VALUES
(1, 1, 1, TRUE),
(2, 2, 2, TRUE),
(3, 3, 3, TRUE),
(4, 4, 4, TRUE),
(5, 5, 5, TRUE),
(6, 6, 6, TRUE),
(7, 7, 7, TRUE),
(8, 8, 8, TRUE),
(9, 9, 9, TRUE),
(10, 10, 10, TRUE),
(11, 1, 11, TRUE),
(12, 2, 12, TRUE);
```

```
INSERT INTO reservation_rooms (reservation_id,
room_id, check_in, check_out) VALUES
(1, 101, '2025-04-01 14:00:00', '2025-04-05 11:00:00'),
(2, 102, '2025-04-02 15:30:00', '2025-04-06 10:00:00'),
(3, 103, '2025-04-03 12:00:00', '2025-04-07 12:00:00'),
(4, 104, '2025-04-04 16:45:00', '2025-04-08 09:00:00'),
(5, 105, '2025-04-05 10:15:00', '2025-04-09 10:00:00'),
(6, 106, '2025-04-06 11:00:00', '2025-04-10 11:00:00'),
(7, 107, '2025-04-07 09:30:00', '2025-04-11 12:00:00'),
(8, 108, '2025-04-08 13:20:00', '2025-04-12 10:30:00'),
(9, 109, '2025-04-09 17:10:00', '2025-04-13 09:45:00'),
(10, 110, '2025-04-10 08:50:00', '2025-04-14 10:15:00'),
(11, 111, '2025-04-07 10:00:00', '2025-04-07 18:00:00'),
(12, 112, '2025-04-07 15:00:00', '2025-04-08 09:00:00');
```

```
INSERT INTO guest_services (guest_id, service_id)
VALUES
(1, 1),
(2, 5),
(3, 6),
(4, 2),
(5, 3),
(6, 7),
(7, 4),
(8, 8),
(9, 9),
(10, 1),
(1, 3),
(2, 8),
(3, 9),
(4, 10),
(5, 4),
(6, 2),
(7, 5),
(8, 3),
(9, 6),
(10, 7);
```

