

1. CREATE CUSTOMERS TABLE

```
CREATE TABLE customers (  
    id NUMBER PRIMARY KEY,  
    customer_id VARCHAR2(20),  
    customer_name VARCHAR2(50),  
    customer_acc_no VARCHAR2(20),  
    password VARCHAR2(50),  
    Date_of_birth DATE,  
    gender VARCHAR2(10),  
    State VARCHAR2(50),  
    City VARCHAR2(50),  
    Zipcode VARCHAR2(10),  
    Street_no VARCHAR2(20),  
    House_no VARCHAR2(20),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE  
);
```

2. CREATE CUSTOMER CONTACT INFO TABLE

```
CREATE TABLE customer_contact_info (  
    id NUMBER PRIMARY KEY,  
    contact_no VARCHAR2(20),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE  
);
```

3. ADD CUSTOMER ID AS A FOREIGN KEY AT CUSTOMER CONTACT INFO TABLE FROM CUSTOMERS TABLE

```
ALTER TABLE customer_contact_info ADD CONSTRAINT fk_customer_id  
FOREIGN KEY (customer_id)  
REFERENCES customers(id);
```

4. CREATE OWNERS TABLE.

```
CREATE TABLE owners (  
    owner_id NUMBER PRIMARY KEY,  
    name VARCHAR2(50),  
    password VARCHAR2(50),  
    Date_of_birth DATE,  
    State VARCHAR2(50),  
    City VARCHAR2(50),  
    Street_no VARCHAR2(20),  
    House_no VARCHAR2(20),  
    Zip_code VARCHAR2(10),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE  
);
```

5. CREATE OWNER CONTACT INFO TABLE.

```
CREATE TABLE owner_contact_info (  
    id NUMBER PRIMARY KEY,  
    owner_id NUMBER,  
    contact_no VARCHAR2(20),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE  
);
```

6. ADD OWNER ID AS A FOREIGN KEY AT OWNER CONTACT INFO TABLE FROM OWNERS TABLE

```
ALTER TABLE owner_contact_info ADD CONSTRAINT fk_owner_id  
    FOREIGN KEY (owner_id)  
    REFERENCES owners(owner_id);
```

7. CREATE OFFICE SPACE TABLE.

```
CREATE TABLE office_space (  
    office_id NUMBER PRIMARY KEY,  
    owners_id NUMBER,  
    floor_no NUMBER,  
    total_cost NUMBER,  
    space_name VARCHAR2(50),  
    recreational_corner VARCHAR2(10),  
    size_of_room VARCHAR(20),  
    parking_space VARCHAR2(10),  
    area_in_sqmtr NUMBER,  
    cost_of_each_room NUMBER,  
    type_of_rooms VARCHAR2(20),  
    shared_space VARCHAR2(20),  
    shared_gender VARCHAR2(10),  
    numbers_of_shares NUMBER,  
    state VARCHAR2(50),  
    city VARCHAR2(50),
```

```
street_no VARCHAR2(20),
house_no VARCHAR2(20),
zipcode VARCHAR2(10),
monthly_availability DATE,
status VARCHAR2(20),
created_at TIMESTAMP DEFAULT SYSDATE,
updated_at TIMESTAMP DEFAULT SYSDATE,
FOREIGN KEY (owners_id) REFERENCES owners(owner_id)
);
```

8. CREATE LIVING SPACE TABLE.

```
CREATE TABLE living_space (
  House_no NUMBER PRIMARY KEY,
  owner_id NUMBER,
  space_name VARCHAR2(50),
  Full_Furnished_room VARCHAR2(20),
  Total_floor NUMBER,
  Area_in_sqmtr NUMBER,
  cost NUMBER,
  number_of_rooms NUMBER,
  shared_space VARCHAR2(20),
  shared_gender VARCHAR2(10),
  numbers_of_shares NUMBER,
  state VARCHAR2(50),
  city VARCHAR2(50),
  zipcode VARCHAR2(10),
  street_no VARCHAR2(20),
  monthly_availability DATE,
  status VARCHAR2(20),
  created_at TIMESTAMP DEFAULT SYSDATE,
  updated_at TIMESTAMP DEFAULT SYSDATE,
  FOREIGN KEY (owner_id) REFERENCES owners(owner_id)
);
```

9. CREATE OFFICE SPACE IMAGES TABLE.

```
CREATE TABLE office_space_images (  
    id NUMBER PRIMARY KEY,  
    officeid NUMBER,  
    images VARCHAR2(100),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE,  
    FOREIGN KEY (officeid) REFERENCES office_space(office_id)  
);
```

10. CREATE PAYMENT INFO TABLE.

```
CREATE TABLE payment_info (  
    payment_id NUMBER PRIMARY KEY,  
    customer_acc_no VARCHAR2(20),  
    customer_id NUMBER,  
    Amount NUMBER,  
    payment_date DATE,  
    payment_mode VARCHAR2(20),  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE,  
    FOREIGN KEY (customer_id) REFERENCES customers(id)  
);
```

11. CREATE OFFICE BOOKING TABLE.

```
CREATE TABLE office_booking (  
    booking_id NUMBER PRIMARY KEY,  
    space_id NUMBER,  
    booking_type VARCHAR2(20),  
    customers_id NUMBER,  
    start_date DATE,  
    end_date DATE,  
    status VARCHAR2(20),  
    created_at TIMESTAMP DEFAULT SYSDATE,  
    updated_at TIMESTAMP DEFAULT SYSDATE,  
    FOREIGN KEY (space_id) REFERENCES office_space(office_id),  
    FOREIGN KEY (customers_id) REFERENCES customers(id)  
);
```

12. CREATE LIVING BOOKING TABLE.

```
CREATE TABLE living_booking (  
    booking_id NUMBER PRIMARY KEY,  
    spaceid NUMBER,  
    booking_type VARCHAR2(20),  
    customers NUMBER,  
    start_date DATE,  
    end_date DATE,  
    status VARCHAR2(20),
```

```

created_at TIMESTAMP DEFAULT SYSDATE,
updated_at TIMESTAMP DEFAULT SYSDATE,
FOREIGN KEY (spaceid) REFERENCES living_space (House_no),
FOREIGN KEY (customers) REFERENCES customers(id)
);

```

13. INSERT INTO CUSTOMERS

```

INSERT INTO customers (id,customer_name, customer_acc_no, password,
Date_of_birth, gender, State, City, Zipcode, Street_no, House_no, status) VALUES
(10,'Grace Zhang', '0123456789', 'passwordqrs', TO_DATE('1980-12-05', 'YYYY-MM-
DD'), 'Female', 'Washington', 'Seattle', '98101', '890 Pike St', 'Unit 201', 'Inactive');

```

14. SEE FROM CUSTOMERS TABLE

```

SELECT * FROM customers

```

15. SEE ALL FROM CUSTOMER CONTACT INFO TABLE

```

SELECT * FROM customer_contact_info;

```

16. PRINT ALL OWNERS WITH TABLE EXAMPLE

```

select * from owners;

```

OWNER_ID NAME	PASSWORD	DATE_OF_B STATE
1 John Smith	password123	12-MAY-90 California
2 Jane Doe	abc123	22-OCT-85 Texas
3 Bob Johnson	pass456	28-FEB-78 New York
4 Sarah Lee	password123	15-NOV-95 Florida

17. SEE ALL OWNERS CONTACT INFORMATION

```
SELECT * FROM owner_contact_info;
```

ID	OWNER_ID	CONTACT_NO	STATUS	CREATED_AT	UPDATED_AT
1	1	123-456-7890	active	30-APR-23 06.45.44.000000000 PM	30-APR-23 06.45.44.000000000 PM
2	2	234-567-8901	active	30-APR-23 06.45.58.000000000 PM	30-APR-23 06.45.58.000000000 PM
3	3	345-678-9012	inactive	30-APR-23 06.46.12.000000000 PM	30-APR-23 06.46.12.000000000 PM
4	4	456-789-0123	active	30-APR-23 06.46.26.000000000 PM	30-APR-23 06.46.26.000000000 PM

18. ADD DATA INTO OFFICE SPACE.

```
INSERT INTO office_space
```

(office_id, owners_id, floor_no, total_cost, space_name, recreational_corner, size_of_room, parking_space, area_in_sqmtr, cost_of_each_room, type_of_rooms, shared_space, shared_gender, numbers_of_shares, state, city, street_no, house_no, zipcode, monthly_availability, status, created_at, updated_at)

```
VALUES
```

(4,2,5,10000, 'Office 1', 'Yes', 'Large', 'Yes', 200, 5000, 'Private', 'No', 'Yes', 5, 'Delhi', 'delhi', '123 Main Street', 'Ghm 100', '144101', TO_DATE('2023-12-05', 'YYYY-MM-DD'),'Active' , SYSDATE, SYSDATE);

19. SEE INFO ALL FROM OFFICE SPACE

```
SELECT * FROM office_space;
```

OFFICE_ID	OWNERS_ID	FLOOR_NO	TOTAL_COST	SPACE_NAME	RECREATION	SIZE_OF_ROOM	PARKING_SP	AREA_IN_SQMTR	COS
1	1	5	10000	Office 1	Yes	Large	Yes	200	
2	3	5	10000	Office 1	Yes	Large	Yes	200	
3	2	5	10000	Office 1	Yes	Large	Yes	200	
4	2	5	10000	Office 1	Yes	Large	Yes	200	

20. INSERT DATA INTO OFFICE SPACE IMAGES

```
INSERT INTO office_space_images (id, officeid, images, status, created_at, updated_at)

VALUES
```

```
(10, 2, 'https://example.com/image10.jpg', 'active', SYSDATE, SYSDATE);
```

21. FETCH ALL IMAGES FROM OFFICE SPACE IMAGE.

```
SELECT * FROM office_space_image;
```

ID	OFFICEID	IMAGES	STATUS	CREATED_AT	UPDATED_AT
1	1	https://example.com/image1.jpg	active	30-APR-23 08.27.44.000000000 PM	30-APR-23 08.27.44.000000000 PM
2	1	https://example.com/image2.jpg	active	30-APR-23 08.28.36.000000000 PM	30-APR-23 08.28.36.000000000 PM
3	2	https://example.com/image3.jpg	active	30-APR-23 08.28.50.000000000 PM	30-APR-23 08.28.50.000000000 PM
4	2	https://example.com/image4.jpg	active	30-APR-23 08.29.13.000000000 PM	30-APR-23 08.29.13.000000000 PM
5	3	https://example.com/image5.jpg	active	30-APR-23 08.29.42.000000000 PM	30-APR-23 08.29.42.000000000 PM
6	3	https://example.com/image6.jpg	active	30-APR-23 08.29.58.000000000 PM	30-APR-23 08.29.58.000000000 PM
7	4	https://example.com/image7.jpg	active	30-APR-23 08.30.09.000000000 PM	30-APR-23 08.30.09.000000000 PM
8	4	https://example.com/image8.jpg	active	30-APR-23 08.30.22.000000000 PM	30-APR-23 08.30.22.000000000 PM
9	3	https://example.com/image9.jpg	active	30-APR-23 08.30.36.000000000 PM	30-APR-23 08.30.36.000000000 PM
10	2	https://example.com/image10.jpg	active	30-APR-23 08.30.42.000000000 PM	30-APR-23 08.30.42.000000000 PM

22. COMBINE OFFICE SPACE TABLE WITH IMAGES FROM OFFICE SPACE IMAGES TABLE

```
SELECT o.floor_no, o.total_cost, o.space_name, i.images

FROM office_space o
JOIN office_space_images i
ON o.office_id = i.officeid;
```

FLOOR_NO	TOTAL_COST	SPACE_NAME	IMAGES
5	10000	Office 1	https://example.com/image1.jpg
5	10000	Office 1	https://example.com/image2.jpg
5	10000	Office 1	https://example.com/image3.jpg
5	10000	Office 1	https://example.com/image4.jpg
5	10000	Office 1	https://example.com/image5.jpg
5	10000	Office 1	https://example.com/image6.jpg
5	10000	Office 1	https://example.com/image7.jpg
5	10000	Office 1	https://example.com/image8.jpg
5	10000	Office 1	https://example.com/image9.jpg
5	10000	Office 1	https://example.com/image10.jpg

10 rows selected.

23. INSERT DATA INTO LIVING SPACE TABLE.

```
INSERT INTO living_space (House_no, owner_id, space_name, Full_Furnished_room,
Total_floor, Area_in_sqmtr, cost, number_of_rooms, shared_space, shared_gender,
numbers_of_shares, state, city, zipcode, street_no, monthly_availability, status,
created_at, updated_at)
VALUES
('123', 1, 'Apartment 1', 'Yes', 5, 100, 50000, 2, 'Yes', 'Female', 2, 'Haryana', 'Ambala',
'10001', '123 Main St', TO_DATE('2023-12-05', 'YYYY-MM-DD'), 'Active', SYSDATE,
SYSDATE);
```

24. SELECT ALL ROWS FROM LIVING SPACE TABLE

```
SELECT * FROM living_space;
```

HOUSE_NO	OWNER_ID	SPACE_NAME	FULL_FURNISHED_ROOM	TOTAL_FLOOR	AREA_IN_SQMTR	COST	NUMBER_OF_ROOMS	SHARED_SPACE	SHARED_GEN	NUMBERS_OF_SHARES	STA
123	1	Apartment 1	Yes	5	100	50000	2	Yes	Female	2	Har
124	3	Apartment 2	Yes	5	100	90000	2	Yes	Both	2	Fun

25. INSERT INTO PAYMENT INFO

```
INSERT INTO payment_info (payment_id, customer_acc_no, customer_id, Amount,
payment_date, payment_mode, status, created_at, updated_at)
VALUES
(1, '1234567890', 1, 10000, TO_DATE('2023-01-05', 'YYYY-MM-DD'), 'Credit Card',
'Paid', SYSDATE, SYSDATE),
```

26. SELECT FROM PAYMENT INFO

```
SELECT * FROM payment_info
```

PAYMENT_ID	CUSTOMER_ACC_NO	CUSTOMER_ID	AMOUNT	PAYMENT_D	PAYMENT_MODE	STATUS	CREATED_AT	UPDATED_AT
1	1234567890	1	10000	05-JAN-23	Credit Card	Paid	30-APR-23 09.06.31.0000000000 PM	30-APR-23 09.06.31.0000000000 PM
2	0987654321	2	20000	05-APR-23	Debit Card	Paid	30-APR-23 09.06.54.0000000000 PM	30-APR-23 09.06.54.0000000000 PM
3	5555555555	3	15000	05-MAR-23	PayPal	Paid	30-APR-23 09.07.21.0000000000 PM	30-APR-23 09.07.21.0000000000 PM
5	1111111111	5	50000	05-DEC-22	Credit Card	Paid	30-APR-23 09.08.23.0000000000 PM	30-APR-23 09.08.23.0000000000 PM
4	9999999999	4	75000	05-FEB-23	Cash	Pending	30-APR-23 10.50.01.0000000000 PM	30-APR-23 10.50.01.0000000000 PM

27. INSERT DATA INTO OFFICE BOOKING

```
INSERT INTO office_booking (booking_id, space_id, booking_type, customers_id,
start_date, end_date, status, created_at, updated_at)
VALUES (1, 1, 'office', 1, TO_DATE('2023-02-10', 'YYYY-MM-DD'), TO_DATE('2024-
01-01', 'YYYY-MM-DD'), 'cancelled', SYSDATE, SYSDATE);
```

28. FETCH DATA FROM OFFICE BOOKING

```
SELECT * office_booking;
```

BOOKING_ID	SPACE_ID	BOOKING_TYPE	CUSTOMERS_ID	START_DAT	END_DATE	STATUS	CREATED_AT	UPDATED_AT
1	1	office	1	10-FEB-23	01-JAN-24	cancelled	30-APR-23 11.46.54.000000000 PM	30-APR-23 11.46.54.000000000 PM
2	1	office	3	10-FEB-23	01-JAN-24	cancelled	30-APR-23 11.47.35.000000000 PM	30-APR-23 11.47.35.000000000 PM
3	3	office	4	12-MAY-23	27-DEC-25	confirmed	30-APR-23 11.48.02.000000000 PM	30-APR-23 11.48.02.000000000 PM
4	4	office	4	24-FEB-23	15-DEC-24	confirmed	30-APR-23 11.48.14.000000000 PM	30-APR-23 11.48.14.000000000 PM

29. INSERT DATA INTO LIVING BOOKING

```
INSERT INTO living_booking (booking_id, spaceid, booking_type, customers,
start_date, end_date, status, created_at, updated_at)
VALUES (1, 123, 'living', 5, TO_DATE('2023-01-05', 'YYYY-MM-DD'),
TO_DATE('2023-12-05', 'YYYY-MM-DD'), 'confirmed',SYSDATE, SYSDATE);
```

30. FETCH DATA FROM LIVING BOOKING

```
SELECT * living_booking;
```

BOOKING_ID	SPACEID	BOOKING_TYPE	CUSTOMERS	START_DAT	END_DATE	STATUS	CREATED_AT	UPDATED_AT
1	123	living	5	05-JAN-23	05-DEC-23	confirmed	01-MAY-23 12.10.06.000000000 AM	01-MAY-23 12.10.06.000000000 AM
2	124	living	7	01-JAN-23	01-DEC-23	pending	01-MAY-23 12.11.00.000000000 AM	01-MAY-23 12.11.00.000000000 AM