



King Saud University
College of Computer and Information Sciences
Department of Information Technology

IT222: Database Principles

1st Semester 1446 H

Le MERIDIEN

Le Méridien Hotel

Phase # 3

Section #	NAME	ID
View Name: Customer view		
77761	Alanoud Khalid Aloraydi	444201111
77761	Sarah alruwayte	444200758
77761	Walaa Saif Aleslam mohammed	444200088
77761	Jana Abdullah Alsuwailem	444200816
77761	Atheer Ibrahim Bin Badie	444200894

Supervised By: Dr. Noof Alfear.

Project Description:

This project is designing a database for a hotel chain with locations worldwide, offering a variety of services such as pools, spas, gyms, and more. Our primary clients are the guests who stay at these hotels, seeking a comfortable and enjoyable experience. Our database is designed to simplify the management of hotel information by centralizing data, thereby improving overall efficiency. It will empower staff to manage bookings, services, and customer information more effectively, ensuring a seamless experience for both employees and guests.

View Description:

The database view is designed to assist hotel guests (customers) in managing their interactions with the hotel. Customers can create accounts, make bookings, and manage their reservations. Each booking is linked to a branch, room details, and additional services.

The relationship between these elements ensures a smooth and integrated experience for customers, making the booking process efficient and user-friendly.

Data Requirements:

Customer: The person who will make the booking. The customer has a unique customerID, first and last name, email, and phone number. Each customer can make one or many bookings.

Booking: The customer's recorded reservation management. It includes a unique bookingNo, each booking has bookingStatus, paymentInfo, checkInDate, numOfNights, and totalCost. Each booking has one or more rooms and can contain many services, one or no service.

Branch: The branch of the hotel where all rooms belong to. is identified by branchID and has branchCity since the hotel has branches in different cities, each branch has many rooms.

Room: The room included in booking and identified by roomNo, and has attributes roomType, roomCost, bedType, floorNo, and roomView. Each room can be in more than one booking at different date. No one branch has two roomNo that are the same although two or more branches have the same roomNo. Room belongs to one branch and a branch has many rooms.

Services: Any service, whether paid or free, that the hotel offers to its customers, such as spa treatments, gym access, meal delivery, housekeeping and other services. It includes a unique serviceID, serviceName, serviceCost. Each service belongs to zero or many booking.

Transaction Requirements:

Data Entry:

- 1- Add New Customer
- 2- Enter the number of nights for booking

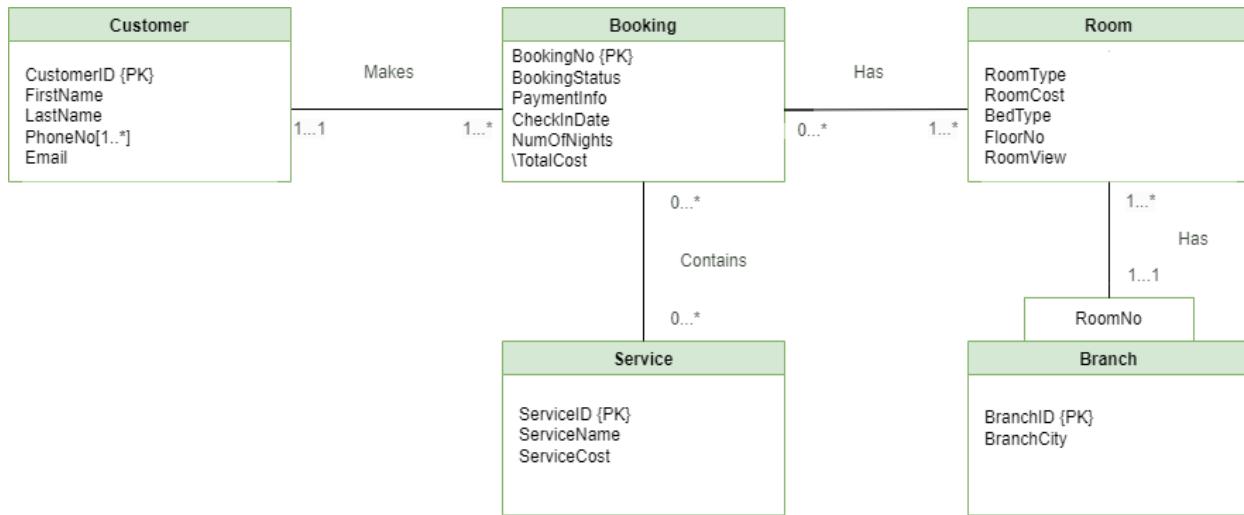
Data update/deletion:

- 1- Delete Reservation
- 2- Update Customer's Phone Number and Email
- 3- Update Payment info for a Booking

Data Queries:

- 1- Show the list of all branches
- 2- Show the list of the customer's current and previous reservations
- 3- Show the list of services priced under 400
- 4- Show the list of rooms and suites sorted from lowest to highest price at the Al Khobar branch
- 5- Show the list of rooms with a view at the Jeddah branch
- 6- Show the list of Single bed rooms at the Riyadh branch
- 7- List of free services
- 8- Show the list of suites at the Al Khobar branch priced between 2000 and 3000
- 9- Show the list of rooms at the Jeddah branch located on the first floor
- 10- Calculate the total price of all services requested by the customer

Global enhanced entity relationship diagram (EER):



Relational Schema:

Customer (customerID, firstName, lastName, email)

Primary key: customerID

Telephone (phoneNo, customerID)

Primary key: phoneNo, customerID

Foreign key: customerID references Customer (customerID)

Booking (bookingNo, bookingStatus, paymentInfo, checkInDate, numOfNights, totalCost, customerID)

Primary key: bookingNo

Foreign key: customerID references Customer (customerID)

Branch (branchID, branchCity)

Primary key: branchID

Room (roomNo, branchID, roomType, roomCost, bedType, floorNo, roomView)

Primary key: roomNo, branchID

Foreign key: branchID references Branch (branchID)

Service (serviceID, serviceName, serviceCost)

Primary key: serviceID

Has (bookingNo, roomNo, branchID)

Primary key: bookingNo, roomNo, branchID

Foreign key: bookingNo references Booking (bookingNo)

Foreign key: roomNo references Room (roomNo)

Foreign key: branchID references Branch (branchID)

Contains (bookingNo, serviceID)

Primary key: bookingNo, serviceID

Foreign key: bookingNo references Booking (bookingNo)

Foreign key: serviceID references Service (serviceID)

Data Dictionary showing description of all entities:

Entity Name	Description	Occurrence
Customer	General term describing all The person who will make the booking.	Each customer can make one or many bookings.
Booking	General term describing all The Customer Is Recorded Reserve Management.	Each booking has one or more rooms and can contain many services, one or no service.
Room	General term describing all the room included in booking	Each room can be in more than one booking at different date. No one branch has two roomNo that are the same although two or more branches have the same roomNo. Room belongs to one branch and a branch has many rooms.
Service	General term describing all services that the hotel offers to its customers.	Each service belongs to zero or many booking.
Branch	General term describing all the branch of the hotel where all rooms belong to.	Each branch has many rooms.

Data Dictionary showing description of all relationships:

Entity Name	Multiplicity	Relationship	Entity Name	Multiplicity
Branch	1..1	Has	Room	1..*
Booking	0..*	Has	Room	1..*
Booking	0..*	Contains	Service	0..*
Customer	1..1	Makes	Booking	1..*

Data Dictionary showing description of all attributes:

Entity Name	Attribute	Description	Data Type	Length	Nulls	Multi-Valued	Default Value	Range	PK
Customer	CustomerID	Unique identifier number assigned to each customer stays at the hotel	number	10					Y
	FirstName	Frist name of customer	varchar	10					
	LastName	Last name of customer	varchar	10					
	PhoneNo	customer contact numbers	char	10		Y			
	Email	The customer email	varchar	64					
Booking	BookingNo	The unique number of the booking	number	10					Y

[IT222-1445(2): Le Méridien Hotel]

	BookingStatus	Booking status if confirmed or not	varchar	10				Confirm/ Canceled	
	PaymentInfo	Information of the customer payment	varchar	10					
	CheckInDate	Booking start date	date	10					
	NumOfNights	Number of customer sleep nights	integer						
	TotalCost	Total cost of the booking	decimal	8.2	Y				
Room	RoomType	The room's type	varchar	10					
	RoomCost	The room's price	decimal	6.2					
	BedType	Type of beds in the room	varchar	10	Y		Queen bed		
	FloorNo	The floor's number	varchar	2					
	RoomView	whether the room has a view or not.	varchar	10	Y		No view	No view/ with view	
	RoomNo	The room's number	varchar	3					Y
Branch	BranchID	The unique number of the Branch	char	5					Y
	BranchCity	City where the hotel is located	varchar	10					
Service	ServiceID	unique service ID	char	5					Y
	ServiceName	The name of the service provided	varchar	10					
	ServiceCost	The service's price	decimal	6.2					

[IT222-1445(2): Le Méridien Hotel]

Table Booking

DB tables creation:

BOOKING							
Columns	Data	Indexes	Constraints	Grants	Statistics	Triggers	Dependencies
Column Name	Data Type	Nullable	Default		Primary Key	Comment	Identity
BOOKING_STATUS	VARCHAR2(10 CHAR)	N					
PAYMENT_INFO	VARCHAR2(10 CHAR)	N					
CHECKIN_DATE	DATE	N					
NUM_OF_NIGHTS	NUMBER	N					
TOTAL_COST	NUMBER(8,2)	Y					
CUSTOMER_ID	NUMBER(10,0)	N					
BOOKING_NO	NUMBER	N					

BOOKING

```

CREATE TABLE "BOOKING"
(
    "BOOKING_STATUS" VARCHAR2(10 CHAR) NOT NULL ENABLE,
    "PAYMENT_INFO" VARCHAR2(10 CHAR) NOT NULL ENABLE,
    "CHECKIN_DATE" DATE NOT NULL ENABLE,
    "NUM_OF_NIGHTS" NUMBER NOT NULL ENABLE,
    "TOTAL_COST" NUMBER(8,2),
    "CUSTOMER_ID" NUMBER(10,0) NOT NULL ENABLE,
    "BOOKING_NO" NUMBER(10,0) NOT NULL ENABLE,
    CONSTRAINT "BOOKING_STATUS_CHK" CHECK ("BOOKING_STATUS" IN ('confirmed', 'canceled')) ENABLE,
    CONSTRAINT "BOOKING_PK" PRIMARY KEY ("BOOKING_NO")
    USING INDEX ENABLE
);

ALTER TABLE "BOOKING" ADD CONSTRAINT "FK_BOOKING_CUSTOMER_ID" FOREIGN KEY ("CUSTOMER_ID")
    REFERENCES "CUSTOMER" ("CUSTOMER_ID") ENABLE;

CREATE OR REPLACE EDITIONABLE TRIGGER "CALCULATE_TOTALCOST"
BEFORE INSERT OR UPDATE ON BOOKING
FOR EACH ROW
DECLARE
    TOTAL_ROOM_COST NUMBER := 0;
    TOTAL_ROOM_COST NUMBER := 0;
BEGIN
    FOR r IN (SELECT r.ROOM_COST
              FROM ROOM r
              JOIN MAS m ON r.ROOM_NO = m.ROOM_NO
              WHERE m.BOOKING_NO = :NEW.BOOKING_NO) LOOP
        TOTAL_ROOM_COST := TOTAL_ROOM_COST + r.ROOM_COST;
    END LOOP;

    IF TOTAL_ROOM_COST > 0 THEN
        :NEW.TOTAL_COST := TOTAL_ROOM_COST * :NEW.NUM_OF_NIGHTS;
    ELSE
        :NEW.TOTAL_COST := 0;
    END IF;
    :NEW.BOOKING_NO := :NEW.BOOKING_NO;
END;
/
ALTER TRIGGER "CALCULATE_TOTALCOST" ENABLE;

```

Data insertion:

BOOKING							
Columns	Data	Indexes	Constraints	Grants	Statistics	Triggers	Dependencies
+ Insert Row	Columns...	Filter...	Count Rows	Load Data	Download	Refresh	
	BOOKING_STATUS	PAYMENT_INFO	CHECKIN_DATE	NUM_OF_NIGHTS	TOTAL_COST	CUSTOMER_ID	BOOKING_NO
	canceled	cash	01/10/2024	6	19800	2	2001
	confirmed	cash	10/10/2025	5	12500	1	2004
	confirmed	cash	12/12/2024	4	6000	1	2002
	confirmed	cash	01/01/2025	8	64000	3	2003

[IT222-1445(2): Le Méridien Hotel]

Table Has

DB tables creation:

The screenshot shows a database table named 'HAS' with three columns: 'BOOKING_NO', 'ROOM_NO', and 'BRANCH_ID'. The 'BOOKING_NO' column is of type NUMBER(10,0) and is set as the primary key (PK). The 'ROOM_NO' column is of type VARCHAR2(3 CHAR). The 'BRANCH_ID' column is of type CHAR(5 CHAR). Below the table definition, the DDL code for creating the table and its constraints is displayed:

```
1 CREATE TABLE "HAS"
2   (
3     "BOOKING_NO" NUMBER(10,0) NOT NULL ENABLE,
4     "ROOM_NO" VARCHAR2(3 CHAR) NOT NULL ENABLE,
5     "BRANCH_ID" CHAR(5 CHAR) NOT NULL ENABLE,
6     CONSTRAINT "HAS_PK" PRIMARY KEY ("BOOKING_NO", "ROOM_NO", "BRANCH_ID")
7     USING INDEX ENABLE
8   );
9
10 ALTER TABLE "HAS" ADD CONSTRAINT "FK_BOOKING_NO" FOREIGN KEY ("BOOKING_NO")
11   REFERENCES "BOOKING" ("BOOKING_NO") ENABLE;
12 ALTER TABLE "HAS" ADD CONSTRAINT "FK_BRANCH_ID" FOREIGN KEY ("BRANCH_ID")
13   REFERENCES "BRANCH" ("BRANCH_ID") ENABLE;
14 ALTER TABLE "HAS" ADD CONSTRAINT "FK_ROOM_NO" FOREIGN KEY ("ROOM_NO", "BRANCH_ID")
15   REFERENCES "ROOM" ("ROOM_NO", "BRANCH_ID") ENABLE;
```

Data insertion:

The screenshot shows the 'Data' tab of the 'HAS' table. The table has three columns: 'BOOKING_NO', 'ROOM_NO', and 'BRANCH_ID'. The data inserted is as follows:

	BOOKING_NO	ROOM_NO	BRANCH_ID
1	2001	101	10001
2	2001	301	10001
3	2002	302	10002
4	2003	102	10000
5	2003	202	10000
6	2003	303	10000
7	2004	201	10001
8	2004	401	10002

[IT222-1445(2): Le Méridien Hotel]

Table Branch

DB tables creation:

The screenshot shows a database management interface with two tabs: 'BRANCH' and 'DDL'. The 'Columns' tab is selected, displaying the table structure:

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
BRANCH_ID	CHAR(5 CHAR)	N		1		
BRANCH_CITY	VARCHAR2(10 CHAR)	N				

The 'DDL' tab is also visible, showing the SQL code for creating the table:

```
1 CREATE TABLE "BRANCH"
2 (
3     "BRANCH_ID" CHAR(5 CHAR) NOT NULL ENABLE,
4     "BRANCH_CITY" VARCHAR2(10 CHAR) NOT NULL ENABLE,
5     CONSTRAINT "BRANCH_PK" PRIMARY KEY ("BRANCH_ID")
6     USING INDEX ENABLE
7 );
```

Data insertion:

The screenshot shows the 'Data' tab for the 'BRANCH' table. It displays three rows of data:

	BRANCH_ID	BRANCH_CITY
	10002	Riyadh
	10000	Alkhobar
	10001	Jeddah

[IT222-1445(2): Le Méridien Hotel]

Table Contains

DB tables creation:

The screenshot shows a table creation interface with the following details:

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
BOOKING_NO	NUMBER(10,0)	N		1		
SERVICE_ID	CHAR(5 CHAR)	N		2		

```
CREATE TABLE "CONTAINS"
(
    "BOOKING_NO" NUMBER(10,0) NOT NULL ENABLE,
    "SERVICE_ID" CHAR(5 CHAR),
    CONSTRAINT "CONTAINS_PK" PRIMARY KEY ("BOOKING_NO", "SERVICE_ID")
    USING INDEX ENABLE
);
ALTER TABLE "CONTAINS" ADD CONSTRAINT "BOOKING_FK" FOREIGN KEY ("BOOKING_NO")
    REFERENCES "BOOKING" ("BOOKING_NO") ON DELETE CASCADE ENABLE;
ALTER TABLE "CONTAINS" ADD CONSTRAINT "CONTAINS_FK" FOREIGN KEY ("SERVICE_ID")
    REFERENCES "SERVICE" ("SERVICE_ID") ON DELETE CASCADE ENABLE;
```

Data insertion:

	BOOKING_NO	SERVICE_ID
	2001	10000
	2002	10002
	2003	10000
	2004	10000
	2004	10001

[IT222-1445(2): Le Méridien Hotel]

Table Room

DB tables creation:

ROOM

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

+ Add Column Modify Column Rename Column Drop Column UI Defaults Refresh More

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
ROOM_NO	VARCHAR2(3 CHAR)	N		1		
BRANCH_ID	CHAR(5 CHAR)	N		2		
ROOM_TYPE	VARCHAR2(10 CHAR)	N				
ROOM_COST	NUMBER(6,2)	N				
BED_TYPE	VARCHAR2(10 CHAR)	Y	'Queen bed'			
FLOOR_NO	VARCHAR2(2 CHAR)	N				
ROOM_VIEW	VARCHAR2(10 CHAR)	Y	'No view'			

ROOM

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Download Refresh

```

1 CREATE TABLE "ROOM"
2   ("ROOM_NO" VARCHAR2(3 CHAR) NOT NULL ENABLE,
3    "BRANCH_ID" CHAR(5 CHAR) NOT NULL ENABLE,
4    "ROOM_TYPE" VARCHAR2(10 CHAR) NOT NULL ENABLE,
5    "ROOM_COST" NUMBER(6,2) NOT NULL ENABLE,
6    "BED_TYPE" VARCHAR2(10 CHAR) DEFAULT 'Queen bed',
7    "FLOOR_NO" VARCHAR2(2 CHAR) NOT NULL ENABLE,
8    "ROOM_VIEW" VARCHAR2(10 CHAR) DEFAULT 'No view',
9    CHECK ("ROOM_VIEW" IN ('No view', 'With view')) ENABLE,
10   CONSTRAINT "ROOM_PK" PRIMARY KEY ("ROOM_NO", "BRANCH_ID")
11   USING INDEX ENABLE
12 );
13
14 ALTER TABLE "ROOM" ADD CONSTRAINT "FK_BRANCH" FOREIGN KEY ("BRANCH_ID")
15   REFERENCES "BRANCH" ("BRANCH_ID") ON DELETE CASCADE ENABLE;

```

Data insertion:

ROOM

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

+ Insert Row Columns... Filter... Count Rows Load Data Download Refresh

	ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW
<input type="text"/>	302	10002	room	1500	Queen	3	With view
<input type="text"/>	101	10001	room	1000	Singal	1	With view
<input type="text"/>	202	10000	suite	2500	Queen	2	No view
<input type="text"/>	201	10001	room	1000	Single	2	No view
<input type="text"/>	102	10000	room	2000	Queen	1	With view
<input type="text"/>	301	10001	suite	2300	Queen	3	No view
<input type="text"/>	303	10000	suite	3500	Queen	3	With view
<input type="text"/>	401	10002	room	1500	Single	4	With view

[IT222-1445(2): Le Méridien Hotel]

Table Service

DB tables creation:

The screenshot shows the 'Columns' tab for the 'SERVICE' table. It lists three columns: 'SERVICE_ID' (CHAR(5 CHAR), NOT NULL, Primary Key), 'SERVICE_NAME' (VARCHAR2(10 CHAR), NOT NULL), and 'SERVICE_COST' (NUMBER(6,2), NOT NULL). The 'Primary Key' column is marked with a '1'.

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
SERVICE_ID	CHAR(5 CHAR)	N		1		
SERVICE_NAME	VARCHAR2(10 CHAR)	N				
SERVICE_COST	NUMBER(6,2)	N				

The screenshot shows the 'DDL' tab for the 'SERVICE' table. It displays the SQL code for creating the table:

```
1 CREATE TABLE "SERVICE"
2   (   "SERVICE_ID" CHAR(5 CHAR) NOT NULL ENABLE,
3       "SERVICE_NAME" VARCHAR2(10 CHAR) NOT NULL ENABLE,
4       "SERVICE_COST" NUMBER(6,2) NOT NULL ENABLE,
5       CONSTRAINT "SERVICE_PK" PRIMARY KEY ("SERVICE_ID")
6       USING INDEX ENABLE
7   ) ;
```

Data insertion:

The screenshot shows the 'Data' tab for the 'SERVICE' table. It displays three rows of data:

	SERVICE_ID	SERVICE_NAME	SERVICE_COST
	10000	Breakfast	0
	10002	Car rental	600
	10001	Dinner	200

[IT222-1445(2): Le Méridien Hotel]

Table Customer

DB tables creation:

The screenshot shows the Oracle SQL Developer interface for creating a table named 'CUSTOMER'. The 'DDL' tab is selected, displaying the SQL code for the table's definition:

```
1 CREATE TABLE "CUSTOMER"
2   "FIRST_NAME" VARCHAR2(10 CHAR) NOT NULL ENABLE,
3   "LAST_NAME" VARCHAR2(10 CHAR) NOT NULL ENABLE,
4   "EMAIL" VARCHAR2(64 CHAR) NOT NULL ENABLE,
5   "CUSTOMER_ID" NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY MINVALUE 1 MAXVALUE 99999999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 28 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,
6   CONSTRAINT "PK_CUSTOMER_ID" PRIMARY KEY ("CUSTOMER_ID")
7   INDEX ENABLE;
```

Data insertion:

The screenshot shows the Oracle SQL Developer interface for inserting data into the 'CUSTOMER' table. The 'Data' tab is selected, and the table contains the following data:

	FIRST_NAME	LAST_NAME	EMAIL	CUSTOMER_ID
	Yasser	Ahmed	Yasser@example.com	1
	Ahmed	Abdullah	Ahmed@example.com	3
	Jana	Abdullah	Jana@gmail.com	2

[IT222-1445(2): Le Méridien Hotel]

Table Telephone

DB tables creation:

The screenshot shows the Oracle Database SQL Developer interface. The top navigation bar has tabs for 'Columns', 'Data', 'Indexes', 'Constraints', 'Grants', 'Statistics', 'Triggers', 'Dependencies', 'DDL', and 'Sample Queries'. The 'DDL' tab is selected. Below the tabs is a toolbar with buttons for 'Add Column', 'Modify Column', 'Rename Column', 'Drop Column', 'UI Defaults', 'Refresh', and 'More'. A table below the toolbar lists columns: PHONE_NO (CHAR(10 CHAR), Nullable N, Primary Key 1) and CUSTOMER_ID (NUMBER(10,0), Nullable N, Identity 2). The bottom half of the screen displays the DDL code for creating the table:

```
1 CREATE TABLE "TELEPHONE"
2 (
3     "PHONE_NO" CHAR(10 CHAR) NOT NULL ENABLE,
4     "CUSTOMER_ID" NUMBER(10,0) NOT NULL ENABLE,
5     CONSTRAINT "TELEPHONE_PK" PRIMARY KEY ("PHONE_NO", "CUSTOMER_ID")
6     USING INDEX ENABLE
7 );
8
9 ALTER TABLE "TELEPHONE" ADD CONSTRAINT "FK_TELEPHONE_CUSTOMER_ID" FOREIGN KEY ("CUSTOMER_ID")
10    REFERENCES "CUSTOMER" ("CUSTOMER_ID") ENABLE;
```

Data insertion:

The screenshot shows the Oracle Database SQL Developer interface. The top navigation bar has tabs for 'Columns', 'Data', 'Indexes', 'Constraints', 'Grants', 'Statistics', 'Triggers', 'Dependencies', 'DDL', and 'Sample Queries'. The 'Data' tab is selected. Below the tabs is a toolbar with buttons for 'Insert Row', 'Columns...', 'Filter...', 'Count Rows', 'Load Data', 'Download', and 'Refresh'. A table below the toolbar lists three rows of data: PHONE_NO (0543859687, CUSTOMER_ID 1), PHONE_NO (0543927163, CUSTOMER_ID 2), and PHONE_NO (0593218643, CUSTOMER_ID 3).

[IT222-1445(2): Le Méridien Hotel]

Data Queries:

1. Show the list of rooms at the Jeddah branch located on the first floor:

The screenshot shows a SQL command window with the following details:

SQL Commands

Language: SQL | Rows: 10 | Clear Command | Find Tables | Save | Run

Query:

```
1 Select *
2   from ROOM
3  where BRANCH_ID=
4    (select BRANCH_ID
5      from BRANCH
6     where BRANCH_CITY='Jeddah'
7    )
8   AND FLOOR_NO='1' AND ROOM_TYPE='room'
```

Results:

ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW
101	10001	room	1000	Singal	1	With view

1 rows returned in 0.05 seconds | Download

2. Show the list of suites at the AlKhobar branch priced between 2000 and 3000:

The screenshot shows a SQL command window with the following details:

SQL Commands

Language: SQL | Rows: 10 | Clear Command | Find Tables | Save | Run

Query:

```
1 Select *
2   from ROOM
3  where BRANCH_ID=
4    (select BRANCH_ID
5      from BRANCH
6     where BRANCH_CITY='Alkhobar'
7    )
8   AND ROOM_COST BETWEEN 2000 AND 3000 AND ROOM_TYPE='suite'
```

Results:

ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW
202	10000	suite	2500	Queen	2	No view

1 rows returned in 0.01 seconds | Download

3. List of free services:

The screenshot shows a SQL command window with the following details:

SQL Commands

Language: SQL | Rows: 10 | Clear Command | Find Tables | Save | Run

Query:

```
1 select *
2   from SERVICE
3  where SERVICE_COST=0
```

Results:

SERVICE_ID	SERVICE_NAME	SERVICE_COST
10000	Breakfast	0

1 rows returned in 0.01 seconds | Download

[IT222-1445(2): Le Méridien Hotel]

4. Calculate the total price of all services requested by the customer:

The screenshot shows a SQL command window with the following details:

SQL Commands:

```
1 SELECT b.BOOKING_NO, b.CUSTOMER_ID ,SUM(s.SERVICE_COST) AS totalservicestotal
2 FROM BOOKING b
3 JOIN CONTAINS c ON b.BOOKING_NO = c.BOOKING_NO
4 JOIN SERVICE s ON c.SERVICE_ID = s.SERVICE_ID
5 WHERE b.CUSTOMER_ID=1
6 GROUP BY b.BOOKING_NO,b.CUSTOMER_ID
```

Results:

BOOKING_NO	CUSTOMER_ID	TOTALSERVICECOST
2002	1	600
2004	1	200

2 rows returned in 0.06 seconds [Download](#)

5. Show the list of rooms and suites sorted from lowest to highest price at the Al Khobar branch:

The screenshot shows a SQL command window with the following details:

SQL Commands:

```
1 SELECT *
2 FROM ROOM R
3 JOIN BRANCH B ON R.BRANCH_ID = B.BRANCH_ID
4 WHERE B.BRANCH_CITY = 'Alkhobar'
5 ORDER BY R.ROOM_COST ASC;
```

Results:

ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW	BRANCH_ID	BRANCH_CITY
102	10000	room	2000	Queen	1	With view	10000	Alkhobar
202	10000	suite	2500	Queen	2	No view	10000	Alkhobar
305	10000	suite	3500	Queen	3	With view	10000	Alkhobar

3 rows returned in 0.01 seconds [Download](#)

[IT222-1445(2): Le Méridien Hotel]

6. Show the list of rooms with a view at the Jeddah branch:

The screenshot shows a SQL command window and its results. The SQL command is:

```
1 SELECT *
2 FROM ROOM R
3 JOIN BRANCH B ON R.BRANCH_ID = B.BRANCH_ID
4 WHERE B.BRANCH_CITY = 'Jeddah'
5 AND R.ROOM_VIEW = 'With view'
6 AND R.ROOM_TYPE = 'room';
7
```

The results table has the following columns: ROOM_NO, BRANCH_ID, ROOM_TYPE, ROOM_COST, BED_TYPE, FLOOR_NO, ROOM_VIEW, BRANCH_ID, and BRANCH_CITY. The single row returned is:

ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW	BRANCH_ID	BRANCH_CITY
101	10001	room	1000	Singal	1	With view	10001	Jeddah

1 rows returned in 0.05 seconds [Download](#)

7. Show the list of Single bed rooms at the Riyadh branch:

The screenshot shows a SQL command window and its results. The SQL command is:

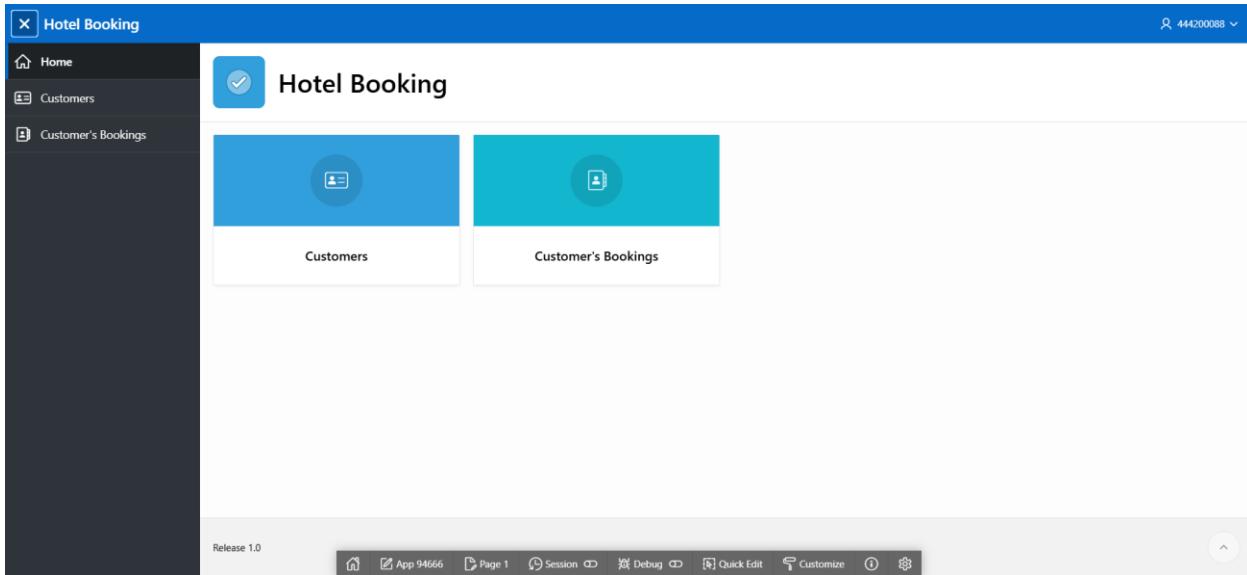
```
1 SELECT *
2 FROM ROOM R
3 JOIN BRANCH B ON R.BRANCH_ID = B.BRANCH_ID
4 WHERE B.BRANCH_CITY = 'Riyadh'
5 AND R.ROOM_TYPE = 'room'
6 AND R.BED_TYPE = 'Single';
```

The results table has the following columns: ROOM_NO, BRANCH_ID, ROOM_TYPE, ROOM_COST, BED_TYPE, FLOOR_NO, ROOM_VIEW, BRANCH_ID, and BRANCH_CITY. The single row returned is:

ROOM_NO	BRANCH_ID	ROOM_TYPE	ROOM_COST	BED_TYPE	FLOOR_NO	ROOM_VIEW	BRANCH_ID	BRANCH_CITY
401	10002	room	1500	Single	4	With view	10002	Riyadh

1 rows returned in 0.05 seconds [Download](#)

App Builder:



- Interactive Grid:

The screenshot shows a web-based application interface titled "Hotel Booking". On the left, a sidebar menu includes "Home", "Customers", and "Customer's Bookings". The main content area is titled "Customers" and displays a table with three rows of data. The columns are "First Name", "Last Name", and "Email". The data is as follows:

First Name	Last Name	Email
Yasser	Ahmed	Yasser@example.com
Ahmed	Abdullah	Ahmed@example.com
Jana	Abdullah	Jana@gmail.com

At the bottom of the grid, it says "1 rows selected" and "Total 3". The footer of the page includes "Release 1.0" and various navigation links like "App 94666", "Page 2", "Session", "Debug", "Quick Edit", "Customize", and "Help".

- Master Detail:

The screenshot shows a web-based application interface titled "Hotel Booking". On the left, a sidebar menu includes "Home", "Customers", and "Customer's Bookings". The main content area is titled "Customer's Bookings" and displays two sections: "Customer" and "Booking".

Customer Section:

Customer	First Name	Last Name	Email
Ahmed	Yasser	Ahmed	Yasser@example.com
Jana			Jana@gmail.com
Yasser			Yasser@example.com

Booking Section:

Booking Status	Payment Info	Checkin Date	Num of Nights	Total Cost
confirmed	cash	10/10/2025	5	12500
confirmed	cash	12/12/2024	4	6000

At the bottom of the booking section, it says "1 - 2". The footer of the page includes "Release 1.0" and various navigation links like "App 94666", "Page 3", "Session", "Debug", "Quick Edit", "Customize", and "Help".

Work Distribution:

NAME	ID	Percentage	WORK
<i>Alanoud Khalid Aloraydi</i>	444201111	20%	<ul style="list-style-type: none"> • <i>Transaction Requirements</i> • <i>Data Dictionary showing description of all entities.</i> • <i>Apex (Table Room + queries)</i>
<i>Sarah alruwayte</i>	444200758	20%	<ul style="list-style-type: none"> • <i>Data Dictionary showing description of all attributes.</i> • <i>Apex (Table (Service+ Contain) + queries)</i>
<i>Walaa Saif Aleslam Mohammed</i>	444200088	20%	<ul style="list-style-type: none"> • <i>Data requirements.</i> • <i>Relational Schema.</i> • <i>Apex (Table (Booking+ Has) + app builder)</i>
<i>Atheer Bin Badie</i>	444200894	20%	<ul style="list-style-type: none"> • <i>EER diagram</i> • <i>Data Dictionary showing description of all attributes.</i> • <i>Apex (Table Branch + app builder)</i>
<i>Jans Alsuwailem</i>	444200816	20%	<ul style="list-style-type: none"> • <i>Project description</i> • <i>View description</i> • <i>Data Dictionary showing description of all relationships.</i> • <i>Apex (Table (Customer + Telephone) + app builder)</i>