

CP 363 Assignment 5

Tourism Agency Booking System DBMS

Group 37

February 14, 2025

Hibah Hibah

Donil Patel

```

1  -- Query 1: Customers who have booked at least one tour (Using EXISTS)
2  • SELECT C.first_name AS "Customer First Name",
3      C.last_name AS "Customer Last Name"
4      FROM Customer C
5      WHERE EXISTS (
6          SELECT 1 FROM Booking B WHERE B.customer_id = C.customer_id
7      );
8
9  -- Query 2: Union of customers who booked expensive tours and customers who made high payments (Using UNION)
10 • SELECT C.first_name, C.last_name, 'High Booking' AS Type
11     FROM Customer C
12     JOIN Booking B ON C.customer_id = B.customer_id
13     WHERE B.total_cost > 2500
14     UNION
15     SELECT C.first_name, C.last_name, 'High Payment' AS Type
16     FROM Customer C
17     JOIN Booking B ON C.customer_id = B.customer_id
18     JOIN Payment P ON B.booking_id = P.booking_id
19     WHERE P.payment_amount > 2500;
20

```

```

-- Query 3: Customers who booked a tour but have NOT made payments yet (Using MINUS equivalent with NOT IN)
SELECT C.first_name, C.last_name
FROM Customer C
JOIN Booking B ON C.customer_id = B.customer_id
WHERE B.booking_id NOT IN (SELECT booking_id FROM Payment);

-- Query 4: Count the number of bookings per customer and categorize them (Using COUNT and CASE)
SELECT C.first_name AS "Customer First Name",
       C.last_name AS "Customer Last Name",
       COUNT(B.booking_id) AS "Total Bookings",
       CASE
           WHEN COUNT(B.booking_id) = 0 THEN 'No Bookings'
           WHEN COUNT(B.booking_id) = 1 THEN 'Single Booking'
           ELSE 'Multiple Bookings'
       END AS "Booking Category"
FROM Customer C
LEFT JOIN Booking B ON C.customer_id = B.customer_id
GROUP BY C.customer_id;

```

```

-- Query 5: Average and Standard Deviation of Tour Prices (Statistical Function)
• SELECT AVG(price_per_person) AS "Average Price",
      STDDEV(price_per_person) AS "Price Standard Deviation"
FROM Tour_Package;

-- Query 6: Most expensive and least expensive package destinations (Nested Subqueries)
• SELECT package_name, destination, price_per_person
FROM Tour_Package
WHERE price_per_person = (SELECT MAX(price_per_person) FROM Tour_Package)
   OR price_per_person = (SELECT MIN(price_per_person) FROM Tour_Package);

-- Query 7: Total revenue per destination, only showing destinations that made more than $3000 (Using GROUP BY and HAVING)
• SELECT TP.destination AS "Destination",
      SUM(B.total_cost) AS "Total Revenue ($)"
FROM Tour_Package TP
JOIN Booking B ON TP.package_id = B.package_id
GROUP BY TP.destination
HAVING SUM(B.total_cost) > 3000;

```

```

-- View 1: Customers and their total spending, using a subquery in the columns
CREATE VIEW CustomerTotalSpending AS
SELECT C.customer_id AS "Customer ID",
       CONCAT(C.first_name, ' ', C.last_name) AS "Customer Name",
       (SELECT SUM(B.total_cost) FROM Booking B WHERE B.customer_id = C.customer_id) AS "Total Spent ($)"
FROM Customer C;

-- View 2: Popular tour packages (using a subquery in the FROM clause)
CREATE VIEW PopularTourPackages AS
SELECT T.package_name AS "Tour Package",
       T.destination AS "Destination",
       T.TotalTourists
FROM (SELECT TP.package_name,
            TP.destination,
            SUM(B.number_of_people) AS TotalTourists
      FROM Tour_Package TP
      JOIN Booking B ON TP.package_id = B.package_id
      GROUP BY TP.package_name, TP.destination) T
WHERE T.TotalTourists > 10;

```

```

-- View 3: Above Average Bookings (using a subquery in the WHERE clause)
CREATE VIEW AboveAverageBookings AS
SELECT B.booking_id AS "Booking ID",
       CONCAT(C.first_name, ' ', C.last_name) AS "Customer Name",
       TP.package_name AS "Tour Package",
       B.number_of_people AS "Number of People"
FROM Booking B
JOIN Customer C ON B.customer_id = C.customer_id
JOIN Tour_Package TP ON B.package_id = TP.package_id
WHERE B.number_of_people > (
    SELECT AVG(B2.number_of_people) FROM Booking B2 WHERE B2.package_id = B.package_id
);

```

Administration Schemas Query 3 SQL File 3\* SQL File 4\* SQL File 5\* SQL File 6\* SQL File 7\* customerspending paymentssummary tourpackagepopularity

SCHEMAS Filter objects

- ap
- ex
- om
- sys
- TourismAgencyDBMS
  - Tables
  - Views
    - customerspending
    - paymentssummary
    - tourpackagepopula...
  - Stored Procedures
  - Functions

Object Info Session

Schema: customerspending 1

1 SELECT \* FROM TourismAgencyDBMS.customerspending;

100% 50:1

Result Grid Filter Rows: Search Export:

customer_id	First Name	Last Name	Total Bookin...	Total Spent (...)
1	Donil	Patel	1	2400.00
2	Hibah	Hibah	1	2700.00
3	Shayna	Mehta	1	750.00
4	Serena	Gomez	1	6000.00
5	Arjun	Kapoor	1	2600.00

Result Grid Form Editor Field Types Query

Read Only

AdministrationSchemas

Query 3SQL File 3\*SQL File 4\*SQL File 5\*SQL File 6\*SQL File 7\*customerspendingpaymentssummarytourpackagepopularity

SCHEMAS

Filter objects

apexexomsysTourismAgencyDBMS

TablesViews

customerspendingpaymentssummarytourpackagepopula...Stored ProceduresFunctions

Object InfoSession

Schema: ex

1SELECT \* FROM TourismAgencyDBMS.paymentssummary;

Limit to 1000 rows

100%1:1

Result Grid

Filter Rows:SearchExport:

payment_id	Customer First Na...	Customer Last Na...	Tour Package ID	Tour Package	Amount Paid (...)	Payment Method	Payment Date
5	Arjun	Kapoor	5	Historical Journey	2600.00	Online	2025-09-06
4	Serena	Gomez	4	Jungle Safari	6000.00	Cash	2025-08-11
3	Shayna	Mehta	3	City Tour	750.00	Cash	2025-07-21
2	Hibah	Hibah	2	Mountain Hiking	2700.00	Online	2025-06-16
1	Doni	Patel	1	Beach Adventure	2400.00	Credit Card	2025-05-02

Result GridForm EditorField TypesQuery Snippets

paymentssummary 1Read Only

AdministrationSchemas

Query 3SQL File 3\*SQL File 4\*SQL File 5\*SQL File 6\*SQL File 7\*customerspendingpaymentssummarytourpackagepopularity

SCHEMAS

Filter objects

apexexomsysTourismAgencyDBMS

TablesViews

customerspendingpaymentssummarytourpackagepopula...Stored ProceduresFunctions

Object InfoSession

Schema: ex

1SELECT \* FROM TourismAgencyDBMS.tourpackagepopularity;

Limit to 1000 rows

100%1:1

Result Grid

Filter Rows:SearchExport:

package_id	Tour Package	Destination	Total Bookin...	Total Touri...
1	Beach Adventure	Hawaii	1	2
2	Mountain Hiking	Bariff	1	3
3	City Tour	New York	1	1
4	Jungle Safari	Kenya	1	4
5	Historical Journey	Rome	1	2

Result GridForm EditorField TypesQuery Snippets

tourpackagepopularity 1Read Only

Context Help

Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

▲ 1	19:45:30	DROP DATABASE IF EXISTS TourismAgencyDBMS	0 row(s) affected, 1 warning(s): 1008 Can't drop data...	0.00052 sec
✓ 2	19:45:30	CREATE DATABASE TourismAgencyDBMS	1 row(s) affected	0.0010 sec
✓ 3	19:45:30	USE TourismAgencyDBMS	0 row(s) affected	0.00017 sec
✓ 4	19:45:30	CREATE TABLE Customer ( customer_id INT PRIMARY KEY NOT NULL UNIQUE, first_name...	0 row(s) affected	0.015 sec
✓ 5	19:45:30	CREATE TABLE Tour_Package ( package_id INT PRIMARY KEY NOT NULL UNIQUE, packa...	0 row(s) affected	0.0072 sec
✓ 6	19:45:30	CREATE TABLE Booking ( booking_id INT PRIMARY KEY NOT NULL UNIQUE, booking_dat...	0 row(s) affected	0.011 sec
✓ 7	19:45:30	CREATE TABLE Payment ( payment_id INT PRIMARY KEY NOT NULL UNIQUE, booking_id...	0 row(s) affected	0.0046 sec
✓ 8	19:45:30	CREATE TABLE Guide ( guide_id INT PRIMARY KEY NOT NULL UNIQUE, guide_name VAR...	0 row(s) affected	0.0044 sec
✓ 9	19:45:30	CREATE TABLE Tour_Schedule ( schedule_id INT PRIMARY KEY NOT NULL UNIQUE, pack...	0 row(s) affected	0.0061 sec
✓ 10	19:45:30	CREATE TABLE Hotel ( hotel_id INT PRIMARY KEY NOT NULL UNIQUE, hotel_name VARC...	0 row(s) affected	0.0039 sec
✓ 11	19:45:30	CREATE TABLE Package_Hotel ( package_id INT NOT NULL, hotel_id INT NOT NULL, st...	0 row(s) affected	0.0044 sec
✓ 12	19:45:30	CREATE TABLE Transportation ( transport_id INT PRIMARY KEY NOT NULL UNIQUE, trans...	0 row(s) affected	0.0035 sec
✓ 13	19:45:30	CREATE TABLE Package_Transport ( package_id INT NOT NULL, transport_id INT NOT N...	0 row(s) affected	0.0040 sec
✓ 14	19:45:30	SHOW TABLES	10 row(s) returned	0.00059 sec / 0.000...
✓ 15	19:45:37	USE TourismAgencyDBMS	0 row(s) affected	0.00022 sec
✓ 16	19:45:37	INSERT INTO Customer (customer_id, first_name, last_name, email, phone, address) VALUES...	5 row(s) affected Records: 5 Duplicates: 0 Warnings...	0.0019 sec
✓ 17	19:45:37	INSERT INTO Tour_Package (package_id, package_name, destination, price_per_person, start...	5 row(s) affected Records: 5 Duplicates: 0 Warnings...	0.0022 sec
✓ 18	19:45:37	INSERT INTO Booking (booking_id, booking_date, customer_id, package_id, number_of_peopl...	5 row(s) affected Records: 5 Duplicates: 0 Warnings...	0.0077 sec
✓ 19	19:45:37	INSERT INTO Payment (payment_id, booking_id, payment_date, payment_amount, payment_...	5 row(s) affected Records: 5 Duplicates: 0 Warnings...	0.0014 sec
✓ 20	19:45:37	INSERT INTO Guide (guide_id, guide_name, contact_number, language, availability_status) VA...	5 row(s) affected Records: 5 Duplicates: 0 Warnings...	0.0024 sec
✓ 21	19:45:37	SELECT * FROM Customer LIMIT 0, 1000	5 row(s) returned	0.00033 sec / 0.0000...
✓ 22	19:45:37	SELECT * FROM Tour_Package LIMIT 0, 1000	5 row(s) returned	0.0070 sec / 0.00000...
✓ 23	19:45:37	SELECT * FROM Booking LIMIT 0, 1000	5 row(s) returned	0.00023 sec / 0.0000...
✓ 24	19:45:37	SELECT * FROM Payment LIMIT 0, 1000	5 row(s) returned	0.00017 sec / 0.0000...
✓ 25	19:45:37	SELECT * FROM Guide LIMIT 0, 1000	5 row(s) returned	0.00017 sec / 0.0000...
✓ 26	19:46:07	CREATE VIEW CustomerSpending AS SELECT C.customer_id, C.first_name AS "First Na...	0 row(s) affected	0.0020 sec
✓ 27	19:46:11	CREATE VIEW TourPackagePopularity AS SELECT TP.package_id, TP.package_name AS...	0 row(s) affected	0.0050 sec
✓ 28	19:46:15	CREATE VIEW PaymentSummary AS SELECT P.payment_id, C.first_name AS "Customer...	0 row(s) affected	0.0028 sec
✓ 29	19:47:14	SELECT C.first_name AS "Customer First Name", C.last_name AS "Customer Last Name"...	5 row(s) returned	0.0018 sec / 0.00001...
✓ 30	19:47:14	SELECT C.first_name, C.last_name, 'High Booking' AS Type FROM Customer C JOIN Booking...	6 row(s) returned	0.0036 sec / 0.0000...
✓ 31	19:47:14	SELECT C.first_name, C.last_name FROM Customer C JOIN Booking B ON C.customer_id = B...	0 row(s) returned	0.00053 sec / 0.0000...
✓ 32	19:47:14	SELECT C.first_name AS "Customer First Name", C.last_name AS "Customer Last Name"...	5 row(s) returned	0.0031 sec / 0.00000...
✓ 33	19:47:14	SELECT AVG(price_per_person) AS "Average Price", STDDEV(price_per_person) AS "Pric...	1 row(s) returned	0.00075 sec / 0.0000...
✓ 34	19:47:14	SELECT package_name, destination, price_per_person FROM Tour_Package WHERE price_per...	2 row(s) returned	0.00074 sec / 0.0000...
✓ 35	19:47:14	SELECT TP.destination AS "Destination", SUM(B.total_cost) AS "Total Revenue (\$)" FROM...	1 row(s) returned	0.00053 sec / 0.0000...
✓ 36	19:47:14	CREATE VIEW CustomerTotalSpending AS SELECT C.customer_id AS "Customer ID", CON...	0 row(s) affected	0.0038 sec
✓ 37	19:47:14	CREATE VIEW PopularTourPackages AS SELECT T.package_name AS "Tour Package", T.d...	0 row(s) affected	0.0033 sec
✓ 38	19:47:14	CREATE VIEW AboveAverageBookings AS SELECT B.booking_id AS "Booking ID", CONCA...	0 row(s) affected	0.0016 sec
✗ 39	19:47:24	CREATE VIEW CustomerTotalSpending AS SELECT C.customer_id AS "Customer ID", CON...	Error Code: 1050. Table 'CustomerTotalSpending' alre...	0.00081 sec
✓ 40	19:47:44	SELECT * FROM TourismAgencyDBMS.customerspending LIMIT 0, 1000	5 row(s) returned	0.00080 sec / 0.000...
✓ 41	19:47:47	SELECT * FROM TourismAgencyDBMS.paymentsummary LIMIT 0, 1000	5 row(s) returned	0.0010 sec / 0.00000...
✓ 42	19:47:49	SELECT * FROM TourismAgencyDBMS.tourpackagepopularity LIMIT 0, 1000	5 row(s) returned	0.0016 sec / 0.00000...