

ANIKET CHAKRABORTY CAST STUDY-1 ANSWER SHEET

Query – 1:

Question-1: Display the number of states present in the LocationTable.
`SELECT
COUNT(DISTINCT(State)) AS NumStates
FROM LocationTable;`

Answer – 1:

	NumStates
1	20

Query – 2:

Question-2: How many products are of regular type
`SELECT
COUNT(*) AS NumRegularProduct
FROM ProductTable WHERE Type = 'Regular';`

Answer – 2:

	NumRegularProduct
1	8

Query – 3:

Question-3: How much spending has been done on marketing of product ID 1
`SELECT
ProductId, SUM(Marketing) AS TotalMarketPrice
FROM FactsTable
WHERE ProductId = 1 GROUP BY ProductId;`

Answer – 3:

	ProductId	TotalMarketPrice
1	1	4658

Query – 4:

Question-4: What is the minimum sales of a product
`SELECT
MIN(Sales) AS MinSales
FROM FactsTable;`

Answer – 4:

	MinSales
1	17

Query – 5:

```
- Question-5: Display the max Cost of Good Sold (COGS).  
SELECT  
    MAX(COGS) AS MaxGoldSold  
FROM FactsTable;
```

Answer – 5:

	MaxGoldSold
1	364

Query – 6:

```
Question-6: Display the details of the product where product type is coffee.  
SELECT TOP 10  
    F.Profit, F.Sales, F.Margin, F.COGS ,F.Total_Expenses, F.Marketing,  
    F.Inventory, F.Budget_Profit, F.Budget_COGS, F.Budget_Margin, F.Budget_Sales,  
    F.Area_Code, P.Product_Type  
FROM FactsTable F JOIN ProductTable P ON F.ProductId = P. ProductId  
WHERE P.Product_Type = 'Coffee';
```

Answer – 6:

	Profit	Sales	Margin	COGS	Total_Expenses	Marketing	Inventory	Budget_Profit	Budget_COGS	Budget_Margin	Budget_Sales	Area_Code	Product_Type
1	94	219	130	89	36	24	777	100	90	130	220	719	Coffee
2	11	45	27	18	16	5	821	20	10	30	40	319	Coffee
3	34	140	80	60	46	19	336	50	60	80	140	740	Coffee
4	13	120	66	54	53	20	312	30	50	70	120	262	Coffee
5	20	93	55	38	35	12	241	30	30	50	80	603	Coffee
6	47	170	92	78	45	24	965	80	90	120	210	801	Coffee
7	-2	145	50	95	52	30	608	30	110	70	180	916	Coffee
8	1	150	87	63	86	57	435	40	70	110	180	541	Coffee
9	105	243	144	99	39	27	766	120	100	150	250	303	Coffee
10	10	45	27	18	17	5	818	20	10	30	40	712	Coffee

Query – 7:

```
Question-7: Display the details where total expenses are greater than 40  
SELECT TOP 10*  
FROM FactsTable  
WHERE Total_Expenses > 40;
```

Answer – 7:

	Date	ProductId	Profit	Sales	Margin	COGS	Total_Expenses	Marketing	Inventory	Budget_Profit	Budget_COGS	Budget_Margin	Budget_Sales	Area_Code
1	2010-01-01	6	53	180	108	72	55	23	558	80	80	130	210	720
2	2010-01-01	8	99	341	171	170	72	47	1091	110	140	160	300	970
3	2010-01-01	9	0	150	87	63	87	57	435	20	50	80	130	719
4	2010-01-01	10	33	140	80	60	47	19	336	40	50	70	120	970
5	2010-01-01	11	17	130	72	58	55	22	338	20	40	70	110	719
6	2010-01-01	2	111	345	201	144	90	47	862	130	150	210	360	217
7	2010-01-01	3	87	234	139	95	52	30	608	100	100	140	240	309
8	2010-01-01	5	203	546	312	234	109	77	1310	260	270	370	640	309
9	2010-01-01	6	140	456	228	228	88	63	1459	180	260	270	530	630
10	2010-01-01	12	54	180	108	72	54	23	558	40	60	90	150	708

Query – 8:

Question-8: What is the average sales in area code 719

```
SELECT
    AVG(Sales) AS MeanSales
FROM FactsTable
WHERE Area_Code = 719;
```

Answer – 8:

	MeanSales
1	186

Query – 9:

Question-9: Find out the total profit generated by Colorado state

```
SELECT
    SUM(Profit) AS Total_Colorado_Profit
FROM FactsTable
WHERE Area_Code IN
    (SELECT Area_Code FROM LocationTable WHERE State = 'Colorado');
```

Answer – 9:

	Total_Colorado_Profit
1	17743

Query – 10:

Question-10: Display the average inventory for each product ID

```
SELECT
    ProductId, AVG(Inventory) AS ProductWise_Mean_Inventory
FROM FactsTable
GROUP BY ProductId ORDER BY ProductId;
```

Answer – 10:

	ProductId	ProductWise_Mean_Inventory
1	1	741
2	2	816
3	3	838
4	4	696
5	5	756
6	6	755
7	7	879
8	8	712
9	9	718
10	10	1095
11	11	737
12	12	757
13	13	900

Query – 11:

Question-11: Display state in a sequential order in a LocationTable

```
SELECT
    DISTINCT(State) AS State_Name
FROM LocationTable ORDER BY State;
```

Answer – 11:

	State_Name
1	California
2	Colorado
3	Connecticut
4	Florida
5	Illinois
6	Iowa
7	Louisiana
8	Massachusetts
9	Missouri
10	Nevada
11	New Hampshire
12	New Mexico
13	New York
14	Ohio
15	Oklahoma
16	Oregon
17	Texas
18	Utah
19	Washington
20	Wisconsin

Query – 12:

Question-12: Display the average budget of the Product where the average budget margin should be greater than 100.

```
SELECT
    ProductId, AVG(Budget_Margin) AS AverageBudget
FROM FactsTable
GROUP BY ProductId
HAVING AVG(Budget_Margin) > 100;
```

Answer – 12:

	ProductId	AverageBudget
1	12	103
2	6	107
3	7	182
4	2	173

Query – 13:

Question-13: What is the total sales done on date 2010-01-01

```
SELECT
    SUM(Sales) AS TotalSales
FROM FactsTable
WHERE Date = '2010-01-01';
```

Answer – 13:

	TotalSales
1	31555

Query – 14:

Question-14: Display the average total expense of each product ID on an individual date

```
SELECT
    ProductId, Date, AVG(Total_Expenses) AS AvgTotalExpense
FROM FactsTable
GROUP BY ProductId, Date
ORDER BY ProductId, Date;
```

Answer – 14:

	ProductId	Date	AvgTotalExpense
13	1	2011-01-01	48
14	1	2011-01-02	46
15	1	2011-01-03	45
16	1	2011-01-04	46
17	1	2011-01-05	44
18	1	2011-01-06	43
19	1	2011-01-07	43
20	1	2011-01-08	43
21	1	2011-01-09	47
22	1	2011-01-10	47
23	1	2011-01-11	46
24	1	2011-01-12	46
25	2	2010-01-01	63
26	2	2010-01-02	63
27	2	2010-01-03	63
28	2	2010-01-04	65

Query – 15:

Question-15: Display the table with the following attributes such as date, productID, product_type, product, sales, profit, state, area_code

```
SELECT TOP 10
    F.Date, F.ProductId, P.Product_Type, P.Product, F.Sales, F.Profit,
    L.State, F.Area_Code
FROM FactsTable F FULL JOIN LocationTable L ON F.Area_Code = L.Area_Code
FULL JOIN ProductTable P ON F.ProductId = P.ProductId;
```

Answer – 15:

	Date	ProductId	Product_Type	Product	Sales	Profit	State	Area_Code
1	2010-01-01	1	Coffee	Amaretto	219	94	Colorado	719
2	2010-01-01	2	Coffee	Columbian	190	68	Colorado	970
3	2010-01-01	3	Coffee	Decaf Irish Cream	234	101	Colorado	970
4	2010-01-01	13	Tea	Green Tea	100	30	Colorado	303
5	2010-01-01	5	Espresso	Caffe Mocha	134	54	Colorado	303
6	2010-01-01	6	Espresso	Decaf Espresso	180	53	Colorado	720
7	2010-01-01	8	Herbal Tea	Chamomile	341	99	Colorado	970
8	2010-01-01	9	Herbal Tea	Lemon	150	0	Colorado	719
9	2010-01-01	10	Herbal Tea	Mint	140	33	Colorado	970
10	2010-01-01	11	Tea	Darjeeling	130	17	Colorado	719

Query – 16:

Question-16: Display the rank without any gap to show the sales wiserank.

```
SELECT
    ProductID,
    Sales,
    DENSE_RANK() OVER (ORDER BY Sales DESC) AS Sales_Rank
FROM FactsTable;
```


Answer – 16:

	ProductID	Sales	Sales_Rank
1	2	912	1
2	2	912	1
3	2	910	2
4	2	910	2
5	2	904	3
6	2	904	3
7	2	902	4
8	2	902	4
9	2	890	5
10	2	890	5
11	2	882	6
12	2	882	6
13	2	842	7
14	2	842	7
15	7	815	8

Query – 17:

```
- Question-17: Find the state wise profit and sales.  
SELECT  
    L.State,  
    SUM(F.Sales) AS Total_Sales,  
    SUM(F.Profit) AS Total_Profit  
FROM FactsTable F  
JOIN LocationTable L ON F.Area_Code = L.Area_Code  
GROUP BY L.State  
ORDER BY Total_Sales DESC;
```

Answer – 17:

	State	Total_Sales	Total_Profit
1	California	96892	31785
2	New York	70852	20096
3	Illinois	69883	30821
4	Nevada	60159	10616
5	Iowa	54750	22212
6	Colorado	48179	17743
7	Oregon	40899	12439
8	Washington	38930	11405
9	Florida	37443	12310
10	Texas	37410	15766
11	Utah	35384	7751
12	Ohio	34517	10773
13	Wisconsin	33069	8702
14	Massachusetts	29965	16442
15	Oklahoma	27463	8558
16	Connecticut	25429	7621
17	Missouri	24647	3601
18	Louisiana	23161	7355
19	New Mexico	15892	799
20	New Hampsh...	14887	2748

Query – 18:

```
- Question-18: Find the state wise profit and sales along with the productname  
SELECT  
    L.State,  
    SUM(F.Sales) AS Total_Sales,  
    SUM(F.Profit) AS Total_Profit,  
    P.Product  
FROM FactsTable F  
JOIN LocationTable L ON F.Area_Code = L.Area_Code  
JOIN ProductTable P ON F.ProductId = P.ProductId  
GROUP BY L.State, P.Product  
ORDER BY Total_Sales DESC;
```

Answer – 18:

	State	Total_Sales	Total_Profit	Product
1	New York	18245	8565	Columbian
2	California	18245	8566	Columbian
3	New York	14842	6092	Lemon
4	California	14607	6580	Decaf Espresso
5	Iowa	14607	6577	Chamomile
6	Nevada	14607	6580	Darjeeling
7	Illinois	14607	6575	Caffe Mocha
8	New York	14576	7589	Regular Espresso
9	Iowa	14449	5624	Earl Grey
10	Massachusetts	13301	12489	Columbian
11	California	12681	5450	Lemon
12	Texas	12681	5452	Columbian
13	Nevada	12001	4498	Earl Grey
14	Iowa	12001	4487	Darjeeling
15	California	12001	4497	Caffe Latte
16	Illinois	12001	4494	Decaf Espresso
17	Nevada	10610	4356	Lemon
18	Illinois	10610	4362	Columbian
19	Oregon	7798	2672	Decaf Espresso
20	Colorado	7798	2678	Chamomile
21	Massachusetts	7798	2685	Regular Espresso
22	Ohio	7798	2672	Caffe Mocha
23	Nevada	7691	886	Chamomile
24	California	7691	886	Caffe Mocha
25	Iowa	7451	3466	Lemon
26	Illinois	7451	3463	Decaf Irish Cream

Query – 19:

Question-19: If there is an increase in sales of 5%, calculate the increasedsales
`SELECT TOP 10
Sales, (Sales * 1.05) AS IncreasedSales
FROM FactsTable;`

Answer – 19:

	Sales	IncreasedSales
1	219	229.95
2	190	199.50
3	234	245.70
4	100	105.00
5	134	140.70
6	180	189.00
7	341	358.05
8	150	157.50
9	140	147.00
10	130	136.50

Query – 20:

Question-20: Find the maximum profit along with the product ID and product type
`SELECT
F.ProductId, P.Product_Type, MAX(F.Profit) AS MaxProductProfit
FROM FactsTable F
JOIN ProductTable P ON F.ProductId = P.ProductId
GROUP BY F.ProductId, P.Product_Type;`

Answer – 20:

	ProductId	Product_Type	MaxProductProfit
1	1	Coffee	199
2	2	Coffee	778
3	3	Coffee	207
4	4	Espresso	233
5	5	Espresso	362
6	6	Espresso	362
7	7	Espresso	646
8	8	Herbal Tea	362
9	9	Herbal Tea	536
10	10	Herbal Tea	207
11	11	Tea	362
12	12	Tea	331
13	13	Tea	180

Query – 21:

Question-21: Create a stored procedure to fetch the result according to the product type from ProductTable.

```
CREATE PROC USP_ProductType
@ProductType VARCHAR(50)
AS
(
SELECT F.*, P.Product_Type
FROM FactsTable F JOIN ProductTable P ON F.ProductId = P.ProductId
WHERE P.Product_Type = @ProductType
);

EXEC USP_ProductType 'Coffee';
```

Answer – 21:

	Date	ProductId	Profit	Sales	Margin	COGS	Total_Expenses	Marketing	Inventory	Budget_Profit	Budget_COGS	Budget_Margin	Budget_Sales	Area_Code	Product_Type
1	2010-01-01	1	94	219	130	89	36	24	777	100	90	130	220	719	Coffee
2	2010-01-01	2	68	190	107	83	39	27	623	80	80	110	190	970	Coffee
3	2010-01-01	3	101	234	139	95	38	26	821	110	100	140	240	970	Coffee
4	2010-01-01	2	111	345	201	144	90	47	862	130	150	210	360	217	Coffee
5	2010-01-01	3	87	234	139	95	52	30	608	100	100	140	240	309	Coffee
6	2010-01-01	1	11	45	27	18	16	5	821	20	10	30	40	319	Coffee
7	2010-01-01	2	5	62	34	28	29	8	965	20	20	40	60	641	Coffee
8	2010-01-01	3	12	54	31	23	19	7	623	20	20	30	50	712	Coffee
9	2010-01-01	2	39	190	105	85	66	32	494	40	90	100	190	636	Coffee
10	2010-01-01	3	47	170	92	78	45	24	965	50	80	90	170	573	Coffee
11	2010-01-01	1	34	140	80	60	46	19	336	50	60	80	140	740	Coffee
12	2010-01-01	2	42	140	83	57	41	18	364	50	60	80	140	614	Coffee
13	2010-01-01	3	0	150	87	63	87	57	435	20	60	90	150	614	Coffee
14	2010-01-01	1	13	120	66	54	53	20	312	30	50	70	120	262	Coffee
15	2010-01-01	2	9	190	111	79	102	72	551	20	80	110	190	262	Coffee
16	2010-01-01	3	51	180	103	77	52	25	432	60	80	100	180	414	Coffee
17	2010-01-01	2	115	310	186	124	71	40	961	130	110	180	290	475	Coffee
18	2010-01-01	3	67	200	116	84	49	26	500	70	80	110	190	754	Coffee
19	2010-01-01	2	68	210	126	84	58	27	651	70	80	110	190	954	Coffee
20	2010-01-01	2	367	494	422	72	55	23	558	360	60	400	460	413	Coffee
21	2010-01-01	1	20	93	55	38	35	12	241	30	30	50	80	603	Coffee
22	2010-01-01	2	25	120	72	48	47	15	372	40	40	70	110	603	Coffee

Query – 22:

Question-22: Write a query by creating a condition in which if the total expenses is less than 60 then it is a profit or else loss.

```
SELECT TOP 10
    Total_Expenses, Feedback =
    CASE
        WHEN Total_Expenses < 60 THEN 'Profit'
        ELSE 'Loss'
    END
FROM FactsTable;
```

Answer – 22:

	Total_Expenses	Feedback
1	36	Profit
2	39	Profit
3	38	Profit
4	26	Profit
5	26	Profit
6	55	Profit
7	72	Loss
8	87	Loss
9	47	Profit
10	55	Profit

Query – 23:

Question-23: Give the total weekly sales value with the date and product ID details. Use roll-up to pull the data in hierarchical order.

```
SELECT
    DATEPART(WEEK, Date) AS SalesWeek,
    ProductId,
    SUM(Sales) AS TotalWeeklySales
FROM FactsTable
GROUP BY ROLLUP (DATEPART(WEEK, Date), ProductId)
ORDER BY SalesWeek, ProductId;
```

Answer – 23:

	SalesWeek	ProductId	TotalWeeklySales
1	NULL	NULL	819811
2	1	NULL	98963
3	1	1	3365
4	1	2	15120
5	1	3	7461
6	1	4	4594
7	1	5	10172
8	1	6	9776
9	1	7	2887
10	1	8	9289
11	1	9	11067
12	1	10	4378
13	1	11	8896
14	1	12	8025
15	1	13	3933
16	2	NULL	482684
17	2	1	14935
18	2	2	76755
19	2	3	37393
20	2	4	21167
21	2	5	49319
22	2	6	46132
23	2	7	13408
24	2	8	43756
25	2	9	56775
26	2	10	21556
27	2	11	43289

Answer – 24:

```
Question-24: Apply union and intersection operator on the tables which consist of attribute area code.  
SELECT Area_Code FROM FactsTable  
UNION  
SELECT Area_Code FROM LocationTable; ---Result in 156 Rows  
  
SELECT Area_Code FROM FactsTable  
INTERSECT  
SELECT Area_Code FROM LocationTable; ---Result in 156 Rows
```

Query – 24:

Since, number of distinct area codes in 'FactsTable' and 'LocationTable' are same as 156, so the UNION and the INTERSECT operators returns 156 rows in result.

Query – 25:

```
Question-25: Create a user-defined function for the product table to fetch a particular product type based upon the user's preference.  
CREATE FUNCTION DBO.FN_FetchProductType  
(@Input VARCHAR(30))  
RETURNS TABLE AS  
RETURN  
(  
SELECT * FROM ProductTable WHERE Product_Type = @Input  
);  
  
SELECT * FROM dbo.FN_FetchProductType('Coffee');
```

Answer – 25:

	ProductId	Product_Type	Product	Type
1	1	Coffee	Amaretto	Regular
2	2	Coffee	Columbian	Regular
3	3	Coffee	Decaf Irish Cream	Decaf

Query – 26:

```
Question-26: Change the product type from coffee to tea where product ID is 1 and undo it.  
BEGIN TRANSACTION  
UPDATE ProductTable  
SET Product_Type = 'Tea' WHERE ProductId = 1;  
SELECT * FROM ProductTable WHERE ProductId = 1;  
  
ROLLBACK TRANSACTION  
SELECT * FROM ProductTable WHERE ProductId = 1;
```

Answer – 26:

	ProductId	Product_Type	Product	Type
1	1	Tea	Amaretto	Regular

	ProductId	Product_Type	Product	Type
1	1	Coffee	Amaretto	Regular

Before Roll back transaction

After Roll back transaction

Query – 27:

Question-27: Display the date, product ID and sales where total expenses are between 100 to 200

```
SELECT  
    Date, ProductId, Sales  
FROM FactsTable  
WHERE Total_Expenses BETWEEN 100 AND 200;
```

Answer – 27:

	Date	ProductId	Sales
1	2010-01-01	5	546
2	2010-01-01	8	546
3	2010-01-01	12	546
4	2010-01-01	2	190
5	2010-01-01	5	190
6	2010-01-01	5	61
7	2010-01-01	2	678
8	2010-01-01	9	483
9	2010-01-01	9	190
10	2010-01-01	5	250
11	2010-01-01	6	546
12	2010-01-01	2	678
13	2010-01-01	8	250
14	2010-01-01	11	546
15	2010-01-01	13	17
16	2010-01-01	3	190
17	2010-01-02	5	545
18	2010-01-02	8	545
19	2010-01-02	12	534
20	2010-01-02	2	220

Query – 28:

Question-28: Delete the records in the Product Table for regular type.

```
DELETE FROM ProductTable  
WHERE Product_Type = 'Regular';  
SELECT DISTINCT(Product_Type) FROM ProductTable;
```

Answer – 28:

	Product_Type
1	Coffee
2	Espresso
3	Herbal Tea
4	Tea

Query – 29:

Question-29: Display the ASCII value of the fifth character from the Product column

```
SELECT  
    Product, SUBSTRING(Product, 5,1) AS FifthChar,  
    ASCII(SUBSTRING(Product,5,1)) AS FifthAsciiValue  
FROM ProductTable;
```

Answer – 29:

	Product	FifthChar	FifthAsciiValue
1	Amaretto	e	101
2	Columbian	m	109
3	Decaf Irish Cream	f	102
4	Caffe Latte	e	101
5	Caffe Mocha	e	101
6	Decaf Espresso	f	102
7	Regular Espresso	l	108
8	Chamomile	o	111
9	Lemon	n	110
10	Mint		NULL
11	Darjeeling	e	101
12	Earl Grey		32
13	Green Tea	n	110