

NTOMBIZA M

PRACTICAL 1 (ANSWERS)

Q1. Display all columns for all transactions.

RETAILSALES.PUBLIC Settings

1 select* from retailsales;

Results Chart

	#	TRANSACTION_ID	🕒	DATE	👤	CUSTOMER_ID	👤	GENDER	#	AGE	👤	PRODUCT_CATEGORY	#	QUANTITY	#	PRICE_PER_UNIT	#	TOTAL_AMOUNT
1		1		2023-11-24		CUST001		Male		34		Beauty		3		50		150
2		2		2023-02-27		CUST002		Female		26		Clothing		2		500		1000
3		3		2023-01-13		CUST003		Male		50		Electronics		1		30		30
4		4		2023-05-21		CUST004		Male		37		Clothing		1		500		500
5		5		2023-05-06		CUST005		Male		30		Beauty		2		50		100
6		6		2023-04-25		CUST006		Female		45		Beauty		1		30		30
7		7		2023-03-13		CUST007		Male		46		Clothing		2		25		50
8		8		2023-02-22		CUST008		Male		30		Electronics		4		25		100
9		9		2023-12-13		CUST009		Male		63		Electronics		2		300		600

Q2. Display only the Transaction ID, Date, and Customer ID for all records.

RETAILSALES.PUBLIC Settings

1 select transaction_id,
2 DATE,
3 CUSTOMER_ID,
4 from retailsales;

Results Chart

	#	TRANSACTION_ID	🕒	DATE	👤	CUSTOMER_ID
1		1		2023-11-24		CUST001
2		2		2023-02-27		CUST002
3		3		2023-01-13		CUST003
4		4		2023-05-21		CUST004
5		5		2023-05-06		CUST005
6		6		2023-04-25		CUST006
7		7		2023-03-13		CUST007
8		8		2023-02-22		CUST008
9		9		2023-12-13		CUST009

Q3. Display all the distinct product categories in the dataset.

RETAILSALES.PUBLIC Settings

1 select distinct product_category
2 from retailsales

Results Chart

	👤	PRODUCT_CATEGORY
1		Beauty
2		Clothing
3		Electronics

Q4. Display all the distinct gender values in the dataset.

RETAILSALES.PUBLIC Settings

```
1 select distinct gender
2 from retailsales
```

>

Results Chart

	GENDER
1	Male
2	Female

Q5. Display all transactions where the Age is greater than 40.

RETAILSALES.PUBLIC Settings

```
1 select* from retailsales
2 where age >40;
```

>

Results Chart

	# TRANSACTION_ID	🕒 DATE	👤 CUSTOMER_ID	👤 GENDER	# AGE	👤 PRODUCT_CATEGORI	# QUANTITY	# PRICE_PER_UNIT
1	3	2023-01-13	CUST003	Male	50	Electronics	1	30
2	6	2023-04-25	CUST006	Female	45	Beauty	1	30
3	7	2023-03-13	CUST007	Male	46	Clothing	2	25
4	9	2023-12-13	CUST009	Male	63	Electronics	2	300
5	10	2023-10-07	CUST010	Female	52	Clothing	4	50
6	14	2023-01-17	CUST014	Male	64	Clothing	4	30
7	15	2023-01-16	CUST015	Female	42	Electronics	4	500
8	18	2023-04-30	CUST018	Female	47	Electronics	2	25

Q6. Display all transactions where the Price per Unit is between 100 and 50.

```
3 select*from retailsales
4 where price_per_unit
5 between 50 and 100
```

Results Chart

	# TRANSACTION_ID	🕒 DATE	👤 CUSTOMER_ID	👤 GENDER	# AGE	👤 PRODUCT_CATEGORY	# QUANTITY	# PRICE_PER_UNIT	# TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
3	10	2023-10-07	CUST010	Female	52	Clothing	4	50	200
4	11	2023-02-14	CUST011	Male	23	Clothing	2	50	100
5	22	2023-10-15	CUST022	Male	18	Clothing	2	50	100
6	25	2023-12-26	CUST025	Female	64	Beauty	1	50	50
7	33	2023-03-23	CUST033	Female	50	Electronics	2	50	100
8	34	2023-12-24	CUST034	Female	51	Clothing	3	50	150
9	38	2023-03-21	CUST038	Male	38	Beauty	4	50	200

Q7. Display all transactions where the Product Category is either 'Beauty' or 'Electronics'.

```

7  select * from retailsales
8  where product_category = 'Beauty' or
9  product_category = 'Electronics'
10
11

```

#	TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
3	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
4	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
5	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
6	9	2023-12-13	CUST009	Male	63	Electronics	2	300	600
7	12	2023-10-30	CUST012	Male	35	Beauty	3	25	75
8	13	2023-08-05	CUST013	Male	22	Electronics	3	500	1500
9	15	2023-01-16	CUST015	Female	42	Electronics	4	500	2000

Q8. Display all transactions where the Product Category is not 'Clothing'.

```

1  select * from retailsales
2  WHERE NOT product_category = 'Clothing'
3
4
5
6

```

#	TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
1	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
3	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
4	6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
5	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
6	9	2023-12-13	CUST009	Male	63	Electronics	2	300	600
7	12	2023-10-30	CUST012	Male	35	Beauty	3	25	75
8	13	2023-08-05	CUST013	Male	22	Electronics	3	500	1500
9	15	2023-01-16	CUST015	Female	42	Electronics	4	500	2000

Q9. Display all transactions where the Quantity is greater than or equal to 3.

```

1  select * from retailsales
2  WHERE QUANTITY >= 3
3
4
5
6

```

#	TRANSACTION_ID	DATE	CUSTOMER_ID	GENDER	AGE	PRODUCT_CATEGORY	QUANTITY	PRICE_PER_UNIT	TOTAL_AMOUNT
	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
	8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
	10	2023-10-07	CUST010	Female	52	Clothing	4	50	200
	12	2023-10-30	CUST012	Male	35	Beauty	3	25	75
	13	2023-08-05	CUST013	Male	22	Electronics	3	500	1500
	14	2023-01-17	CUST014	Male	64	Clothing	4	30	120
	15	2023-01-16	CUST015	Female	42	Electronics	4	500	2000
	16	2023-02-17	CUST016	Male	19	Clothing	3	500	1500
	17	2023-04-22	CUST017	Female	27	Clothing	4	25	100

Q10. Count the total number of transactions.

```

1  select count (distinct TRANSACTION_ID)
2  from retailsales as TOTAL_TRANSACTIONS
3
4
5
6

```

#	COUNT (DISTINCT TRANSACTION_ID)
	1000

Q11. Find the average Age of customers.

```
1 select AVG(AGE) AS average_age
2 FROM retailsales
3
4
5
6
```

# AVERAGE_AGE	
	41.392000

Q12. Find the total quantity of products sold.

RETAILSALES.PUBLIC Settings Code Versions

```
1 select SUM(QUANTITY) as total_quantity
2 FROM retailsales
3 group by product_category
4
5
6
```

# TOTAL_QUANTITY	
	771
	894
	849

Q13. Find the maximum Total Amount spent in a single transaction.

```
5 select max(total_amount) as max_total_amount
6 from retailsales
7 group by transaction_id
8
9
10
```

# MAX_TOTAL_AMOUNT	
1	150
2	1000
3	30
4	500
5	100
6	30
7	50
8	600
9	100

Q14. Find the minimum Price per Unit in the dataset.

```
9 select min(total_amount) as min_total_amount
10 from retailsales
11 order by total_amount DESC
12
```

# MIN_TOTAL_AMOUNT	
1	25

Q15. Find the number of transactions per Product Category.

RETAILSALES.PUBLIC Settings

```

1 SELECT COUNT (DISTINCT transaction_id)
2 AS Transaction_count, product_category
3 from retailsales
4 group by 2;

```

Results Chart

	# TRANSACTION_COUNT	PRODUCT_CATEGORY
1	307	Beauty
2	351	Clothing
3	342	Electronics

Q16. Find the total revenue (Total Amount) per gender. Expected output: Gender, Total Revenue.

RETAILSALES.PUBLIC Settings

```

6 select case
7 when total_amount > 1000 then 'high'
8 else 'low'
9 end as spending_low
10 from retailsales;

```

Results Chart

	SPENDING_LOW
1	low
2	low
3	low
4	low
5	low
6	low

Q17. Find the average Price per Unit per product category.

RETAILSALES.PUBLIC Settings

```

1 SELECT
2 PRODUCT_CATEGORY,
3 AVG (PRICE_PER_UNIT)
4 FROM RETAILSALES
5 GROUP BY PRODUCT_CATEGORY;

```

Results Chart

	PRODUCT_CATEGORY	# AVG (PRICE_PER_UNIT)
1	Beauty	184.055375
2	Clothing	174.287749
3	Electronics	181.900585

Q18. Find the total revenue per product category where total revenue is greater than 10,000.

RETAILSALES.PUBLIC ▾ Settings ▾

```
1 SELECT
2 PRODUCT_CATEGORY,
3 SUM (PRICE_PER_UNIT * QUANTITY) AS TotalRevenue
4 from retailsales
5 group by product_category
6 having SUM(PRICE_PER_UNIT * QUANTITY) > 10000;
```

Results Chart

	PRODUCT_CATEGORY	TOTALREVENUE
1	Beauty	143515
2	Clothing	155580
3	Electronics	156905

Q19. Find the average quantity per product category where the average is more than 2.

RETAILSALES.PUBLIC ▾ Settings ▾

```
1 SELECT AVG (QUANTITY) AS Average_quantity,
2 PRODUCT_CATEGORY
3 FROM RETAILSALES
4 GROUP BY PRODUCT_CATEGORY
5 HAVING AVG (QUANTITY) >2;
```

Results Chart

	AVERAGE_QUANTITY	PRODUCT_CATEGORY
1	2.511401	Beauty
2	2.547009	Clothing
3	2.482456	Electronics

Q20. Display a column called Spending Level that shows 'High' if Total Amount > 1000, otherwise 'Low'.

RETAILSALES.PUBLIC Settings

1
2
3
4
5

```
select gender,  
sum(total_amount) AS TOTAL_REVENUE  
FROM retailsales  
group by gender;
```

Results Chart

	GENDER	TOTAL_REVENUE
1	Male	223160
2	Female	232840