

## BUSINESS AD-HOC REQUEST

```
use retail_events_db;
select * from fact_events;
select * from dim_products;
select * from dim_stores;
select * from dim_campaigns;
```

/\*Question 1: Products with base price more than 500 and promotion type BOGOF\*/

```
select DISTINCT dp.product_code, dp.product_name
from dim_products as dp
JOIN fact_events as fte
ON dp.product_code = fte.product_code
where (fte.base_price > 500) and (fte.promo_type = "BOGOF");
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
product_code	product_name		
P08	Atiq_Double_Bedsheet_set		
P14	Atiq_waterproof_Immersion_Rod		

/\*Question 2 - generate a report that provides number of stores in a city. Arrange it in descending order\*/

```
select city, count(*) as total_stores from dim_stores
group by city
order by total_stores DESC;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
city	total_stores		
Bengaluru	10		
Chennai	8		
Hyderabad	7		
Coimbatore	5		
Visakhapatnam	5		
Madurai	4		
Mysuru	4		
Mangalore	3		
Trivandrum	2		
Vijayawada	2		

Result 22 x

/\*Question 3 Campaign name and total revenue generation before promotion and after promotion\*/

```
SELECT dc.campaign_name,
```

```

sum(round(fte.base_price*fte.`quantity_sold(before_promo)`,2))/1000000 as
total_revenue_before_promo,
sum(round(fte.base_price*fte.`quantity_sold(after_promo)`,2))/1000000 as
total_revenue_after_promo
FROM fact_events as fte
JOIN dim_campaigns as dc
ON fte.campaign_id = dc.campaign_id
GROUP BY dc.campaign_name
ORDER BY dc.campaign_name ASC;

```

	campaign_name	total_revenue_before_promo	total_revenue_after_promo
▶	Diwali	82.5738	207.4562
	Sankranti	58.1274	140.4039

/\*Question 4 -Calculation of ISU (Incremental Sold Quantity) for each category.

Ranking category based on ISU% \*/

```

SELECT p.category,
(round((SUM(fte.`quantity_sold(after_promo)`) -
SUM(fte.`quantity_sold(before_promo)`))/
SUM(fte.`quantity_sold(before_promo)`) *100,2)) as ISU_percentage,
RANK() OVER (order by ((SUM(fte.`quantity_sold(after_promo)`) -
SUM(fte.`quantity_sold(before_promo)`))/
SUM(fte.`quantity_sold(before_promo)`) *100) DESC) as rank_order
FROM dim_products as p
JOIN fact_events as fte
ON p.product_code = fte.product_code
WHERE fte.campaign_id = 'CAMP_DIW_01'
group by p.category;

```




	category	ISU_percentage	rank_order
▶	Home Appliances	244.23	1
	Combo1	202.36	2
	Home Care	79.63	3
	Personal Care	31.06	4
	Grocery & Staples	18.05	5

/\*Question 5 - Generate a query with top 5 products, ranked by IR %.  
Report should have product name, category and ir%\*/

```

SELECT  p.product_name, p.category,
ROUND((SUM(fte.`quantity_sold(after_promo)` * fte.base_price) -
SUM(fte.`quantity_sold(before_promo)` * fte.base_price))/
SUM(fte.`quantity_sold(before_promo)` * fte.base_price)*100,2) AS
IR_Percentage
from dim_products as p
JOIN fact_events as fte
ON p.product_code = fte.product_code
group by p.product_name, p.category
Order by IR_percentage DESC
LIMIT 5;

```

Result Grid     Filter Rows: <input type="text"/>   Export:    Wrap Cell Content: <input type="checkbox"/>			
	product_name	category	IR_Percentage
▶	Atliq_waterproof_Immersion_Rod	Home Appliances	266.19
	Atliq_High_Glo_15W_LED_Bulb	Home Appliances	262.98
	Atliq_Double_Bedsheet_set	Home Care	258.27
	Atliq_Curtains	Home Care	255.34
	Atliq_Home_Essential_8_Product_Combo	Combo1	183.33