Answer File

1. **Finding Active Branches list**

```
SELECT * FROM branch_data
WHERE branch_status = 'Active';
```

2. **Fetching Basic Product Details from the table**

```
SELECT Product_ID, Product_name, Product_type, Product_size, Price
FROM products;
```

3. **Recent Customer Signups**

```
SELECT Customer_ID, Customer_name, SignUp_timestamp
FROM customers_info
WHERE SignUp_timestamp >= DATE_SUB(NOW(), INTERVAL 2 YEAR);
```

4. **Finding the Orders Overview**

```
SELECT Orders_ID, Customer_ID, Order_timestamp, Payment_status
FROM orders_info;
```

5. **Unresolved Customer Support Tickets**

```
SELECT ticket_id, customer_id, date_timeofcontact, ticket_status
FROM customer_support
WHERE ticket_status <> 'Resolved';
```

6. **Finding Customers Order info**

```
SELECT o.Orders_ID, o.Order_timestamp, c.Customer_name, c.Email_address
FROM orders_info o
```

```
JOIN customers_info c ON o.Customer_ID = c.Customer_ID;
```

7. **Total Revenue**

```
SELECT ROUND(SUM(Bill_Amount), 2) AS total_revenue
FROM finance_transactions;
```

8. **Average Response Time**

```
SELECT contact_channel, ROUND(AVG(time_taken_in_sec), 2) AS avg_respon
se_time_sec
FROM customer_support
GROUP BY contact_channel;
```

9. **Finding Active Employee count**

```
SELECT Department, COUNT(Emp_ID) AS employee_count
FROM employee_data
GROUP BY Department;
```

10. **Finding the Expired Products from Inventory**

```
SELECT Item_id, Item_Name, expiry_date
FROM inventory
WHERE expiry_date < NOW();
```

11. **Finding the campaigns by channel**

```
SELECT Channel, COUNT(Campaign_ID) AS campaign_count
FROM marketing_campaigns
GROUP BY Channel;
```

12. **Finding the Active Assets**

```
SELECT b.Branch_name, COUNT(a.asset_id) AS asset_count
FROM assets a
JOIN branch_data b ON a.deployed_branch_id = b.Branch_ID
GROUP BY b.Branch_name;
```

13. **Top 5 Products by revenue**

```
SELECT p.Product_name, ROUND(SUM(f.Bill_Amount), 2) AS product_revenue
FROM finance_transactions f
JOIN products p ON f.Product_ID = p.Product_ID
GROUP BY p.Product_name
ORDER BY product_revenue DESC
LIMIT 5;
```

14. **Asset Depreciation**

```
SELECT
  asset_id,
  asset_name,
  asset_cost,
  depreciation_percent,
  deployment_date,
  TIMESTAMPDIFF(YEAR, deployment_date, CURDATE()) AS years_used,
  ROUND(
    asset_cost * (1 - ((depreciation_percent / 100) * TIMESTAMPDIFF(YEAR, de
ployment_date, CURDATE()))),
    2
  ) AS current_value
FROM assets;
```

15. **Employees by Department**

```
SELECT *
FROM (
```

```
  SELECT Emp_ID, Employee_name, Department, Current_Annual_Cost,
      RANK() OVER (PARTITION BY Department ORDER BY Current_Annual_Cost
DESC) AS rank_in_dept
  FROM employee_data
) sub
WHERE rank_in_dept = 1;
```

16. **ROI from Active branches**

```
WITH branch_revenue AS (
  SELECT o.Branch_ID, SUM(f.Bill_Amount) AS total_revenue
  FROM orders_info o
  JOIN finance_transactions f ON o.Orders_ID = f.Orders_ID
  GROUP BY o.Branch_ID
)

SELECT
  b.Branch_ID, b.Branch_name, b.Branch_Investment_cost,
  r.total_revenue,
  ROUND(r.total_revenue / b.Branch_Investment_cost, 2) AS ROI_ratio
FROM branch_data b
LEFT JOIN branch_revenue r ON b.Branch_ID = r.Branch_ID
WHERE b.Branch_status = 'Active';
```

18. C**ustomer order frequency**

```
-- Step 1: Calculate delivery duration per delivery
WITH delivery_times AS (
  SELECT
    Delivery_ID,
    Delivery_by,
    TIMESTAMPDIFF(MINUTE, Delivery_start_timestamp, Delivery_end_timesta
mp) AS delivery_duration
  FROM deliveries
  WHERE Delivery_status = 'Delivered'
```

```
),

-- Step 2: Get average delivery time per delivery type (Direct / Partner)
avg_by_type AS (
  SELECT
    Delivery_by,
    ROUND(AVG(delivery_duration), 2) AS avg_duration
  FROM delivery_times
  GROUP BY Delivery_by
),

-- Step 3: Identify deliveries that took longer than average
delayed_deliveries AS (
  SELECT
    dt.Delivery_by,
    dt.delivery_duration,
    abt.avg_duration,
    dt.delivery_duration - abt.avg_duration AS delay_minutes
  FROM delivery_times dt
  JOIN avg_by_type abt ON dt.Delivery_by = abt.Delivery_by
  WHERE dt.delivery_duration > abt.avg_duration
)

-- Final Output: Count of delays and average delay per delivery type
SELECT
  Delivery_by,
  COUNT(*) AS delayed_deliveries,
  ROUND(AVG(delay_minutes), 2) AS avg_delay_minutes
FROM delayed_deliveries
GROUP BY Delivery_by
ORDER BY delayed_deliveries DESC;
```

19. **Find Top 3 products based on their price**

```
SELECT *
FROM (
  SELECT
    Product_ID, Product_name, Product_type, Price,
    RANK() OVER (PARTITION BY Product_type ORDER BY Price DESC) AS pric
e_rank
  FROM products
) ranked
WHERE price_rank <= 3;
```

20. F**inding the Inventory Expiry tracker**

```
SELECT
  Item_id, Item_name, expiry_date,
  DATEDIFF(expiry_date, CURDATE()) AS days_left
FROM inventory
WHERE expiry_date BETWEEN CURDATE() AND DATE_ADD(CURDATE(), INTE
RVAL 10 DAY)
ORDER BY days_left;
```