

## Example 1: Simple Filter

Description:

```
{  
    "description": "Which employees were hired after 2019?"  
}
```

Result:

```
{  
    "sql_query": "SELECT first_name, last_name FROM employee_dim WHERE hire_date > '2019-12-31' LIMIT 100",  
    "raw_data": [  
        {  
            "last_name": "Fischer",  
            "first_name": "Sophie"  
        },  
        {  
            "last_name": "Neumann",  
            "first_name": "Tom"  
        },  
        {  
            "last_name": "Fischer",  
            "first_name": "Sophie"  
        },  
        {  
            "last_name": "Neumann",  
            "first_name": "Tom"  
        }  
    ]  
}
```

---

## Example 2: Advanced Analytics

Description:

```
{  
    "description": "For each employee, show monthly total sales,  
    their rank within the department, and the department average  
    sales"  
}
```

Result:

```

SUM(s.total_amount) as total_sales\n
ee_id, e.department, e.first_name, e.last_\n
e,\n      year, \n          month,\n
avg_department_sales AS (\n      SELECT \r
month\n    )\n    SELECT \n        rs.first_\nFROM ranked_sales rs\n    JOIN avg_departm_\nes_rank\n    LIMIT 100",
"raw_data": [
{
  "year": 2025,
  "month": 11,
  "avg_sales": 811772.01875,
  "last_name": "Schmidt",
  "department": "Sales",
  "first_name": "Anna",
  "sales_rank": 1,
  "total_sales": 1284574.46
},
{
  "year": 2025,
  "month": 11,
  "avg_sales": 811772.01875,
  "last_name": "Müller",
  "department": "Sales",
  "first_name": "Max",
  "sales_rank": 2,
  "total_sales": 1163814.19
}
],

```

---

### Example 3: Comparison

Description:

```
{
  "description": "Compare the sales of employees this month
compared to last month."
}
```

Result:

```
{
  "sql_query": "WITH \n      -- Get current and last month sales\n      _amount) AS total_sales\n          FROM \n              sales s\n      WHERE \n              d.month = EXTRACT(MONTH FROM CURRENT_DATE)\n      UNION ALL\n      SELECT \n          e.first_name || ' ' || e.last_name\n      FROM \n          employee_dim e\n      ON s.employee_id = e.employee_id\n      WHERE \n          EXTRACT(YEAR FROM CURRENT_DATE) = e.dim_year\n      OR (d.month = 12\n          AND EXTRACT(YEAR FROM CURRENT_DATE) = e.dim_year + 1)\n      ORDER BY \n          _amount DESC\n      LIMIT 10\n      \n      -- Compare sales\n      SELECT \n          e.first_name || ' ' || e.last_name AS employee_name,\n          COALESCE(lms.total_sales, 0) AS last_month_sales,\n          COALESCE(lms.current_sales, 0) AS current_month_sales,\n          COALESCE(lms.sales_diff, 0) AS sales_diff\n      FROM \n          last_month_sales lms\n      ON cms.employee_name = lms.employee_name\n      ORDER BY \n          sales_diff\n      LIMIT 10",
  "raw_data": [
    {
      "sales_diff": -117609.27,
      "employee_name": "Sophie Fischer",
      "last_month_sales": 1456362.84,
      "current_month_sales": 1338753.57
    },
    {
      "sales_diff": -313215.92,
      "employee_name": "Anna Schmidt",
      "last_month_sales": 1529896.17,
      "current_month_sales": 1216680.25
    },
    {
      "sales_diff": -512428.25,
      "employee_name": "Max Müller",
      "last_month_sales": 1718383.26,
      "current_month_sales": 1205055.01
    }
  ]
}
```

#### Example 4: Cumulative Analysis

Description:

```
{
  "description": "Calculate the accumulated sales per product this month by day."
}
```

Result:

```
{  
  "sql_query": "WITH daily_sales AS (\n      SELECT  
      product_dim p ON s.product_id = p.product_id\n      = EXTRACT(MONTH FROM CURRENT_DATE)\n      GROUP BY product_name ORDER BY date_id) as accumulated_sales\n  \"raw_data\": [  
    {  
      \"date_id\": \"2026-01-07\",  
      \"product_name\": \"Accord\",  
      \"accumulated_sales\": 35412.88  
    },  
    {  
      \"date_id\": \"2026-01-09\",  
      \"product_name\": \"Accord\",  
      \"accumulated_sales\": 110593.16  
    },  
    {  
      \"date_id\": \"2026-01-13\",  
      \"product_name\": \"Accord\",  
      \"accumulated_sales\": 184189.59  
    },  
    {  
      \"date_id\": \"2026-01-15\",  
      \"product_name\": \"Accord\",  
      \"accumulated_sales\": 219357.16  
    },  
    {  
      \"date_id\": \"2026-01-22\"  
    }  
  ]  
}
```

---

### Example 5: Hierarchy Performance

Description:

```
{  
  \"description\": \"Analyze sales performance by department, team,  
  and employee and show percentage contribution at each level.\"  
}
```

Result:

```
{  
    "sql_query": "WITH sales_performance AS  
AS total_sales\n        FROM sales_fact s\ns AS (\n            SELECT \n                department,  
department,\n                    team,\n                        SUM(total_sales) AS  
sp.employee,\n                            sp.total_sales,\n                                ntribution,\n                                    ROUND(sp.total_sales * .02) AS  
p.department = ds.department\n        JOIN team  
    "raw_data": [  
        {  
            "team": "Salesperson",  
            "employee": "Anna Schmidt",  
            "department": "Sales",  
            "total_sales": 4887028.72,  
            "team_contribution": 27.76,  
            "overall_contribution": 27.76,  
            "department_contribution": 27.76  
        },  
        {  
            "team": "Salesperson",  
            "employee": "Max Müller",  
            "department": "Sales",  
            "total_sales": 4787609.36,  
            "team_contribution": 27.2,  
            "overall_contribution": 27.2,  
            "department_contribution": 27.2  
        },  
        {  
            "team": "Salesperson",  
            "employee": "Sarah Johnson",  
            "department": "Marketing",  
            "total_sales": 3500000.0,  
            "team_contribution": 20.0,  
            "overall_contribution": 20.0,  
            "department_contribution": 20.0  
        }  
    ]  
}
```

---

### Example 6: Creative Llama

Description:

```
{  
    "description": "Tell me something creative about my data."  
}
```

Result:

```
},
{
  "name": "Max Müller",
  "value": 4787609.36,
  "category": "Top Earning Employees"
},
{
  "name": "Tom Neumann",
  "value": 4069241.4,
  "category": "Top Earning Employees"
},
{
  "name": "Sophie Fischer",
  "value": 3857985.87,
  "category": "Top Earning Employees"
},
{
  "name": "Model S",
  "value": 4594493.79,
  "category": "Top Selling Products"
},
{
  "name": "Mustang",
  "value": 2496466.15,
  "category": "Top Selling Products"
},
/
```