

Case Study : Employee Data Warehouse and Data Mart

Business Intelligence Batch 9



This event has live translations.

Presented by:
Anniza Mega.

Date Presented:
February 23th, 2024

Agenda

- Introduction
- Data Warehouse & Data Mart
- Tools & Source
- Objective
- Problem Statment
- ERD Diagram
- Dimensions & Fact
- Data Mart Analyzing
- Recomendation



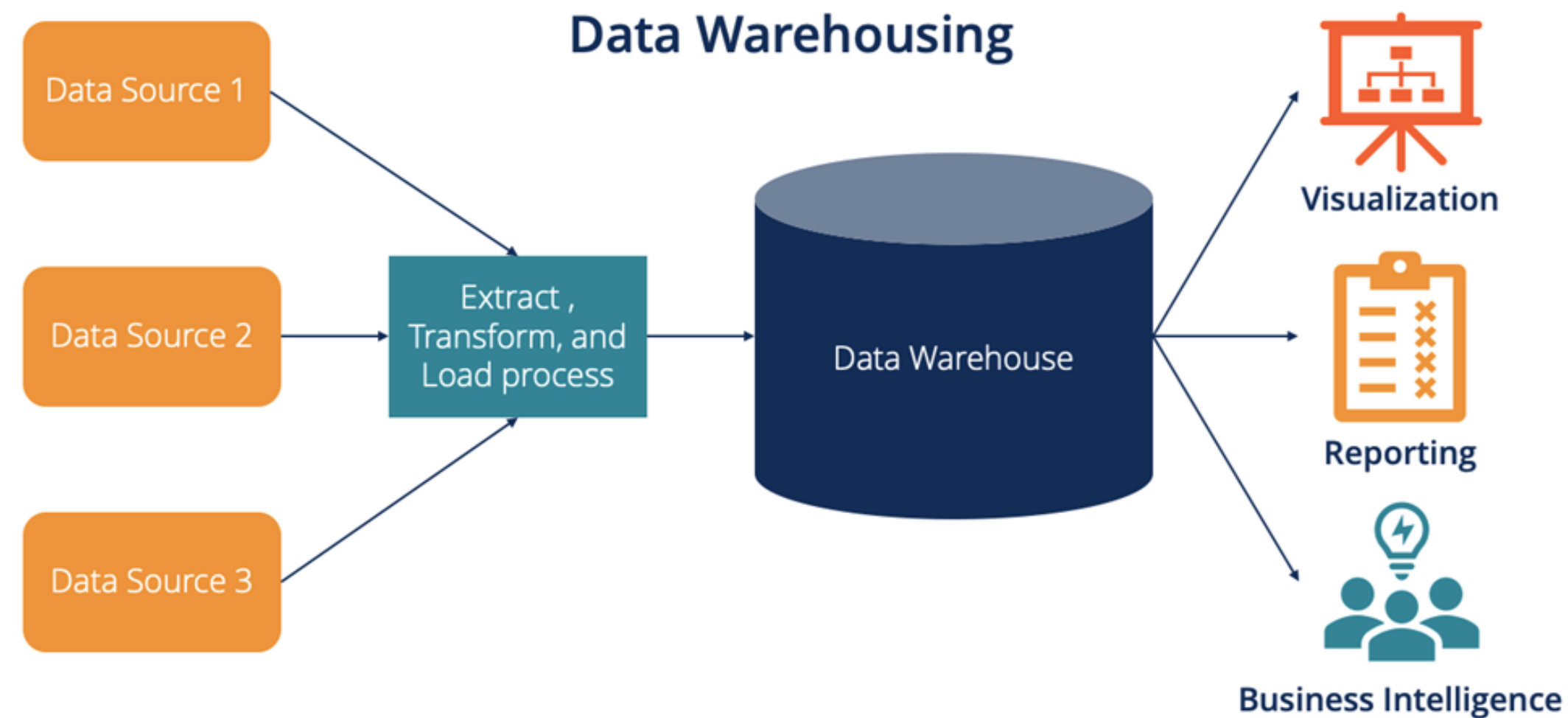
Introduction

Hello! I'm a student in Business Intelligence Bootcamp Batch 9 at Dibimbing.id. I'm excited to share my case study with you. I used Draw SQL and PostgreSQL to analyze data employee of the company. I construct an Entity-Relationship Diagram (ERD) for data warehouse. The primary objective of this analysis is to offer strategic recommendations for improving overall performance levels within the company.

Data Warehouse

A data warehouse is a system that aggregates data from one or more sources into single consistent data store to support data analytics.

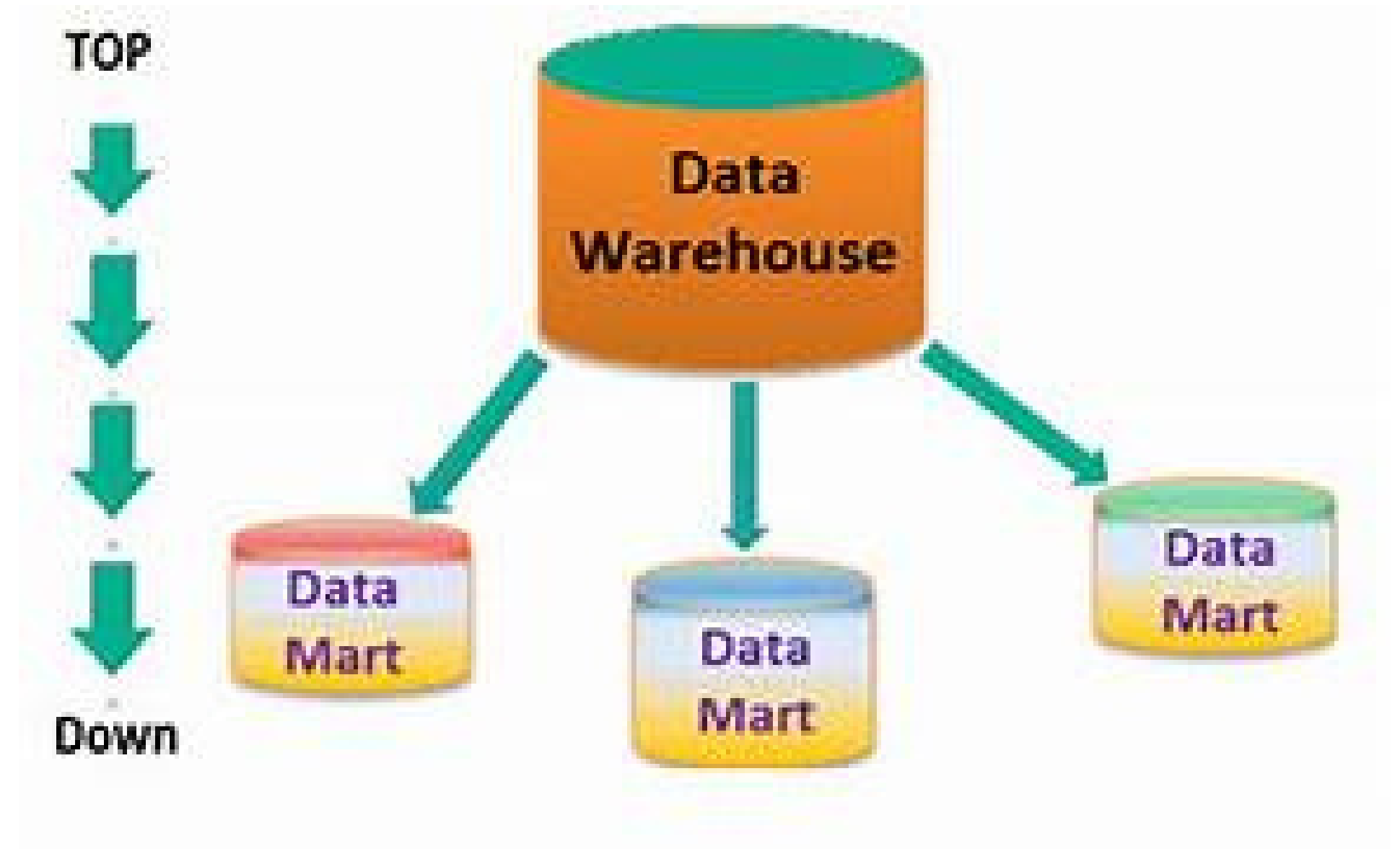
Source: oracle.com



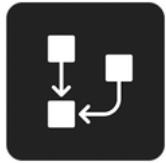
Data Mart

A data mart is a simple form of data warehouse focused on a single subject or line of business. With a data mart, teams can access data and gain insights faster, because they don't have to spend time searching within a more complex data warehouse or manually aggregating data from different sources.

source : [oracle.com](https://www.oracle.com)



Tools & Sources



DrawSQL :

Design a data warehouse (DWH) schema for the Employee domain. Create an Entity-Relationship Diagram (ERD) representing the schema and describe it using the star schema.

POSTGRES SQL

In this project I used PostgreSQL as a data warehouse tool. This will involve the query of SQL and determining of the dimensions of the tables.



EXCEL & CANVA :

Excel used for creating diagram for simple visualization
Canva used for creating presentation



Objective and Requirement

Objective :

Design a data warehouse (DWH) schema for the Employee domain. Create an Entity-Relationship Diagram (ERD) representing the schema and describe it using either the star schema or snowflake schema. Provide detailed descriptions of the tables and columns involved. Finally, create three sample queries (Data Mart tables) from the DWH schema.

Requirement :

ERD Diagram

- Create an ERD representing the chosen dataset's schema.
- Clearly illustrate the relationships between tables.

Schema Description:

- Describe the schema using either star schema or snowflake schema.
- Provide explanations for each table, highlighting the primary key, foreign keys, and any important attributes.

Sample Queries (Data Mart Tables):

- Create three Data Mart tables (sample queries) based on the designed schema.
- Each Data Mart table should serve a specific analytical or reporting purpose.
- Include SQL queries to generate these Data Mart tables.

[Back to Agenda](#)

Problem Statement

01

Create ERD Diagram

Create an ERD representing the chosen dataset's schema with clearly illustrate the relationships between tables.

02

Schema Description

Provide explanations for each table, highlighting the primary key, foreign keys, and any important attributes.

03

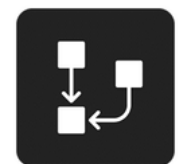
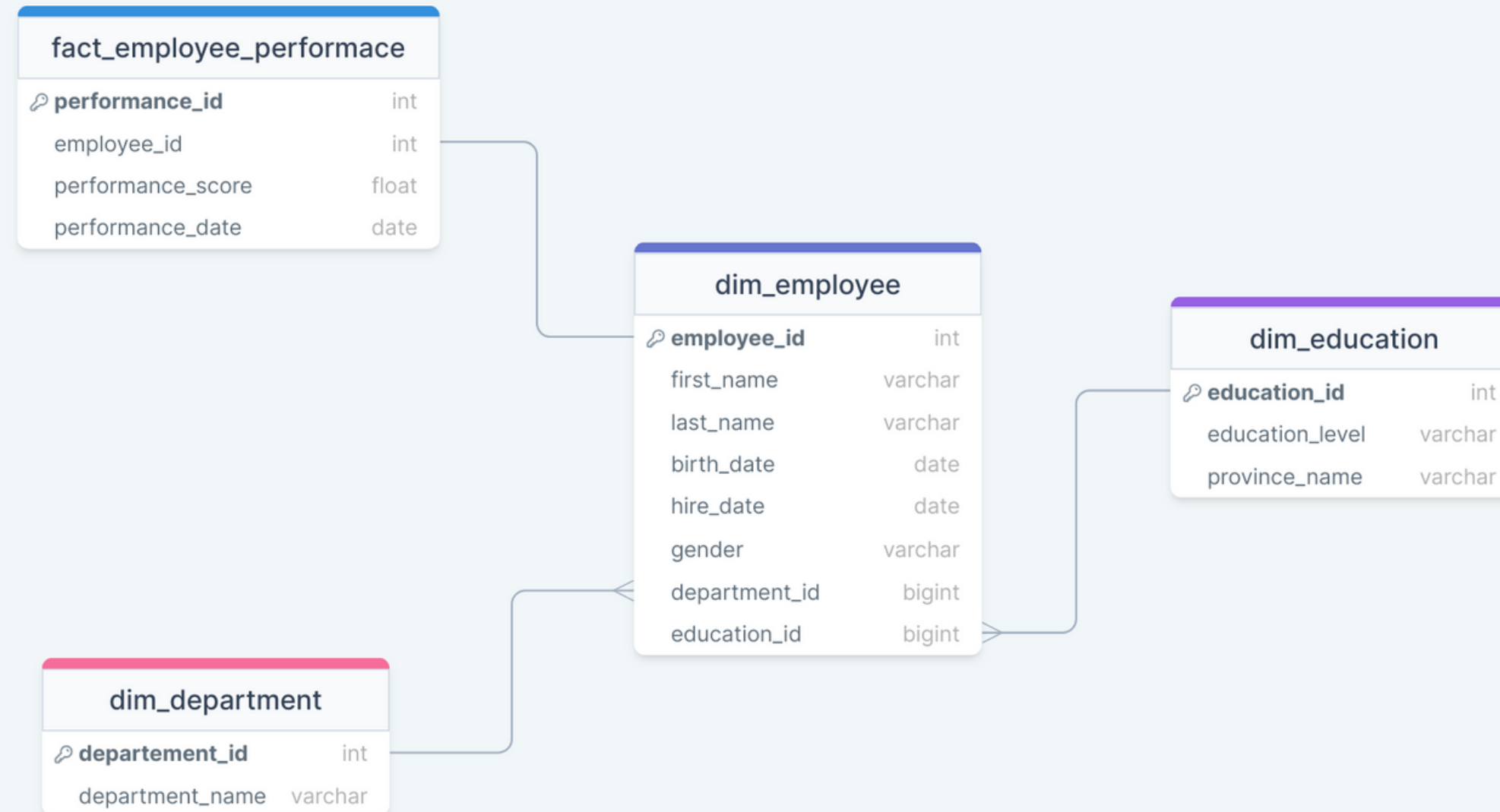
Create Sample Queries (Data Mart Tables)

Create three Data Mart, serve a specific analytical or reporting purpose, add SQL queries to generate these Data Mart tables.

ERD Diagram

Employee Data
Warehouse

Star Schema

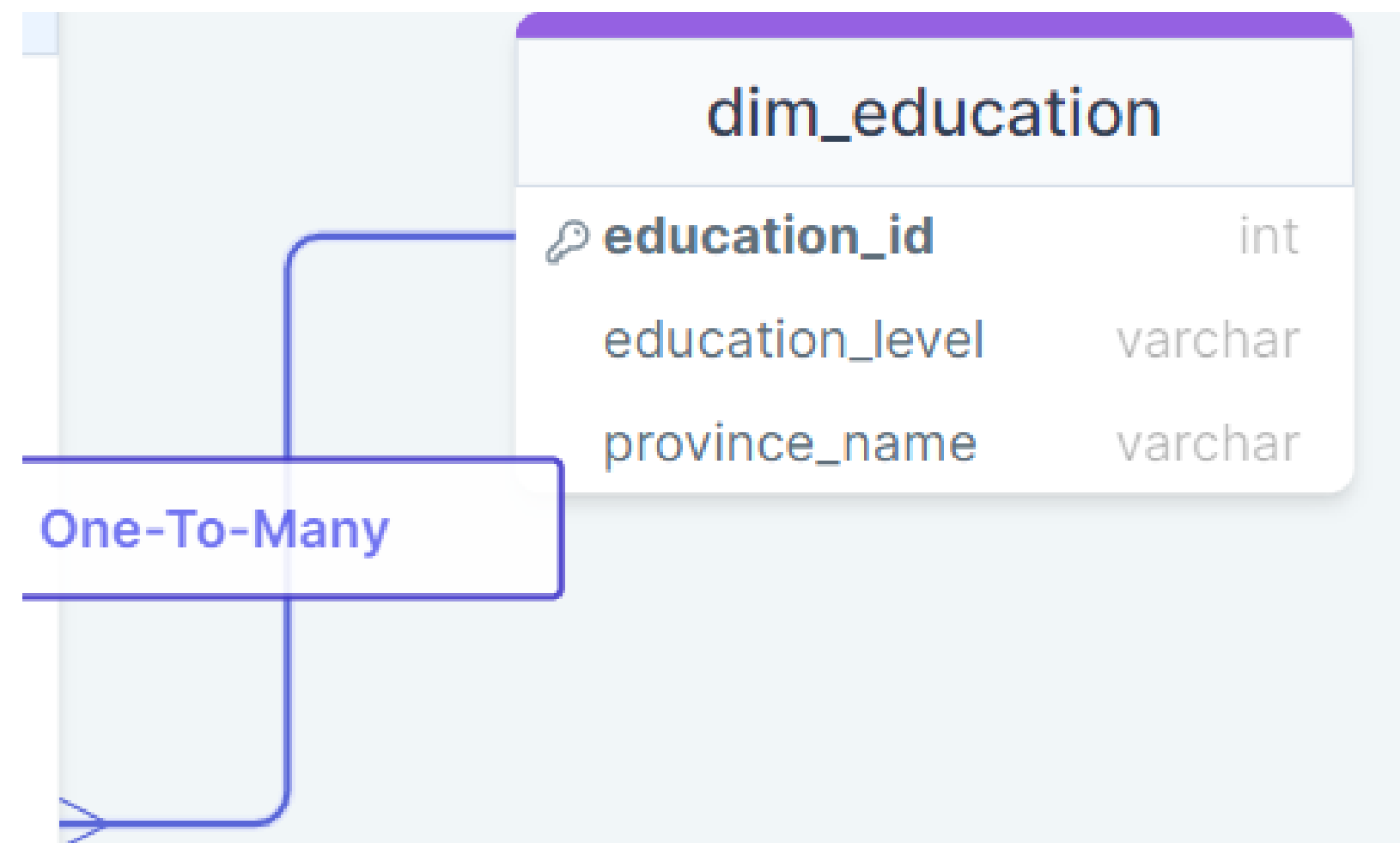
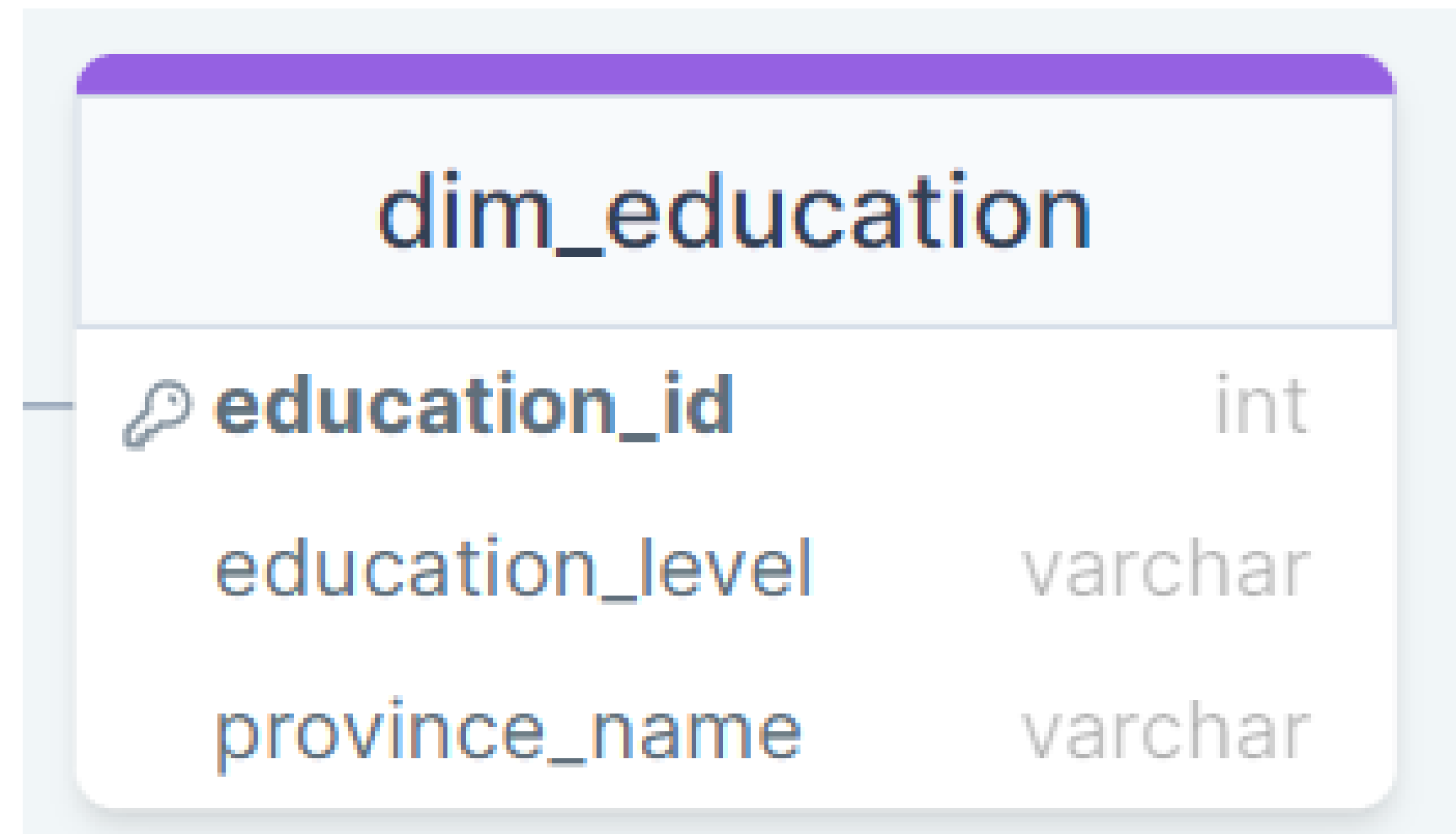


The ERD in Employee data warehouse consists of 1 fact table and 3 dimension tables with a star schema

Dimension & Fact

Dimension Education

- **education_id** (Primary Key): This is a unique identifier within the table, serving as the primary means of identifying each education record. It's of type integer.
- **education level**: This attribute represents the level of education attained by individuals, with possible values including high school, associate degree, master's degree, and PhD. It's of type varchar.
- **Foreign Key**: The education_id serves as a foreign key referencing the dim_employee table. The relationship from dim_education to dim_employee is described as one-to-many, indicating that one education record can be associated with multiple employee records.




Dimension & Fact


Fact Employee Performance

- **performance_id** (Primary Key): This is an integer value serving as the primary key, representing a unique identifier for each performance evaluation record. The value varies depending on the time of the evaluation.
- **employee_id**: This attribute is an integer representing a unique identifier for each employee. It's linked to the dim_employee table, indicating a one-to-one relationship between fact_employee_performance and dim_employee.
- **performance_score**: This attribute represents the performance score of an employee and is of decimal type.
- **performance_date**: This attribute denotes the date when the employee's performance was evaluated. It's of date type, providing information about the timing of the evaluation.

fact_employee_performance

 performance_id	int
employee_id	int
performance_score	float
performance_date	date

fact_employee_performance

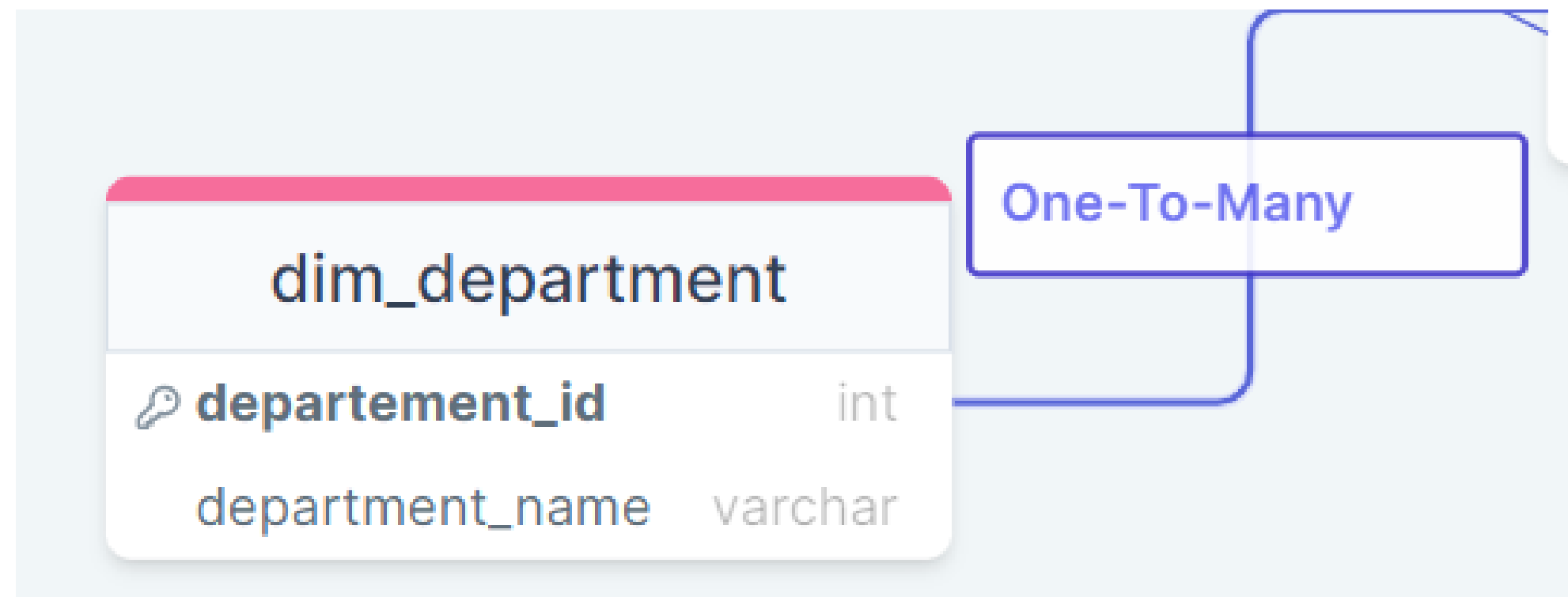
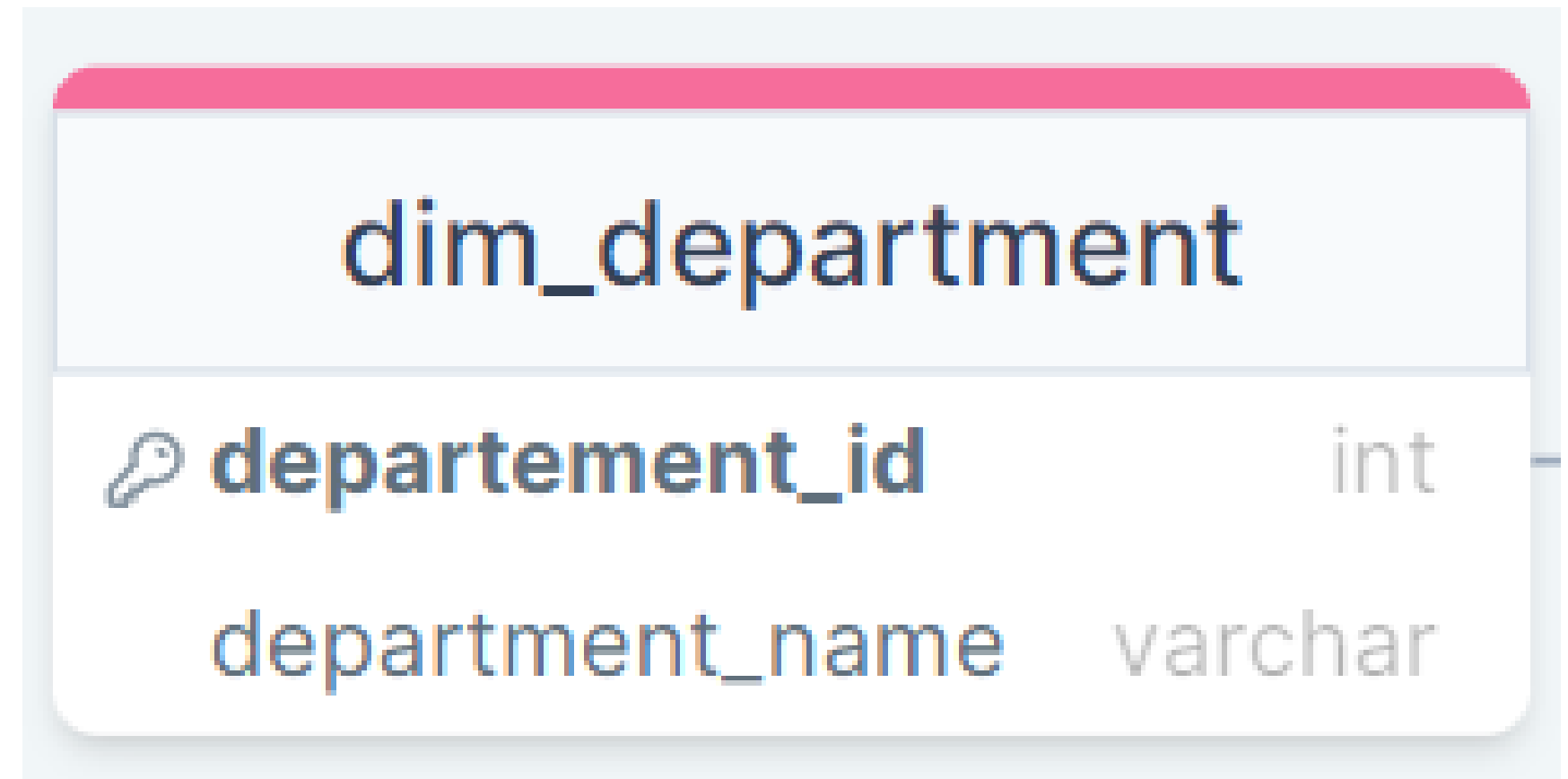
 performance_id	int
employee_id	int
performance_score	float
performance_date	date

One-To-One

Dimension & Fact

Dimension Department


- **Primary key** : department_id (unique identifier for each department, of type integer) refferencing the dim_employee table. The relationship from dim_department to dim_employee is described as one to many indicating that one department id record can be assosiated with multiple employee record.
- **department_name** : foreign key with data type varchar

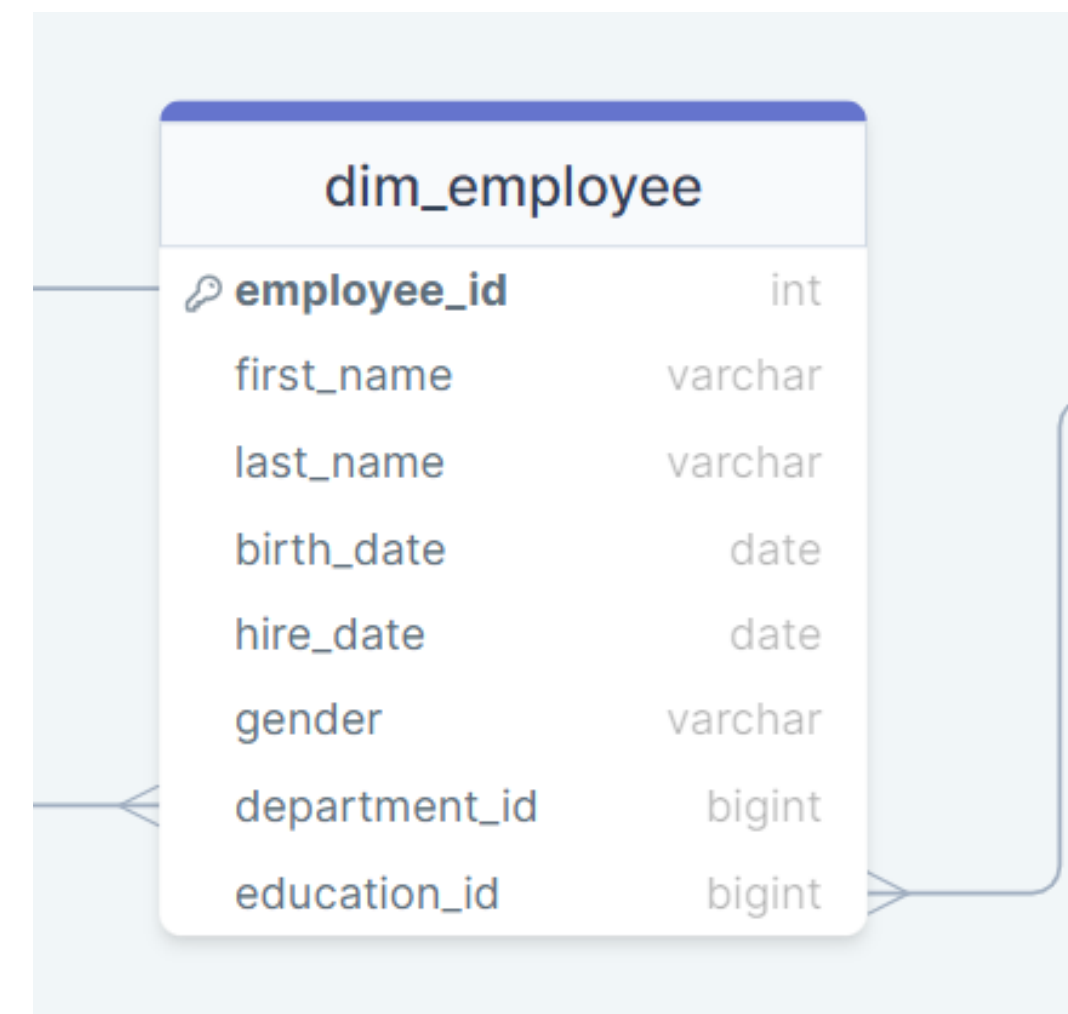


Dimension & Fact

Dimension Employee

- **employee_id (Primary Key):** This is a unique identifier within the table, serving as the primary means of identifying each employee id. It's of type integer.
- **Foreign Key:** department_id serves as a foreign key referencing the dim_education table. The relationship from dim_employee to dim_education is described as -many to one , indicating that one education record can be associated with multiple employee records.
- **Attribute :** first_name, last_name, birt_date, hire_date, gender, department_id, and education_id

dim_employee	
 employee_id	int
first_name	varchar
last_name	varchar
birth_date	date
hire_date	date
gender	varchar
department_id	bigint
education_id	bigint



Data Mart Analyzing

An Employee's performance created to analyze employee who has a bad or good or excellent performance during works on the Company.

[Back to Agenda](#)



SQL CODE

Code to analyze the employee's bad performance by performance score under 75

```
-- Data Mart Employee's bad performance with Performance score < 75
CREATE VIEW public.performace_employee_bad AS
SELECT
    f.performance_id,
    f.employee_id,
    f.performance_score,
    f.performance_date,
    e.first_name,
    e.last_name,
    e.birt_date,
    e.hire_date,
    e.gender,
    e.department_id,
    d.department_name,
    e.education_id,
    ed.education_level
FROM
    fact_employee_performance f
LEFT JOIN
    dim_employee e ON f.employee_id = e.employee_id
LEFT JOIN
    dim_department d ON e.department_id = d.department_id
LEFT JOIN
    dim_education ed ON e.education_id = ed.education_id
WHERE f.performance_score < 75 ;
```

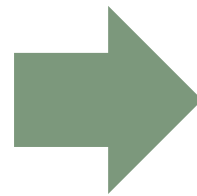


Table Result

Query

Query History

```
1 SELECT * FROM public.performace_employee_bad
2
```

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

	performance_id integer	employee_id integer	performance_score double precision	performance_date date	first_name character varying (50)	last_name character varying (50)	birth_date date
1	1	1	36.4784694903924	2022-10-13	Ignazio	Wharby	1990-01-01
2	2	2	38.40222708316501	2020-07-31	Oralla	Minnette	1990-01-01
3	3	3	61.0410367577725	2022-06-12	Uriah	Plowell	1990-01-01
4	4	4	15.068316886181066	2020-02-22	Jecho	Laraway	1990-01-01
5	5	5	54.08936348060487	2020-02-11	Nettle	Kindleysides	1990-01-01
6	7	7	45.78888591104575	2022-04-21	Annamaria	Camden	1990-01-01
7	8	8	48.15436852138222	2022-08-28	Hashim	Raper	1990-01-01
8	12	12	28.01159941513005	2021-06-15	Rudiger	Warlow	1990-01-01
9	13	13	45.386553647844565	2020-01-09	Allistir	Byron	1990-01-01
10	14	14	53.4976948478143	2020-12-27	Paquito	Naisey	1990-01-01
11	15	15	22.642974591848763	2022-09-19	Winne	Melloi	1990-01-01
12	16	16	6.8976309817160075	2022-10-14	Tamas	Alkins	1990-01-01

Total rows: 758 of 758

Query complete 00:00:00.423

Ln 1, Col 1

SQL CODE

Code to analyze the employee's good performance by performance score between 75 and 81

```
-- Data Mart Employee's good performance with Performance score > 75 < 81
CREATE VIEW public.performace_employee_good AS
SELECT
    f.performance_id,
    f.employee_id,
    f.performance_score,
    f.performance_date,
    e.first_name,
    e.last_name,
    e.birt_date,
    e.hire_date,
    e.gender,
    e.department_id,
    d.department_name,
    e.education_id,
    ed.education_level
FROM
    fact_employee_performance f
LEFT JOIN
    dim_employee e ON f.employee_id = e.employee_id
LEFT JOIN
    dim_department d ON e.department_id = d.department_id
LEFT JOIN
    dim_education ed ON e.education_id = ed.education_id
WHERE f.performance_score BETWEEN 75 AND 81 ;
```

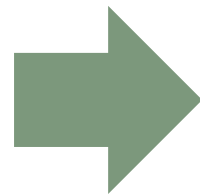


Table Result

Query Query History

```

1 SELECT * FROM public.performace_employee_good
2

```

Data Output Messages Notifications

	performance_id integer	employee_id integer	performance_score double precision	performance_date date	first_name character varying (50)	last_name character varying (50)	birt_da date
1	6	6	80.67333418394043	2022-03-24	Emmie	Simper	1995-
2	19	19	78.76723501250082	2021-03-26	Aldo	Riseam	1998-
3	32	32	79.35685971839135	2020-03-13	Ax	Monnelly	1995-
4	66	66	77.85367200014284	2021-03-21	Gage	Siemianowicz	1990-
5	68	68	76.79171556504343	2020-03-07	Poppy	Farrimond	1996-
6	89	89	79.38012640451468	2021-11-28	Alisun	Jeffer	1997-
7	100	100	77.73676383239967	2020-09-30	Gayleen	Spentley	1993-
8	133	133	77.62409652690778	2022-01-11	Kippy	Gingell	1991-
9	210	210	80.2689204081429	2022-06-02	Sisile	Tabourier	1996-
10	213	213	76.31564970766442	2021-04-09	Cullin	Holdron	1998-
11	216	216	76.22128917396867	2022-01-22	Lyssa	Gaveltone	1997-
12	243	243	77.87149639233645	2022-04-30	Antonius	Ives	1995-

Total rows: 50 of 50 Query complete 00:00:00.174 Ln 1, Col 1

SQL CODE

Code to analyze the employee's excellent performance by performance score more than 81

```
-- Data Mart Employee's excellent performance with Performance score > 81
CREATE VIEW public.performace_employee_excellent AS
SELECT
    f.performance_id,
    f.employee_id,
    f.performance_score,
    f.performance_date,
    e.first_name,
    e.last_name,
    e.birt_date,
    e.hire_date,
    e.gender,
    e.department_id,
    d.department_name,
    e.education_id,
    ed.education_level
FROM
    fact_employee_performance f
LEFT JOIN
    dim_employee e ON f.employee_id = e.employee_id
LEFT JOIN
    dim_department d ON e.department_id = d.department_id
LEFT JOIN
    dim_education ed ON e.education_id = ed.education_id
WHERE f.performance_score > 81 ;
```

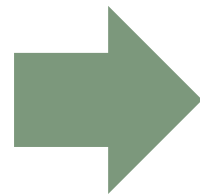


Table Result

Query

Query History

```

1 SELECT * FROM public.performace_employee_excellent
2

```

Data Output

Messages

Notifications

+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

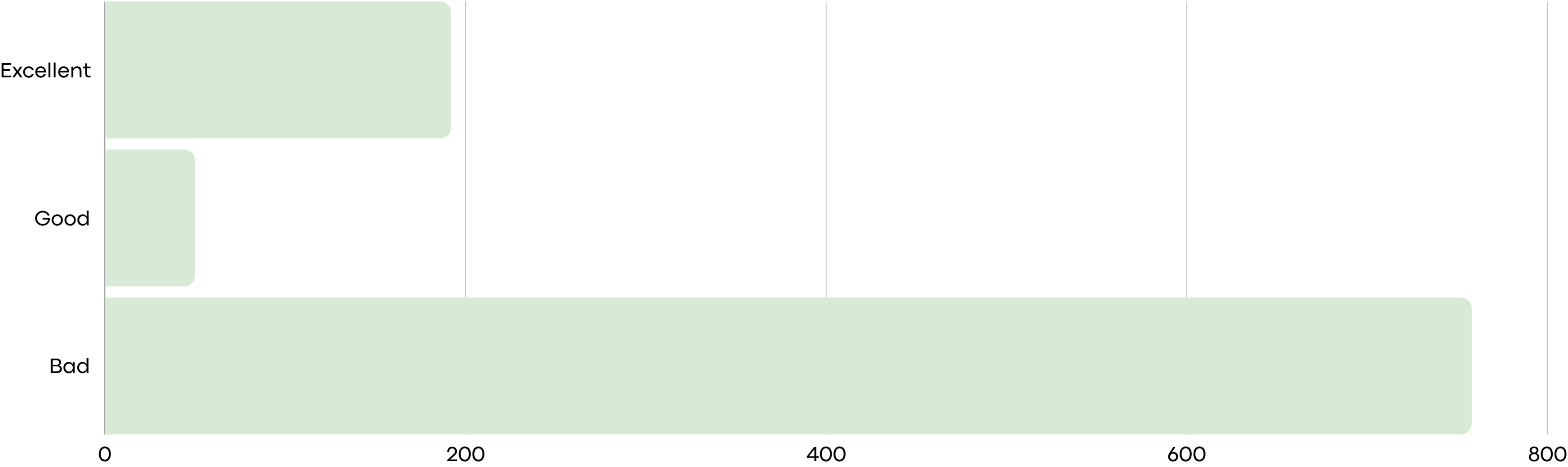
	performance_id integer	employee_id integer	performance_score double precision	performance_date date	first_name character varying (50)	last_name character varying (50)	birth_date date
1	9	9	94.05782915152795	2022-03-25	Ezequiel	Speedy	1996-01-01
2	10	10	95.32636516472932	2021-07-26	Bernard	Whatsize	1994-01-01
3	11	11	83.25610647569377	2020-11-26	Neils	Boland	1992-01-01
4	21	21	98.34831252250262	2020-10-26	Hillery	Parris	1992-01-01
5	23	23	86.62816016205352	2020-11-10	Cheslie	Skures	1994-01-01
6	45	45	99.12381793011704	2021-07-11	Jerri	Cholton	1994-01-01
7	49	49	90.65680676366286	2022-01-05	Rand	Gianninotti	1994-01-01
8	58	58	92.00939277912774	2020-09-10	Ajay	Bank	1992-01-01
9	76	76	94.87317576347156	2020-09-09	Sibyl	Garken	1998-01-01
10	82	82	97.37783692925015	2021-01-28	Frederik	Rolstone	1991-01-01
11	85	85	82.27825671836482	2020-04-12	Dorette	Crandon	1991-01-01
12	94	94	82.90274883642972	2022-12-17	Christiana	Abrahamsen	1992-01-01

Total rows: 192 of 192

Query complete 00:00:00.147

Ln 1, Col 1

Employee's Performance Score Analysis



Source: chart data mart .

[Back to Agenda](#)

Recommendations

Implement Performance Improvement Plans (PIPs):

- For employees with poor performance, initiate structured PIPs outlining clear expectations, goals, and timelines for improvement.
- Provide support, training, and regular feedback throughout the process.

Training and Development Opportunities:

- Invest in training and development programs to upskill employees across all performance levels.
- Offer targeted training sessions to address specific skill gaps identified in poor performers.

Recognition and Rewards Program:

- Recognize and reward employees who consistently deliver good and excellent performance.
- Implement a structured recognition program to acknowledge their contributions publicly, which can motivate others to strive for similar achievements.

Regular Performance Reviews and Feedback:

- Conduct regular performance reviews for all employees to provide constructive feedback and set performance goals.
- Use these reviews as an opportunity to identify challenges faced by poor performers and offer support to overcome them.

Thank You

*Data that provided in this project is purely based on hypothetical assumption and for illustrative purpose only

Get In Touch

Email

annizarmegas@gmail.com

Social Media

@annizamega