Name: Bhumika Patil
Classs: D15C
Roll no-61**.**

<div align="center"><u>**Practical No. 1**</u></div>

AIM: Design a Star Schema for the given system.

THEORY:

Introduction to Dimensional Modeling

Dimensional Modeling is a design technique used in data warehouse and business intelligence systems to structure data in a way that is easy to understand and fast to query for analytical purposes.

Unlike OLTP databases, which are optimized for transaction processing (inserts, updates, deletes), dimensional models are optimized for OLAP (Online Analytical Processing) — queries that aggregate, slice, dice, and drill down through data.

Key Features

- Denormalized structure: Data is stored in fewer tables to improve query performance.
- Two main components:
    1. Fact Tables – Store numeric measurements (facts) of the business.
    2. Dimension Tables – Store descriptive attributes that describe the facts.
- Schema designs such as Star Schema and Snowflake Schema.

Fact Table

- Central table in a dimensional model.
- Stores measurable, quantitative data (e.g., sales amount, quantity sold, length of stay).
- Contains foreign keys to link to dimension tables.
- Can store aggregated or transaction-level data.

Dimension Table

- Contains descriptive attributes that describe the facts.
- Usually textual and denormalized.
- Used for filtering, grouping, and labeling in queries.

Types of Dimensional Models

Dimensional modeling can be implemented using different schema types.

Star Schema

- Structure: One central fact table surrounded by dimension tables.
- Simplicity: Easy for users to understand.

- Performance: Fewer joins make queries faster.
- Usage: Most common schema in data warehouses.

Snowflake Schema

- Structure: Similar to Star Schema, but dimension tables are normalized into multiple related tables.
- Advantage: Saves storage space by reducing data redundancy.
- Disadvantage: More joins are needed, which can slow down queries.

Example:
Dim_Product split into Product → Product_Category → Product_Department.
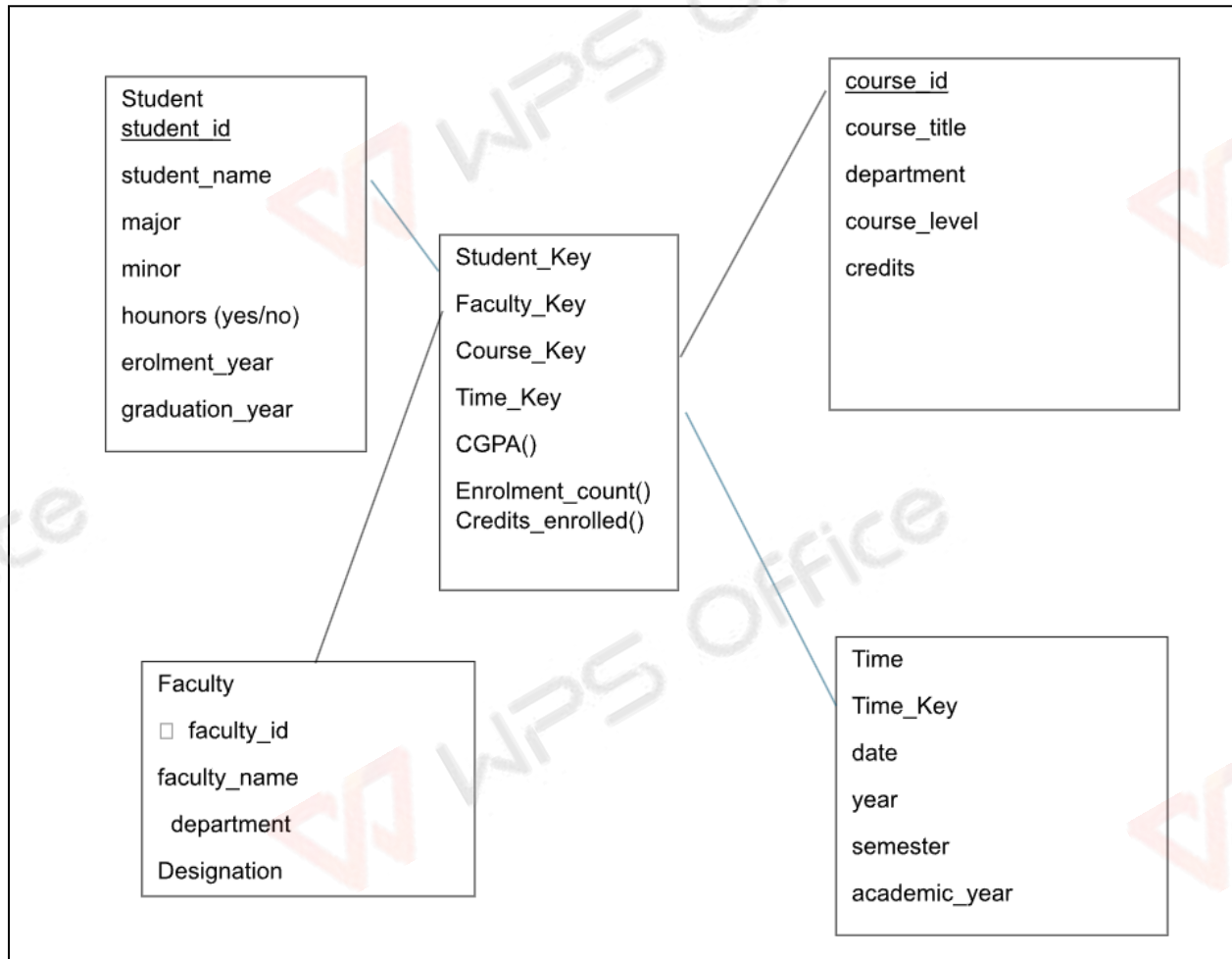
Advantages of Dimensional Modeling

- User-friendly for non-technical analysts.
- High performance for read-heavy queries.
- Supports historical data analysis.
- Scalable for growing business data.

Disadvantages

- Not suitable for transactional operations.
- Denormalization can lead to data redundancy.
- Updates to dimension data can be tricky (slowly changing dimensions).

Q1.A university wants to design a data warehouse to analyze student performance, course enrollments, and faculty workload. The university's operational database is highly normalized, making it difficult to perform analytical queries.

```
Student
student_id
student_name
major
minor
hounors (yes/no)
erolment_year
graduation_year
```

```
Student_Key
Faculty_Key
Course_Key
Time_Key
CGPA()
Enrolment_count()
Credits_enrolled()
```

```
course_id
course_title
department
course_level
credits
```

```
Faculty
☐ faculty_id
faculty_name
  department
Designation
```

```
Time
Time_Key
date
year
semester
academic_year
```

Q2.A retail company wants to analyze its sales performance across different regions, time periods, products, and customer segments. The company wants to track total sales, number of transactions, and discounts offered.

Date

Date_ID
Date
Day
Month
Qurter
Year

Region

Region_ID
Region_Name
State
Country
City

Fact Table

Sales_ID
Date_ID
Product_ID
Region_ID
Customer_ID
Total_Sales
Number_of_Transactions
Discount_offered

Product

Product_ID
Product_Name
Category
Sub_Category
Brand

Customer

Customer_ID
Segment_Name
Descriptiom
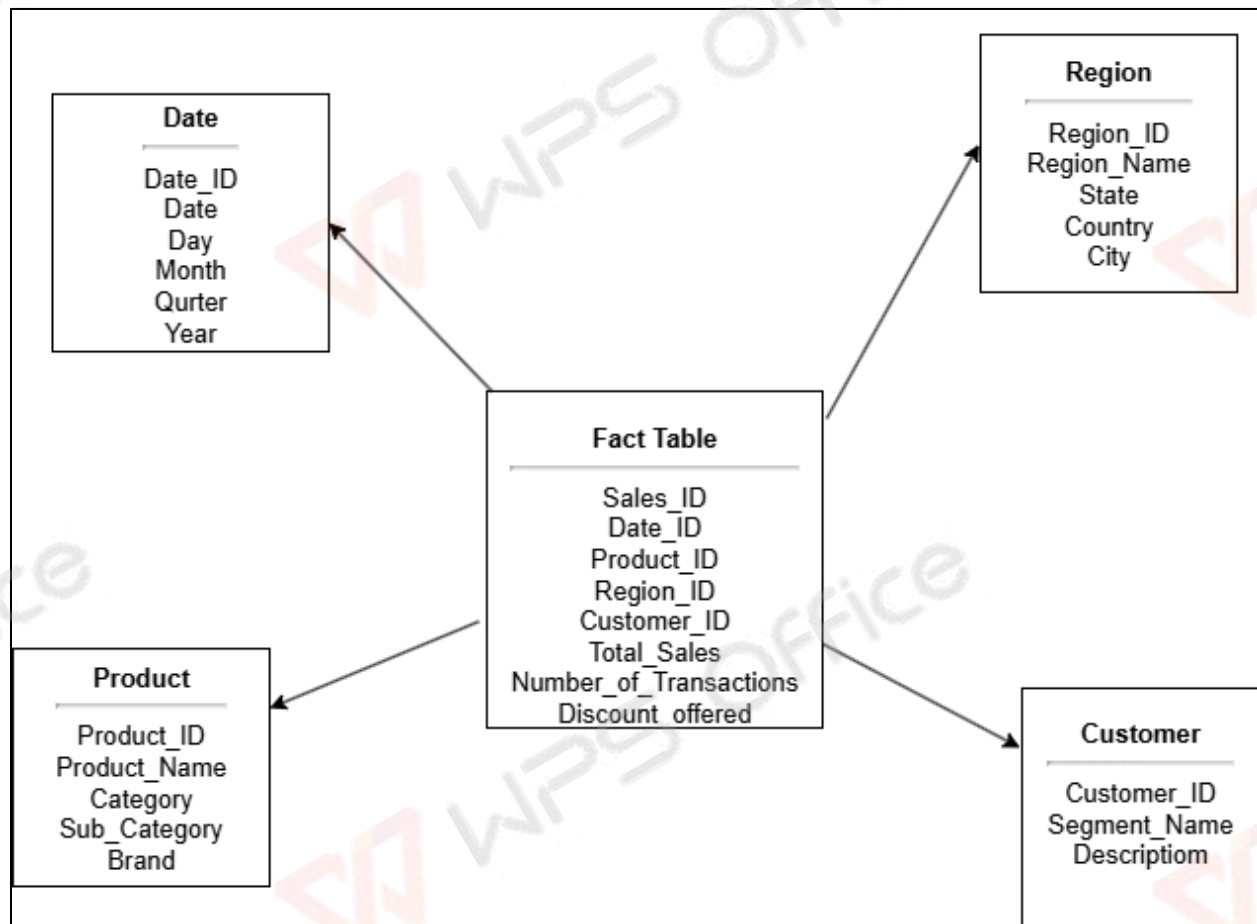
Q3.A hospital management wants to create a data warehouse to analyze patient admissions, procedures, and billing information. The goal is to improve operational efficiency and patient care by answering questions such as:

Dim_Patient

Patient_ID
Name
Gender
DOB
Contact
Address
Contact_No

Dim_Doctor

Doctor_ID
Name
Specialization
Contact_no

Dim_Department

Department_ID
Name
Location

Fact_Admissions

Admission_ID
Date_ID
Patient_ID
Diagnosis_ID
Doctor_ID
Department_ID
Procedure_ID
Billing_ID
Length_of_Stay
Total_Amount_Billed
Paid_Amount
Number_of_Procedures

Dim_Date

Date_ID
Day
Month
Quarter
Year
Day_of_Week

Dim_Diagnosis

Diagnosis_ID
Diagnosis_Code
Diagnosis_Name
Category

Dim_Procedure

Procedure_ID
Procedure_Name
Category

Dim_Billing

Billing_ID
Payment_Method
Insurance_Provider