

Introduction

Written by Daniel Loewito.

Database to manage employees within an organization, based on EDTS Technical Assessment.

Requirements Analysis

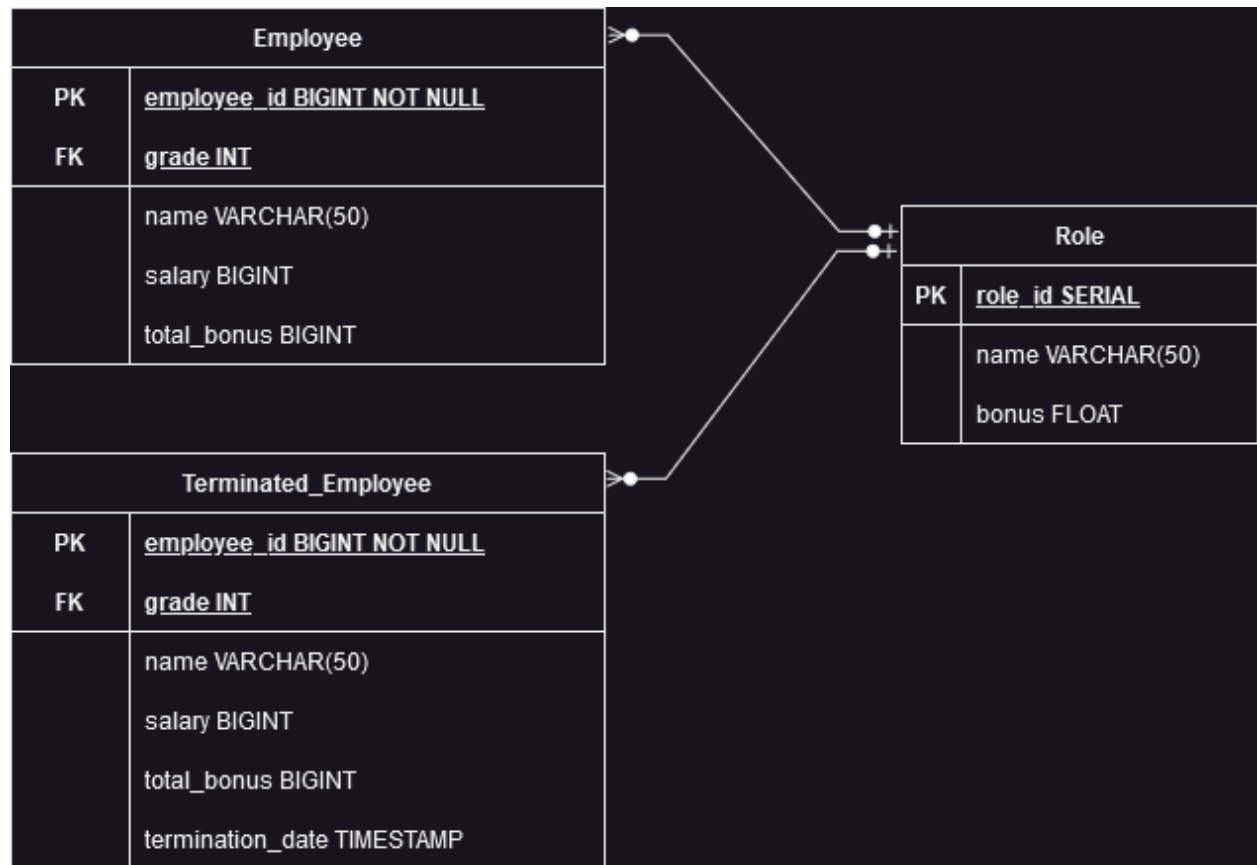
As per user stories in the spec, this database must be capable of:

1. Adding a new employee
2. Editing the details of an existing employee
3. View all employees with their bonuses

In addition to these requirements, I decided to add the following features to the database to increase functionality in real world use cases:

1. Automatic bonus calculation
2. Termination of an employee but maintaining their records

Conceptual Design



Normalization

Took the “Grade” field out of the example shown in the spec to reduce repetition. This allowed me to place a “bonus” field within the “grade” table, allowing for automatic calculation of bonuses in the event of inserts and updates in the “employee” and “grade” tables.

Constraints

Primary keys:

employee_id
role_id

Foreign keys:

grade in “employee” table reference **role_id** in “role” table

Functions:

calculate_bonus()

Calculates bonus in employee table for newly inserted or updated employees. Triggers before insert or update on “employee” table.

recalculate_employee_bonus()

Recalculates bonus for all employees with corresponding grade when there is a change in relevant data in the “role” table. Triggers after insert or update on “role” table.

move_terminated_employee()

Moves employees from “employee” table to “terminated_employee” table. Adds a timestamp for the date and time of termination. Triggers before delete on “employee” table.