

|  |
| --- |
| Business Template for Subway (Metro). |
|  |



Contents

[1 Business Description 3](#__RefHeading___Toc390_2862916375)

[1.1 Business background 3](#__RefHeading___Toc392_2862916375)

[1.2 Problems. Current Situation 3](#__RefHeading___Toc394_2862916375)

[1.3 the Benefits of implementing a database. Project Vision 3](#__RefHeading___Toc396_2862916375)

[2 Model description 4](#__RefHeading___Toc398_2862916375)

[2.1 Definitions & Acronyms 4](#__RefHeading___Toc400_2862916375)

[2.2 Logical Scheme 5](#__RefHeading___Toc402_2862916375)

[2.3 Objects 5](#__RefHeading___Toc404_2862916375)

# 

# Business Description

## Business background

The subway system is a vital public transportation network, serving millions of passengers daily across multiple stations and lines. Managing the infrastructure efficiently requires maintaining trains, tracks, and stations, ensuring timely schedules, and optimizing employee workflows. Additionally, customer service, ticketing, and promotions are critical to maintaining ridership and revenue. The need for accurate data management in this complex system is crucial for operational efficiency, safety, and user satisfaction.

Traditionally, subway systems have relied on disparate tools and manual processes to track data related to stations, trains, schedules, ticket sales, and repairs. This results in a lack of centralized data management, leading to inefficiencies, errors, and delays in operational decision-making.

## Problems. Current Situation

The current system suffers from several major problems:

**Decentralized data management**: Train schedules, ticket sales, and repair logs are stored across various platforms, making it difficult to integrate and track in real-time.

**Inconsistent reporting**: Without a single source of truth, station managers and operational teams often report conflicting data, affecting decisions about scheduling, resource allocation, and repairs.

**Lack of transparency in maintenance**: Upkeep and repair records are manually tracked, leading to missed or delayed maintenance, increasing the risk of system failures.

**Customer experience issues**: There’s limited visibility into passenger flows and ticketing data, which affects promotions and pricing strategies.

## the Benefits of implementing a database. Project Vision

The implementation of a centralized database for subway infrastructure and management will address these challenges by providing:

**Centralized Data Management**: A single repository of data for stations, lines, schedules, tickets, and repairs will reduce discrepancies and increase operational transparency.

**Improved Operational Efficiency**: Real-time tracking of train schedules, employee shifts, and maintenance needs will streamline operations, reduce downtime, and ensure trains run on time.

**Better Maintenance Planning**: Maintenance logs for trains, stations, and tunnels will be digitized and tracked, enabling proactive scheduling of repairs and extending the lifespan of critical assets.

**Enhanced Customer Experience**: With accurate ticketing data and promotions, the system can better cater to customer needs, improving ridership and ensuring customer loyalty.

**Data-Driven Decisions**: By providing analytical insights into passenger flows, ticket sales, and operational efficiency, management can make data-driven decisions to optimize the network's performance.

# Model description

## Definitions & Acronyms

**Stations**: Physical locations where passengers board or exit subway trains.

**Lines**: Defined subway routes connecting stations.

**Trains:** Subway vehicles that operate on predefined lines.

**StationSchedules**: Timetables detailing when trains arrive and depart at specific stations.

**Employees**: Subway staff, including drivers, conductors, maintenance workers, and station managers.

**Payroll:** Employee payslips information.

**Tickets**: Various types of passes allowing passengers access to subway services.

**UpkeepRepairsMonitoring**: Maintenance activities performed on subway infrastructure, such as tracks, trains, stations, tunnels and ticket readers.

**TicketDiscount**: Discounts or offers applied to tickets, targeting different customer demographics (e.g., NONE, child, students, seniors).

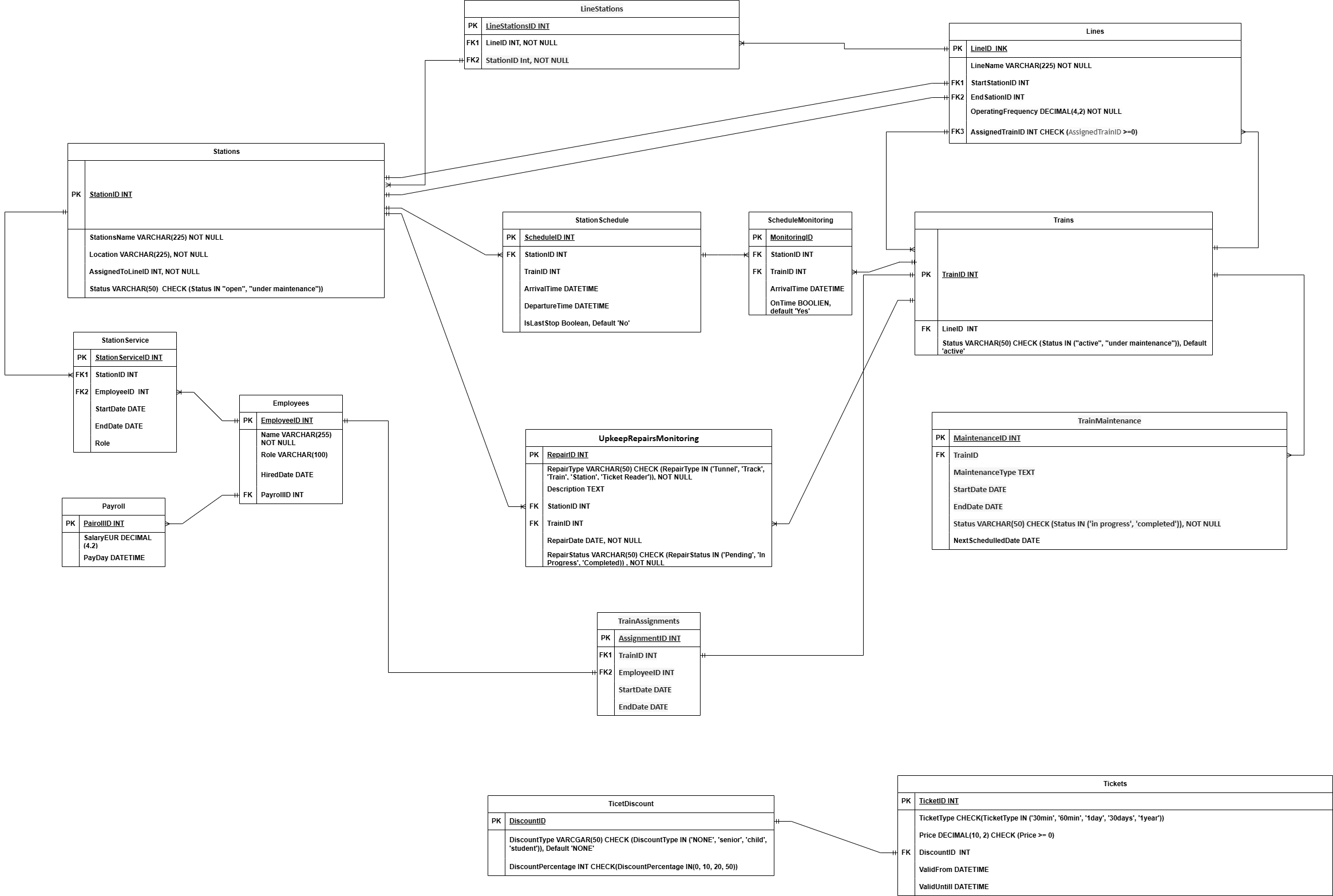
**ScheduleMonitoring: Tracking** if trains arrived on time to stations.

**TrainAssignments:** List of employees, that are assigned to specific train for specific period of time.

**TrainMaintenance:** Table of routine maintenance of trains.

**StationService:** List of employees assigned to specific station.

## Logical Scheme



## Objects

2.3.1

The Stations table stores information about each subway station in the network.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Stations | StationID | Unique identifier for each station, PK | Int |
| StationName | Name of the station | VARCHAR(225) NOT NULL |
|  | Location | Location of the station | VARCHAR(225) NOT NULL |
|  | Status | Condition of the station (open, under maintenance) | VARCHAR(50)  CHECK (Status IN "open", "under maintenance")) |

Linked to the StationService table, which defines train arrival and departure times for each station. Relationship type one to many.

Linked with **UpkeepRepairsMonitoring** table with one to many relation.

Linked with **StationSchedule**  table with one to many relation.

Linked with **Line**  table with one to many relation.

|  |  |  |  |
| --- | --- | --- | --- |
| StationID | StationName | Location | Status |
| 1 | Central Station | Central | Open |
| 2 | St. Bernard | North District | Under Maintenance |

2.3.2

The Lines table stores information about each line of stations in the network.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Lines | LineID | Unique identifier for each line, PK | Int, NOT NULL |
| LineName | Unique name of the line | VARCHAR(50) NOT NULL |
|  | StartStationID | ID of the first station, FK | INT, NOT NULL |
|  | EndSationID | Last station ID, FK | INT, NOT NULL |
|  | OperatingFrequency | How often train arrives | Decimal(4,2), NOT NULL |
|  | AssignedTrainID | Trains, that functioning at the line, FK | INT CHECK (AssignedTrainID>=0) |

StartStationID and EndSationID linked to Stations.StationID in one to one relation.

AssignedTrainID linked to Trains.TrainID in one to many relation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| LineID | LineName | StartStationID | EndSationID | OperatingFrequency | AssignedTrainID |
| 1 | Green | 1 | 56 | 5.00 | 5 |

2.3.3

The Trains table stores information about each trains in the network.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Trains | TrainID | Unique identifier for each train, PK | Int |
| LineID | Which line train is assigned to, FK | Int |
|  | Status | Status of train: active or under maintenance | VARCHAR(50) CHECK (Status IN ‘active’, ‘under maintenance’)) |

TrainID linked to Lines.AssignedTrainID in many to one relation.

TrainID linked to UpkeepRepairMoniotirng. TrainID in one to many relation.

TrainID linked to TrainAssigments. TrainID in one to one relation.

TrainID linked to ScheduleMonitoring.TrainID in one to many relation.

|  |  |  |
| --- | --- | --- |
| TrainID | LineID | Status |
| 5 | 1 | active |

2.3.4

The Employees table stores information about the employees of the subway.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Employees | EmployeeID | Unique identifier, PK | Int |
| Name | Name and surname of the employee | VARCHAR(255), NOT NULL |
|  | Role | Role of the employee | VARCHAR(100), NOT NULL |
|  | HiredDate | When employee was hired | DATE, NOT NULL |
|  | PayrollID | Payroll ID, FK | Int |

EmployeeID linked to Station.Service in one to many relation.

PayrollID linked to Payroll.PayrollID in one to many relation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EmployeeID | Name | Role | HiredDate | PayrollID |
| 1 | Jack Jones | Conductor | 1999-01-15 | 15 |

2.3.5

The Tickets table stores information about ticket types, prices duration.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Tickets | TicketID | Unique identifier, PK | Int |
| TicketType | What kind of the ticket it is: 30min, 60min, 1day, 30days, 1yesr. | VARCHAR (50) CHECK(TicketType IN ('30min', '60min', '1day', '30days', '1year')) |
|  | Price | Price of the ticket | Price DECIMAL(10, 2) CHECK (Price >= 0) |
|  | DiscountID | Information of discount, FK | Int |
|  | ValidFrom | When did ticket were activated | DATETIME |
|  | ValidUntil | Until when ticket is valid | DATETIME |

DiscountID linked to TicketDiscount.DiscountID in one to many relation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TicketID | TicketType | Price | DiscountID | ValidFrom | ValidUntil |
| 1 | 30min | 6,50 | 1 | 2025-01-01 12:30 | 2025-01-01 13:00 |

2.3.6

The TicketDiscounts table stores information about ticket discounts.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| TicketDiscount | DiscountID | Unique identifier, PK | Int |
| DiscountType | Who is obligated to the discount: child, student, senior. NONE identifies if ticket is not obligates to discount. | VARCGAR(50) CHECK (DiscountType IN ('NONE', 'senior', 'child', 'student')) |
|  | DiscountPercentage | How big is discount in percentage | INT CHECK (DiscountPercentage IN (0, 10, 20, 50)) |

DiscountID linked to Tickets.DiscountID in many to one relation.

|  |  |  |
| --- | --- | --- |
| DiscountID | DiscountType | DiscountPercentage |
| 1 | Child | 50 |

2.3.7

The LineStations table stores list of stations, assigned to specific lines.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| LineStations | LineStationsID | Unique identifier, PK | Int |
| LineID | Id of the line that contains specific station, FK | Int |
|  | StationID | ID of station, that belong to specific line, FK | Int |

StationID linked to Station.StationID in many to many relation.

|  |  |  |
| --- | --- | --- |
| LineStationsID | LineID | StationID |
| 1 | 1 | 10 |

2.3.8

The StationSchedules table stores timetables regarding detailing when trains arrive and depart at specific stations.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| StationSchedules | ScheduleID | Unique identifier, PK | Int |
| StationID | Specific station in the schedule ID number, FK | Int |
|  | TrainID | Specific train in the schedule ID number | Int |
|  | ArrivalTime | Train arriving time | DATETIME |
|  | DepartureTime | Train departing time | DATETIME |
|  | IsLastStop | Identifies if station is the last one in the line | Boolean, default ‘No’ |

StationID linked to Station.StationID in many to one relation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ScheduleID | StationID | TrainID | ArrivalTime | DepartureTime | IsLastStop |
| 1 | 5 | 8 | 2025-01-01 12:30 | 2025-01-01 12:31 | No |

2.3.9

The Payroll table stores information about employee payslips information.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Payroll | PayrollID | Unique identifier, PK | Int |
| SalaryEUR | Monthly salary in EUR | Decimal(4,2), NOT NULL |
|  | PayDay | Date of the salary | DATE, NOT NULL |

PayrollID linked to Employee.PayrollID in many to one relation.

|  |  |  |
| --- | --- | --- |
| PayrollID | SalaryEUR | PayDay |
| 1 | 3000,25 | 2025-02-05 |

2.3.10

The UpkeepRepairsMonitoring table stores information about maintenance activities performed on subway infrastructure, such as tracks, trains, stations, tunnels and ticket readers.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| UpkeepRepairsMonitoring | RepairID | Unique identifier, PK | Int |
| RepairType | What was repaired | VARCHAR(50) CHECK (RepairType IN ('Tunnel', 'Track', 'Train', 'Station', 'Ticket Reader')), NOT NULL |
|  | Description | Information on the repair | Text |
|  | StationID | Station under repair, may be null, if not aplicable, FK | Int |
|  | TrainID | Train under repair, may be null, if not aplicable, FK | Int |
|  | RepairDate | Start date of the repair work | Date, NOT NULL |
|  | RepairStatus | Status of the work | VARCHAR(50) CHECK (RepairStatus IN ('Pending', 'In Progress', 'Completed)) , NOT NULL |

StationID linked to Stations.StationID in many to one relation.

TrainID linked to Trains.TrainIDin many to one relation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| RepairID | RepairType | Description | StationID | TrainID | RepairDate | RepairStatus |
| 1 | Train | Front window shield replaced |  | 458 | 1999-05-03 | Completed |

2.3.11

The ScheduleMonitoring table stores information for tracking if trains arrived on time to stations.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| ScheduleMonitoring | MonitoringID | Unique identifier, PK | Int |
| StationID | Specific station in the schedule ID number, FK | Int |
|  | TrainID | Specific train in the schedule ID number, FK | Int |
|  | ArrivalTime | Date and Time train arrived | DATETIME |
|  | OnTime | Did train arrived on time | Boolean, default ‘Yes’ |

TrainID linked to Trains.TrainID in one to one relation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MonitoringID | StationID | TrainID | ArrivalTime | OnTime |
| 1 | 55 | 23 | 2025-02-02 | Yes |

2.3.12

The TrainAssignments table stores list of employees, that are assigned to specific train for specific period of time.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| TrainAssignments | TrainAssignmentsID | Unique identifier, PK | Int |
| TrainID | Specific train in the list ID number, FK | Int |
|  | EmployeeID | Employee ID, FK | Int |
|  | StartDate | Assignment to train start date | Date |
|  | EndDate | Assignment to train start date | Date |

TrainID linked to Trains.TrainID in one to one relation.

EmployeeID linked to Employee.EmployeeIDin one to one relation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TrainAssignmentsID | TrainID | EmployeeID | StartDate | EndDate |
| 1 | 58 | 89 | 2025-02-02 | 2025-03-02 |

2.3.13

The TrainMaintenance table stores information about routine maintenance of trains.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| TrainMaintenance | MaintenanceID | Unique identifier, PK | Int |
| TrainID | Train ID, FK | Int |
|  | MaintenanceType | Information on work needed | TEXT |
|  | StartDate | Maintenance started | Date |
|  | EndDate | Maintenance ended | date |
|  | Status | What is the work status | VARCHAR(50) CHECK (Status IN ('in ‘progress', 'completed')) |
|  | NextSchedulledDate | Date of next routine check | Date |

TrainID linked to Trains.TrainID in many to one relation.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| MaintenanceID | TrainID | MaintenanceType | StartDate | EndDate | Status | NextSchedulledDate |
| 1 | 125 | Checking breaks condition | 2025-02-02 | 2025-02-20 | Completed | 2025-05-02 |

2.3.14

The StationService table stores list of employees assigned to specific station.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| StationService | StationServiceID | Unique identifier, PK | Int |
| StationID | Station ID, FK | Int |
|  | EmployeeID | Employee ID, FK | Int |
|  | StartDate | Start date of assigned role | Date, NOT NULL |
|  | EndDate | End date of assigned role | Date, NOT NULL |
|  | Role | Employee responsibilities | VARCHAR(50) CHECK (Role IN ‘Maintenance’, ‘Station Manager’, ‘Cleaner’, ‘Conductor’)), Not Null |

EmployeeID linked to Employee.EmployeeID in many to one relation.

StationID linked to Stations.StationID in many to one relation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| StationServiceID | StationID | EmployeeID | StartDate | EndDate | Role |
| 1 | 58 | 25 | 2025-02-20 | 2025-05-20 | Cleaner |