Here are the following common potential data quality issues in database:

# Duplicated data:

We need to check duplicate data in the table to run below SQL queries in all the three tables. These queries like the following will return any duplicate employeeid, travelid and attendanceid values:

--Check duplicates in Table employee based on employeeid

SELECT

employeeid,

SUM(1) AS count

FROM employee

GROUP BY 1

HAVING SUM(1) > 1

--Check duplicates in Table TRAVELS based on travelid

SELECT

travelid,

SUM(1) AS count

FROM TRAVELS

GROUP BY 1

HAVING SUM(1) > 1

--Check duplicates in Table ATTENDANCE based on attendanceId

SELECT

attendanceid,

SUM(1) AS count

FROM ATTENDANCE

GROUP BY 1

HAVING SUM(1) > 1

We can create different types of queries to find out duplicate data in the table as per our needs. This is the basic idea to detect duplicate values in table.

# Incomplete fields in the table:

I have notes there are useful columns are missing in the employee table.

Table Employee in task:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **employeeId** | **name** | **surname** | **personalCode** | **startDate** |
| 1 | Alberts | Keda | 310172-11223 | 2021-01-14 |
| 2 | Sniedze | Ieviņa | 280282-22133 | 2021-02-10 |
| 3 | Tīna | Zibens | 310392-33211 | 2021-02-11 |

Employee table with additional columns (new columns marked light blue) in table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **employeeId** | **name** | **surname** | **personalCode** | **dateofbirth** | **jobtitle** | **phoneno** | **country** | **department** | **gender** | **created\_at** | **updated\_at** | **startDate** |
| 1 | Alberts | Keda | 310172-11223 | 1972-01-31 | python developer | 22222222 | Latvia | IT | female | 2021-02-16 | 2021-02-16 | 2021-01-14 |
| 2 | Sniedze | Ieviņa | 280282-22133 | 1982-02-28 | BI developer | 22222223 | Latvia | BI | female | 2021-02-17 | 2021-02-17 | 2021-02-10 |
| 3 | Tīna | Zibens | 310392-33211 | 1992-03-31 | CFO | 22222224 | Latvia | finance | male | 2021-02-18 | 2021-02-18 | 2021-02-11 |

Details of new columns in employee table:

|  |  |
| --- | --- |
| **dateofbirth** | employee's data of birth |
| **jobtitle** | employee's job title or position in the company |
| **phoneno** | employee's phone number |
| **country** | employee's working country |
| **department** | employee's department |
| **gender** | employee's gender |
| **created\_at** | employee's profile created in database |
| **updated\_at** | employee's profile updated in database |

Table travels in task:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **travelId** | **employeeId** | **process** | **byEmployee** | **country** | **startDate** | **endDate** | **endDate2** |
| 1 | 2 | approved | 1 | Estonia | 2022-01-02 | 2022-01-31 | 2022-01-31 |
| 2 | 3 | approved | 1 | Spain | 2021-12-12 | 2022-01-12 | 2022-01-12 |
| 3 | 3 | rejected | 1 | Greece | 2022-01-30 | 2022-02-14 | 2022-02-14 |

Table travels with additional columns (new columns marked light blue) in the table:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **travelId** | **employeeId** | **process** | **byEmployee** | **decidetorequest** | **country** | **startDate** | **endDate** | **created\_at** | **updated\_at** |
| 1 | 2 | approved | 1 | yes | Estonia | 2022-01-02 | 2022-01-31 | 2021-02-16 | 2021-02-16 |
| 2 | 3 | approved | 1 | yes | Spain | 2021-12-12 | 2022-01-12 | 2021-02-17 | 2021-02-17 |
| 3 | 3 | rejected | 1 | yes | Greece | 2022-01-30 | 2022-02-14 | 2021-02-18 | 2021-02-18 |

Details of new columns in travels table:

|  |  |
| --- | --- |
| **decidetorequest** | employee decides to request |
| **created\_at** | travel's data created in database |
| **updated\_at** | travel's data updated in database |

Table attendance in task:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **attendanceId** | **employeeId** | **date** | **office** | **floor** | **table** |
| 1 | 1 | 2022-01-03 | 1 | 3 | G1 |
| 2 | 1 | 2022-01-04 | 1 | 3 | G1 |
| 3 | 1 | 2022-01-05 | 0 |  |  |

Table attendance with additional columns (new columns marked light blue) in the table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **attendanceId** | **employeeId** | **date** | **office** | **workplace** | **floor** | **table** | **created\_at** | **updated\_at** |
| 1 | 1 | 2022-01-03 | 1 | office | 3 | G1 | 2021-02-16 | 2021-02-16 |
| 2 | 1 | 2022-01-04 | 1 | office | 3 | G1 | 2021-02-17 | 2021-02-17 |
| 3 | 1 | 2022-01-05 | 0 | home |  |  | 2021-02-18 | 2021-02-18 |

Details of new columns in attendance table:

|  |  |
| --- | --- |
| **workplace** | employee's work place value in words (office or home) |
| **created\_at** | attendance's data created in database |
| **updated\_at** | attendance's data updated in database |

# Missing values:

In table attendance, values are not available or missing for columns floor and table. If someone work’s from home. We should provide default value for column floor and table.

If column office value is 0, floor value should be 0 and table value should be null.

|  |  |  |
| --- | --- | --- |
| **office** | **floor** | **table** |
| 1 | 3 | G1 |
| 1 | 3 | G1 |
| 0 | 0 | null |

# Data Integrity

Data integrity means data should be complete, consistent and accurate in database.

Data type can be used in data integrity by providing proper data type to the column. For example int, varchar, text, date, timestamp, money are data types in SQL.

## Primary Key:

Primary key is a key in a table which identifies the each record uniquely. Primary key is used to locate a record. In primary key null’s should not be allowed, it should be unique and it cannot be modified.

## Foreign Key:

Foreign key is key in a one table refers to the primary key in another table. The referenced table is called parent table and table with foreign key is called child table.

For example, I have implemented in our tables. Please see below SQL queries:

----------------Table EMPLOYEE----------------------

-- Table: public.EMPLOYEE

-- DROP TABLE public.EMPLOYEE;

CREATE TABLE public.EMPLOYEE

(

employeeId integer ,

name character varying(45) NOT NULL,

surname character varying(45),

personalCode character varying(45) NOT NULL,

startDate date,

CONSTRAINT EMPLOYEE\_pkey PRIMARY KEY (employeeId)

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

ALTER TABLE public.EMPLOYEE

OWNER to postgres;

-- Index: idx\_EMPLOYEE\_name

-- DROP INDEX public.idx\_EMPLOYEE\_name;

CREATE INDEX idx\_name

ON public.EMPLOYEE USING btree

(name COLLATE pg\_catalog."default")

TABLESPACE pg\_default;

----------------Table TRAVELS----------------------

-- Table: public.TRAVELS

-- DROP TABLE public.TRAVELS;

CREATE TABLE public.TRAVELS

(

travelId integer,

employeeId integer,

process character varying(45),

byEmployee integer,

country character varying(45),

startDate date,

endDate date,

PRIMARY KEY (travelId),

FOREIGN KEY (employeeId) REFERENCES EMPLOYEE (employeeId)

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

ALTER TABLE public.TRAVELS

OWNER to postgres;

-- Index: idx\_TRAVELS\_startDate

-- DROP INDEX public.idx\_TRAVELS\_startDate;

CREATE INDEX idx\_startDate

ON public.TRAVELS (startDate);

----------------Table ATTENDANCE----------------------

-- Table: public.ATTENDANCE

-- DROP TABLE public.ATTENDANCE;

CREATE TABLE public.ATTENDANCE

(

attendanceId integer,

employeeId integer,

date date,

office integer,

floor integer,

tab character varying(2),

PRIMARY KEY (attendanceId),

FOREIGN KEY (employeeId) REFERENCES EMPLOYEE (employeeId)

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

ALTER TABLE public.ATTENDANCE

OWNER to postgres;

-- Index: idx\_ATTENDANCE\_Date

-- DROP INDEX public.idx\_ATTENDANCE\_Date;

CREATE INDEX idx\_Date

ON public.ATTENDANCE (date);