Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Базы данных (БД)

ОТЧЕТ

по лабораторной работе № 5

Тема работы: Улучшение модели базы данных

Выполнил

студент: гр. 151003 Барановский Р.А.

Проверил: Фадеева Е.Е.

Минск 2023

**Вариант 3(Работа авиакомпании)**

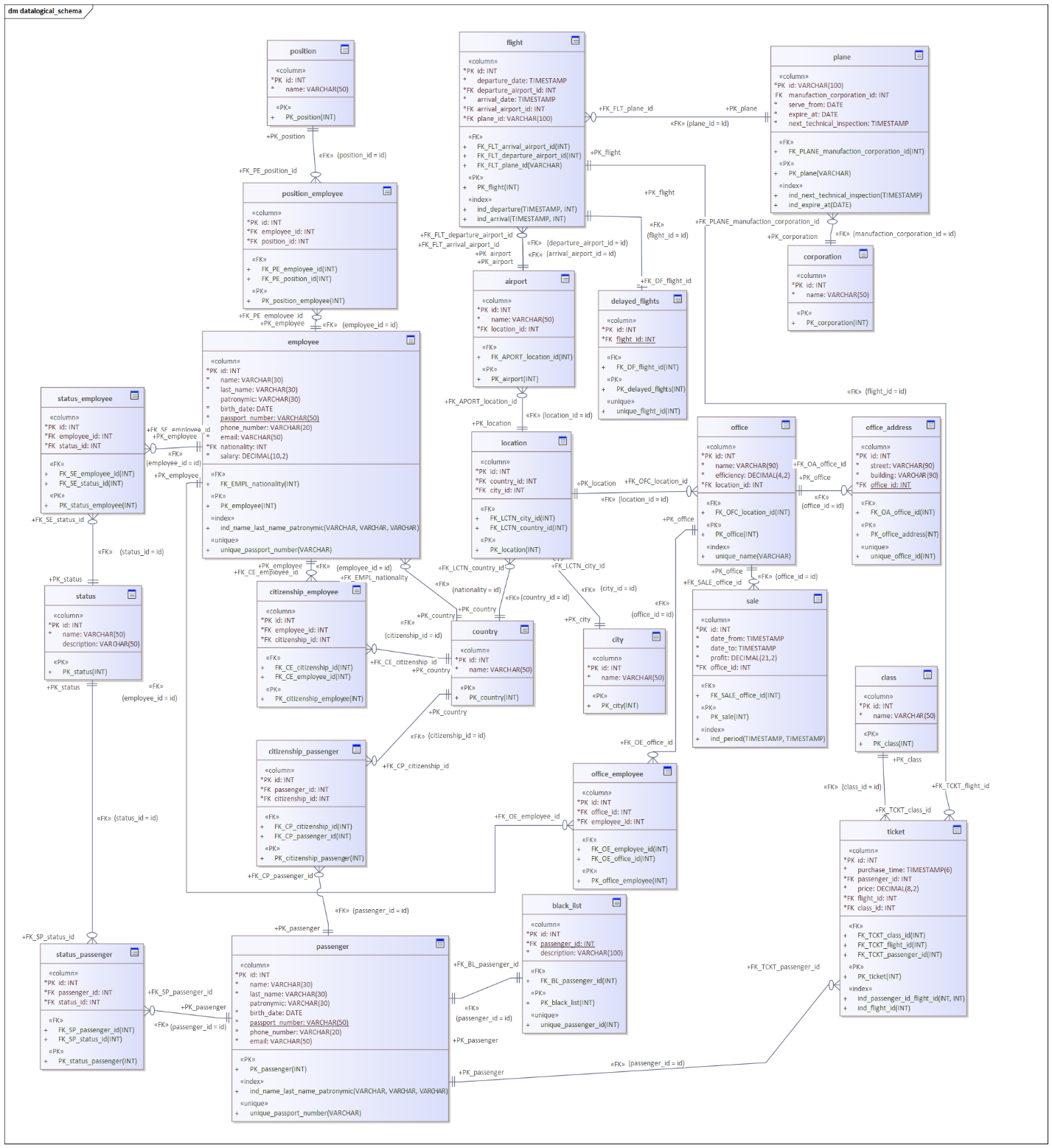


Рисунок 1 – Даталогическая модель БД

**1)**

Необходимо добавить следующие каскады(ниже приведен код создания необходимых FK вместе с соответствующими каскадными операциями):

ALTER TABLE `airport`

ADD CONSTRAINT `FK\_APORT\_location\_id`

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `black\_list`

ADD CONSTRAINT `FK\_BL\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_employee`

ADD CONSTRAINT `FK\_CE\_citizenship\_id`

FOREIGN KEY (`citizenship\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_employee`

ADD CONSTRAINT `FK\_CE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_passenger`

ADD CONSTRAINT `FK\_CP\_citizenship\_id`

FOREIGN KEY (`citizenship\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_passenger`

ADD CONSTRAINT `FK\_CP\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `delayed\_flights`

ADD CONSTRAINT `FK\_DF\_flight\_id`

FOREIGN KEY (`flight\_id`) REFERENCES `flight` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_arrival\_airport\_id`

FOREIGN KEY (`arrival\_airport\_id`) REFERENCES `airport` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_departure\_airport\_id`

FOREIGN KEY (`departure\_airport\_id`) REFERENCES `airport` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_plane\_id`

FOREIGN KEY (`plane\_id`) REFERENCES `plane` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `location`

ADD CONSTRAINT `FK\_LCTN\_city\_id`

FOREIGN KEY (`city\_id`) REFERENCES `city` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `location`

ADD CONSTRAINT `FK\_LCTN\_country\_id`

FOREIGN KEY (`country\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office`

ADD CONSTRAINT `FK\_OFC\_location\_id`

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_address`

ADD CONSTRAINT `FK\_OA\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_employee`

ADD CONSTRAINT `FK\_OE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_employee`

ADD CONSTRAINT `FK\_OE\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `plane`

ADD CONSTRAINT `FK\_PLANE\_manufaction\_corporation\_id`

FOREIGN KEY (`manufaction\_corporation\_id`) REFERENCES `corporation` (`id`) ON DELETE Set Null ON UPDATE Cascade

;

ALTER TABLE `position\_employee`

ADD CONSTRAINT `FK\_PE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `position\_employee`

ADD CONSTRAINT `FK\_PE\_position\_id`

FOREIGN KEY (`position\_id`) REFERENCES `position` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `sale`

ADD CONSTRAINT `FK\_SALE\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_employee`

ADD CONSTRAINT `FK\_SE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_employee`

ADD CONSTRAINT `FK\_SE\_status\_id`

FOREIGN KEY (`status\_id`) REFERENCES `status` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_passenger`

ADD CONSTRAINT `FK\_SP\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_passenger`

ADD CONSTRAINT `FK\_SP\_status\_id`

FOREIGN KEY (`status\_id`) REFERENCES `status` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_class\_id`

FOREIGN KEY (`class\_id`) REFERENCES `class` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_flight\_id`

FOREIGN KEY (`flight\_id`) REFERENCES `flight` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

Как можно заметить, везде используется «ON DELETE Cascade ON UPDATE CASCADE», за исключением FK\_PLANE\_manufaction\_corporation\_id, где используется «ON DELETE Set Null ON UPDATE Cascade», так как при удалении информации о корпорации-производителе самолета не должна удаляться информация о самом воздушном судне.

**2)**

Для грамотного создания представлений(виртуальных таблиц) необходимо четко знать о правах доступа приложения к БД, какая структура данных необходима приложению, использующему БД, какие данные в принципе необходимы приложению. В связи с отсутствием данной информации принято решение не добавлять представления в базу данных.

**3)**

Необходимо добавить следующие проверки:

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`name`, 1)) AS BINARY) = LEFT(`name`, 1));

;

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_last\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`last\_name`, 1)) AS BINARY) = LEFT(`last\_name`, 1));

;

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_patronymic\_first\_upper` CHECK (CAST(UPPER(LEFT(`patronymic`, 1)) AS BINARY) = LEFT(`patronymic`, 1));

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`name`, 1)) AS BINARY) = LEFT(`name`, 1));

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_last\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`last\_name`, 1)) AS BINARY) = LEFT(`last\_name`, 1));

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_patronymic\_first\_upper` CHECK (CAST(UPPER(LEFT(`patronymic`, 1)) AS BINARY) = LEFT(`patronymic`, 1));

;

Данные проверки необходимы для проверки, что имя, фамилия, отчество работников и пассажиров начинаются с заглавной буквы.

**4)**

Необходимо добавить следующие триггеры:

CREATE DEFINER=`root`@`localhost` TRIGGER `employee\_BEFORE\_INSERT\_age\_must\_be\_greater\_than\_or\_equal\_to\_18` BEFORE INSERT ON `employee` FOR EACH ROW BEGIN

IF datediff(CURRENT\_DATE(), NEW.birth\_date) / 365.2425 < 18 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Employee age must be greater than or equal to 18';

END IF;

END

Необходимы для исключения возможности добавления сотрудников, чей возраст меньше 18 лет.

**5), 6)**

В добавлении подпрограмм нет необходимости. Для исключения возможности наличия в БД данных, нарушающих условия предметной области, добавлены различные FK, индексы, проверки и триггеры. Приложение заказчика может составлять необходимые запросы к БД без каких-либо ограничений.

Итоговый SQL-скрипт:

SET FOREIGN\_KEY\_CHECKS=0 ;

/\* Shema creation \*/

DROP SCHEMA IF EXISTS `bd\_lab5` ;

CREATE SCHEMA IF NOT EXISTS `bd\_lab5` DEFAULT CHARACTER SET utf8 ;

USE `bd\_lab5` ;

/\* Drop Tables \*/

DROP TABLE IF EXISTS `airport` CASCADE

;

DROP TABLE IF EXISTS `black\_list` CASCADE

;

DROP TABLE IF EXISTS `citizenship\_employee` CASCADE

;

DROP TABLE IF EXISTS `citizenship\_passenger` CASCADE

;

DROP TABLE IF EXISTS `city` CASCADE

;

DROP TABLE IF EXISTS `class` CASCADE

;

DROP TABLE IF EXISTS `corporation` CASCADE

;

DROP TABLE IF EXISTS `country` CASCADE

;

DROP TABLE IF EXISTS `delayed\_flights` CASCADE

;

DROP TABLE IF EXISTS `employee` CASCADE

;

DROP TABLE IF EXISTS `flight` CASCADE

;

DROP TABLE IF EXISTS `location` CASCADE

;

DROP TABLE IF EXISTS `office` CASCADE

;

DROP TABLE IF EXISTS `office\_address` CASCADE

;

DROP TABLE IF EXISTS `office\_employee` CASCADE

;

DROP TABLE IF EXISTS `passenger` CASCADE

;

DROP TABLE IF EXISTS `plane` CASCADE

;

DROP TABLE IF EXISTS `position` CASCADE

;

DROP TABLE IF EXISTS `position\_employee` CASCADE

;

DROP TABLE IF EXISTS `sale` CASCADE

;

DROP TABLE IF EXISTS `status` CASCADE

;

DROP TABLE IF EXISTS `status\_employee` CASCADE

;

DROP TABLE IF EXISTS `status\_passenger` CASCADE

;

DROP TABLE IF EXISTS `ticket` CASCADE

;

/\* Create Tables \*/

CREATE TABLE `airport`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

`location\_id` INT NOT NULL,

CONSTRAINT `PK\_airport` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `black\_list`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`passenger\_id` INT NOT NULL,

`description` VARCHAR(100) NOT NULL,

CONSTRAINT `PK\_black\_list` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `citizenship\_employee`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`employee\_id` INT NOT NULL,

`citizenship\_id` INT NOT NULL,

CONSTRAINT `PK\_citizenship\_employee` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `citizenship\_passenger`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`passenger\_id` INT NOT NULL,

`citizenship\_id` INT NOT NULL,

CONSTRAINT `PK\_citizenship\_passenger` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `city`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_city` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `class`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_class` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `corporation`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_corporation` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `country`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_country` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `delayed\_flights`

(

`id` INT NOT NULL,

`flight\_id` INT NOT NULL,

CONSTRAINT `PK\_delayed\_flights` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `employee`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(30) NOT NULL,

`last\_name` VARCHAR(30) NOT NULL,

`patronymic` VARCHAR(30) NULL,

`birth\_date` DATE NOT NULL,

`passport\_number` VARCHAR(50) NOT NULL,

`phone\_number` VARCHAR(20) NOT NULL,

`email` VARCHAR(50) NOT NULL,

`nationality` INT NOT NULL,

`salary` DECIMAL(10,2) NOT NULL,

CONSTRAINT `PK\_employee` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `flight`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`departure\_date` TIMESTAMP NOT NULL,

`departure\_airport\_id` INT NOT NULL,

`arrival\_date` TIMESTAMP NOT NULL,

`arrival\_airport\_id` INT NOT NULL,

`plane\_id` VARCHAR(100) NOT NULL,

CONSTRAINT `PK\_flight` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `location`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`country\_id` INT NOT NULL,

`city\_id` INT NOT NULL,

CONSTRAINT `PK\_location` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `office`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(90) NOT NULL,

`efficiency` DECIMAL(4,2) NOT NULL,

`location\_id` INT NOT NULL,

CONSTRAINT `PK\_office` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `office\_address`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`street` VARCHAR(90) NOT NULL,

`building` VARCHAR(90) NOT NULL,

`office\_id` INT NOT NULL,

CONSTRAINT `PK\_office\_address` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `office\_employee`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`office\_id` INT NOT NULL,

`employee\_id` INT NOT NULL,

CONSTRAINT `PK\_office\_employee` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `passenger`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(30) NOT NULL,

`last\_name` VARCHAR(30) NOT NULL,

`patronymic` VARCHAR(30) NULL,

`birth\_date` DATE NOT NULL,

`passport\_number` VARCHAR(50) NOT NULL,

`phone\_number` VARCHAR(20) NOT NULL,

`email` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_passenger` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `plane`

(

`id` VARCHAR(100) NOT NULL,

`manufaction\_corporation\_id` INT NULL,

`serve\_from` DATE NOT NULL,

`expire\_at` DATE NOT NULL,

`next\_technical\_inspection` TIMESTAMP NOT NULL,

CONSTRAINT `PK\_plane` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `position`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

CONSTRAINT `PK\_position` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `position\_employee`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`employee\_id` INT NOT NULL,

`position\_id` INT NOT NULL,

CONSTRAINT `PK\_position\_employee` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `sale`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`date\_from` TIMESTAMP NOT NULL,

`date\_to` TIMESTAMP NOT NULL,

`profit` DECIMAL(21,2) NOT NULL,

`office\_id` INT NOT NULL,

CONSTRAINT `PK\_sale` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `status`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NOT NULL,

`description` VARCHAR(50) NULL,

CONSTRAINT `PK\_status` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `status\_employee`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`employee\_id` INT NOT NULL,

`status\_id` INT NOT NULL,

CONSTRAINT `PK\_status\_employee` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `status\_passenger`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`passenger\_id` INT NOT NULL,

`status\_id` INT NOT NULL,

CONSTRAINT `PK\_status\_passenger` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `ticket`

(

`id` INT NOT NULL AUTO\_INCREMENT,

`purchase\_time` TIMESTAMP(6) NOT NULL,

`passenger\_id` INT NOT NULL,

`price` DECIMAL(8,2) NOT NULL,

`flight\_id` INT NOT NULL,

`class\_id` INT NOT NULL,

CONSTRAINT `PK\_ticket` PRIMARY KEY (`id` ASC)

)

;

/\* Create Primary Keys, Indexes, Uniques, Checks \*/

ALTER TABLE `black\_list`

ADD CONSTRAINT `unique\_passenger\_id` UNIQUE (`passenger\_id` ASC)

;

ALTER TABLE `delayed\_flights`

ADD CONSTRAINT `unique\_flight\_id` UNIQUE (`flight\_id` ASC)

;

ALTER TABLE `employee`

ADD CONSTRAINT `unique\_passport\_number` UNIQUE (`passport\_number` ASC)

;

ALTER TABLE `employee`

ADD INDEX `ind\_name\_last\_name\_patronymic` (`name` ASC, `last\_name` ASC, `patronymic` ASC)

;

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`name`, 1)) AS BINARY) = LEFT(`name`, 1));

;

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_last\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`last\_name`, 1)) AS BINARY) = LEFT(`last\_name`, 1));

;

ALTER TABLE `employee`

ADD CONSTRAINT `CHK\_employee\_patronymic\_first\_upper` CHECK (CAST(UPPER(LEFT(`patronymic`, 1)) AS BINARY) = LEFT(`patronymic`, 1));

;

ALTER TABLE `flight`

ADD INDEX `ind\_departure` (`departure\_date` ASC, `departure\_airport\_id` ASC)

;

ALTER TABLE `flight`

ADD INDEX `ind\_arrival` (`arrival\_date` ASC, `arrival\_airport\_id` ASC)

;

ALTER TABLE `office`

ADD INDEX `unique\_name` (`name` ASC)

;

ALTER TABLE `office\_address`

ADD CONSTRAINT `unique\_office\_id` UNIQUE (`office\_id` ASC)

;

ALTER TABLE `passenger`

ADD CONSTRAINT `unique\_passport\_number` UNIQUE (`passport\_number` ASC)

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`name`, 1)) AS BINARY) = LEFT(`name`, 1));

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_last\_name\_first\_upper` CHECK (CAST(UPPER(LEFT(`last\_name`, 1)) AS BINARY) = LEFT(`last\_name`, 1));

;

ALTER TABLE `passenger`

ADD CONSTRAINT `CHK\_passenger\_patronymic\_first\_upper` CHECK (CAST(UPPER(LEFT(`patronymic`, 1)) AS BINARY) = LEFT(`patronymic`, 1));

;

ALTER TABLE `country`

ADD CONSTRAINT `unique\_name` UNIQUE (`name` ASC)

;

ALTER TABLE `city`

ADD CONSTRAINT `unique\_name` UNIQUE (`name` ASC)

;

ALTER TABLE `passenger`

ADD INDEX `ind\_name\_last\_name\_patronymic` (`name` ASC, `last\_name` ASC, `patronymic` ASC)

;

ALTER TABLE `plane`

ADD INDEX `ind\_next\_technical\_inspection` (`next\_technical\_inspection` ASC)

;

ALTER TABLE `plane`

ADD INDEX `ind\_expire\_at` (`expire\_at` ASC)

;

ALTER TABLE `sale`

ADD INDEX `ind\_period` (`date\_from` ASC, `date\_to` ASC)

;

ALTER TABLE `ticket`

ADD INDEX `ind\_passenger\_id\_flight\_id` (`passenger\_id` ASC, `flight\_id` ASC)

;

ALTER TABLE `ticket`

ADD INDEX `ind\_flight\_id` (`flight\_id` ASC)

;

/\* Create Foreign Key Constraints \*/

ALTER TABLE `employee`

ADD CONSTRAINT `FK\_EMPL\_nationality`

FOREIGN KEY (`nationality`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `airport`

ADD CONSTRAINT `FK\_APORT\_location\_id`

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `black\_list`

ADD CONSTRAINT `FK\_BL\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_employee`

ADD CONSTRAINT `FK\_CE\_citizenship\_id`

FOREIGN KEY (`citizenship\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_employee`

ADD CONSTRAINT `FK\_CE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_passenger`

ADD CONSTRAINT `FK\_CP\_citizenship\_id`

FOREIGN KEY (`citizenship\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `citizenship\_passenger`

ADD CONSTRAINT `FK\_CP\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `delayed\_flights`

ADD CONSTRAINT `FK\_DF\_flight\_id`

FOREIGN KEY (`flight\_id`) REFERENCES `flight` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_arrival\_airport\_id`

FOREIGN KEY (`arrival\_airport\_id`) REFERENCES `airport` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_departure\_airport\_id`

FOREIGN KEY (`departure\_airport\_id`) REFERENCES `airport` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `flight`

ADD CONSTRAINT `FK\_FLT\_plane\_id`

FOREIGN KEY (`plane\_id`) REFERENCES `plane` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `location`

ADD CONSTRAINT `FK\_LCTN\_city\_id`

FOREIGN KEY (`city\_id`) REFERENCES `city` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `location`

ADD CONSTRAINT `FK\_LCTN\_country\_id`

FOREIGN KEY (`country\_id`) REFERENCES `country` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office`

ADD CONSTRAINT `FK\_OFC\_location\_id`

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_address`

ADD CONSTRAINT `FK\_OA\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_employee`

ADD CONSTRAINT `FK\_OE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `office\_employee`

ADD CONSTRAINT `FK\_OE\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `plane`

ADD CONSTRAINT `FK\_PLANE\_manufaction\_corporation\_id`

FOREIGN KEY (`manufaction\_corporation\_id`) REFERENCES `corporation` (`id`) ON DELETE Set Null ON UPDATE Cascade

;

ALTER TABLE `position\_employee`

ADD CONSTRAINT `FK\_PE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `position\_employee`

ADD CONSTRAINT `FK\_PE\_position\_id`

FOREIGN KEY (`position\_id`) REFERENCES `position` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `sale`

ADD CONSTRAINT `FK\_SALE\_office\_id`

FOREIGN KEY (`office\_id`) REFERENCES `office` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_employee`

ADD CONSTRAINT `FK\_SE\_employee\_id`

FOREIGN KEY (`employee\_id`) REFERENCES `employee` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_employee`

ADD CONSTRAINT `FK\_SE\_status\_id`

FOREIGN KEY (`status\_id`) REFERENCES `status` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_passenger`

ADD CONSTRAINT `FK\_SP\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `status\_passenger`

ADD CONSTRAINT `FK\_SP\_status\_id`

FOREIGN KEY (`status\_id`) REFERENCES `status` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_class\_id`

FOREIGN KEY (`class\_id`) REFERENCES `class` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_flight\_id`

FOREIGN KEY (`flight\_id`) REFERENCES `flight` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `ticket`

ADD CONSTRAINT `FK\_TCKT\_passenger\_id`

FOREIGN KEY (`passenger\_id`) REFERENCES `passenger` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

/\* Triggers \*/

DELIMITER $$

CREATE DEFINER=`root`@`localhost` TRIGGER `employee\_BEFORE\_INSERT\_age\_must\_be\_greater\_than\_or\_equal\_to\_18` BEFORE INSERT ON `employee` FOR EACH ROW BEGIN

IF datediff(CURRENT\_DATE(), NEW.birth\_date) / 365.2425 < 18 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Employee age must be greater than or equal to 18';

END IF;

END$$

DELIMITER ;

SET FOREIGN\_KEY\_CHECKS=1

;