Таблица сотрудники:

0 – не будет отображаться в главном окне (будет доступно только при открытии товара)

1 – данный товар будет отображаться в главном окне приложения

1 – стоит обратить внимание

***Таблица «Сотрудник»:***

* ***ФИО*** – является обязательным полем для заполнения
* Паспортные данные – является обязательным полем для заполнения и должно содержать в себе
* Боковой номер – 10-значное число, является уникальным;
* Дата выдачи – является числом;
* Код подразделения – 6 цифр, где 3 цифры разделены чертой;
* Телефон
* Адрес проживания
* Должность
* Заработная плата
* Логин
* Пароль
* Серийный номер

***Таблица «Клиент»:***

* ФИО
* Телефон
* Адрес
* Дата рождения – поле является не обязательным, необходимо для получения скидки

***Таблица «Поставщик»:***

* Название компании
* Адрес
* Стоимость поставки
* Дата поставки
* Телефон

***Таблица «Дверь»:***

* Название
* Модель
* Стоимость
* Артикул
* Кол-во
* Поставщик
* Цвет

/\*Сотрудник\*/

CREATE TABLE employee

(

employeeId SERIAL PRIMARY KEY UNIQUE,/\*id сотрудника\*/

image TEXT,/\*Изображение\*/

employeeFIO VARCHAR(500) NOT NULL,/\*ФИО\*/

sideNumber VARCHAR(50) NOT NULL UNIQUE ,/\*Cерия нoмер паспорта\*/

dateOfIssue DATe NOT NULL,/\*Дата выдачи\*/

departmentCode INTEGER NOT NULL CHECK(departmentCode > 0),/\*Код подразделения\*/

employeePhone VARCHAR(11) NOT NULL,/\*Телефон\*/

employeeAdress VARCHAR(500) NOT NULL,/\*Адрес\*/

jobTitle VARCHAR(150) NOT NULL,/\*Должность\*/

wage MONEY NOT NULL ,/\*ЗП\*/

employeeLogin VARCHAR(50) NOT NULL UNIQUE,/\*Логин\*/

employeePassword VARCHAR(50)NOT NULL UNIQUE,/\*Пароль\*/

serialNumber VARCHAR(50) NOT NULL UNIQUE/\*Серийный номер\*/

);

insert into employee

values (1, , 'Козманов Олег Генадьевич', 7586214995, '2022-05-25', 655489, 78965412598, 'г. Арамиль, ул. Ленина, дом 34', 'Менеджер', 23000, 'kozmanov.og@gmail.com', 'Figman567', '8A4')

/\*Клиент\*/

CREATE TABLE client

(

clientId SERIAL PRIMARY KEY UNIQUE,/\*id клиента\*/

clientFIO VARCHAR(500) NOT NULL,/\*ФИО\*/

clientPhone VARCHAR(11)NOT NULL,/\*Телефон\*/

clientAdress VARCHAR(500) NOT NULL/\*Адрес\*/

);

/\*Поставщики\*/

CREATE TABLE suppliers

(

suppliersId SERIAL PRIMARY KEY UNIQUE,/\*id поставщика\*/

suppliersTitle VARCHAR(150) NOT NULL, /\*Название компании\*/

suppliersAdress VARCHAR(200) NOT NULL, /\*Адрес поставщиков\*/

deliveryDate DATe NOT NULL,/\*Дата поставки\*/

deliveryCost MONEY NOT NULL, /\*Стоимость поставки\*/

supplierstPhone VARCHAR(20) NOT NULL/\*Телефон\*/

);

/\*Дверь\*/

CREATE TABLE door

(

doorId SERIAL PRIMARY KEY UNIQUE,/\*id поставщика\*/

doorTitle VARCHAR(150) NOT NULL,/\*Название\*/

doorModel VARCHAR(150) NOT NULL,/\*Модель\*/

doorCost MONEY NOT NULL,/\*Стоимость\*/

doorArticleNumber INTEGER NOT NULL CHECK(doorArticleNumber > 0) UNIQUE,/\*Артикул\*/

doorQuantity INTEGER NOT NULL,/\*Количесто\*/

doorSuppliers VARCHAR(150) not NULL,

doorColor VArchar(150) NOT NULL

);

/\*Ручка\*/

CREATE TABLE pen

(

penId SERIAL PRIMARY KEY UNIQUE,/\*id поставщика\*/

penTitle VARCHAR(150) NOT NULL,/\*Название\*/

penModel VARCHAR(150) NOT NULL,/\*Модель\*/

penCost MONEY NOT NULL,/\*Стоимость\*/

penArticleNumber INTEGER NOT NULL CHECK(penArticleNumber > 0) UNIQUE,/\*Артикул\*/

penQuantity INTEGER NOT NULL,/\*Количесто\*/

penSuppliers VARCHAR(150) not NULL,/\*\*/

penColor VArchar(150) NOT NULL

);

/\*Монтаж\*/

CREATE TABLE montage

(

montageId SERIAL PRIMARY KEY UNIQUE, /\*id монтажа\*/

dateMontage DATE NOT NULL, /\*Дата\*/

timeMontage TIME NOT NULL, /\*Время\*/

fitter VARCHAR(500) NOT NULL,/\*Монтажник\*/

montageCost money NOT NULL DEFAULT 0/\*Итоговая сумма за все работы\*/

);

CREATE TABLE froze

(

frozeId SERIAL PRIMARY KEY UNIQUE, /\*id монтажа\*/

dateFroze DATE NOT NULL, /\*Дата\*/

timeFroze TIME NOT NULL, /\*Время\*/

measurer VARCHAR(500) NOT NULL,/\*Монтажник\*/

frozeCost money NOT NULL DEFAULT 0/\*Итоговая сумма за все работы\*/

);

create table jobTitle

(

jobId Serial primary key

title varchar(50) uniqiue

);

Необходимо создать таблицы

-***«Сотрудники (employee)»:***

1. id (employeeId)

2. ФИО (employeeFIO)

2.1. Фото сотрудника (employeeImage)

3. Паспортные данные

3.1. Серийный номер (10 цифр) (serialNumber)

3.2. Дата выдачи (dateOfissue)

3.3. Код подразделения (divisionСode)

4. Телефон (employeePhone)

5. Адрес (employeeAdress)

6. Должность (employeeJobTitle)

7. Заработная плата (employeeWage)

8. Регистрационные данные

8.1. Логин (login)

8.2. Пароль (passwordEmployee)

8.3. Серийный номер (serialNumberEmployee)

***Код программы:***

create table employee

(

employeeId serial primary key not null,

employeeFIO varchar(150) not null,

employeeImage text not null,

serialNumber varchar(10) unique not null default 'xxxxx xxxxx',

dateOfissue date not null,

divisionСode varchar(7) not null default '666-666',

employeePhone VARCHAR(12) unique not null default '+7(999)999-99-99',

employeeAdress varchar(500) not null,

employeeJobTitle varchar(50) unique not null,

employeeWage money not null,

login varchar(50) unique not null default 'xxxxxx.xxx@gmail.com',

passwordEmployee varchar(50) unique not null,

serialNumberEmployee varchar(4) unique not null

);

***- «Должность (job)»:***

1. id (jobId)

2. Название должности (titleJob)

3. Описание, что включает в себя данная работа (jobDescription)

***Код программы:***

create table job

(

jobId serial primary key not null,

titleJob varchar(50) unique not null,

jobDescription varchar(500) not null

);

***-«Клиент (client)»:***

1. id (clientId)

2. ФИО (clientFIO)

3. Телефон (clientPhone)

4. Адрес (clientAdress)

5. Дата рождения (поле является не обязательным, необходимо для скидки в размере 5% вместо обычных 3%) (dateOfBirth)

***Код программы:***

create table client

(

clientId serial primary key not null,

clientFIO varchar(150) not null,

clientPhone VARCHAR(12) unique not null default '+7(999)999-99-99',

clientAdress varchar(500) not null,

dateOfBirth date not null

);

***- «Оплата (payment)»:***

1. id (paymentId)

2. Название способа оплаты (paymentMethods)

***Код программы:***

create table payment

(

paymentId serial primary key not null,

paymentMethods varchar(50) not null

);

***-«Поставщики(suppliers)»:*** 1. Id (suppliersId)

2. Название фирмы (suppliersTitle)

3. Адрес фирмы (suppliersAdress)

4. Телефон (suppliersPhone)

***Код программы:***

create table suppliers

(

suppliersId serial primary key not null,

suppliersTitle varchar(500) unique not null,

suppliersAdress varchar(500) not null,

suppliersPhone VARCHAR(12) unique not null default '+7(999)999-99-99'

);

***-«Цвет двери (doorСolor)»:***

1. id (doorColorId)

2. Название цвета (doorColorTitle)

***Код программы:***

create table doorСolor

(

doorColorId serial primary key not null,

doorColorTitle varchar(100) unique not null

);

***-«Цвет ручек (penСolor)»:***

1. id (penСolorId)

2. Название цвета (penСolorTitle)

***Код программы:***

create table penСolor

(

penСolorId serial primary key not null,

penСolorTitle varchar(100) unique not null

);

***-«Цвет стекла (glassСolor)»:***

1. id (glassСolorId)

2. Название (glassСolorTitle)

Код программы:

create table glassСolor

(

glassСolorId serial primary key not null,

glassСolorTitle varchar(100) unique not null

);

***- «Двери (door)»:***

1. id (doorId)

2. Название (doorTitle)

3. Модель (doorModel)

4. Стоимость (doorCost)

5. Количество на складе (NumberOfDoors)

6. Цвет двери (doorColor)

7. Стекло – булевая кнопка, если эта кнопка == true, то появляется поле для ввода цвета стекла (doorGlass)

8. Артикул (vendorCodeDoor)

Код программы:

create table door

(

doorId serial primary key not null,

doorTitle varchar(100) not null,

doorModel varchar(100) not null,

doorCost money not null,

NumberOfDoors integer not null,

doorColor varchar(100) not null,

doorGlass bool,

vendorCodeDoor varchar(10) not null

);

***-«Ручка(pen)»:*** 1. Id (penId)

2. Название (penTitle)

3. Стоимость (penCost)

4. Количество на складе (NumberOfPen)

5. Цвет ручки (penColor)

6. Артикул (vendorCodePen)

Код программы:

create table pen

(

penId serial primary key not null,

penTitle varchar(100) not null,

penCost money not null,

NumberOfPen integer not null,

penColor varchar(100) not null,

vendorCodePen varchar(10) not null

);

***-«Продажа (sale)»:***

1. id (saleId)

2. Покупатель (buyer)

3. Дверь (saleDoor)

4. Цвет двери (saleDoorColor)

5. Ручка (salePen)

6. Цвет ручки (salePenColor)

7. Цвет стекла (saleGlassColorSS)

8. Общая стоимость (saleCost)

***Код программы:***

create table sale

(

saleId serial primary key not null,

buyer varchar(100) not null,

salesman varchar(100) not null,

saleDoor varchar(100) not null,

saleDoorColor varchar(100) not null,

salePen varchar(100) not null,

salePenColor varchar(100) not null,

saleGlassColor varchar(100) not null,

salePayment varchar(100) not null,

saleCost money not null

);

***Заполнение:***

Итоговая версия

create table employee

(

employeeId serial primary key not null,

employeeFIO varchar(150) not null,

employeeImage text not null,

serialNumber varchar(10) unique not null default 'xxxxx xxxxx',

dateOfissue date not null,

divisionСode varchar(7) not null default '666-666',

employeePhone VARCHAR(12) unique not null default '+7(999)999-99-99',

employeeAdress varchar(500) not null,

employeeJobTitle varchar(50) unique not null,

employeeWage money not null,

login varchar(50) unique not null default 'xxxxxx.xxx@gmail.com',

passwordEmployee varchar(50) unique not null,

serialNumberEmployee varchar(4) unique not null,

foreign key (employeeId) references client (clientId),

foreign key (employeeJobTitle) references job (jobId),

foreign key (employeeId) references sale (saleId)

);

create table job

(

jobId serial primary key not null,

titleJob varchar(50) unique not null,

jobDescription varchar(500) not null

);

create table client

(

clientId serial primary key not null,

clientFIO varchar(150) not null,

clientPhone VARCHAR(12) unique not null default '+7(999)999-99-99',

clientAdress varchar(500) not null,

dateOfBirth date not null

);

create table payment

(

paymentId serial primary key not null,

paymentMethods varchar(50) not null

);

create table suppliers

(

suppliersId serial primary key not null,

suppliersTitle varchar(500) unique not null,

suppliersAdress varchar(500) not null,

suppliersPhone VARCHAR(12) unique not null default '+7(999)999-99-99'

);

create table doorСolor

(

doorColorId serial primary key not null,

doorColorTitle varchar(100) unique not null

);

create table penСolor

(

penСolorId serial primary key not null,

penСolorTitle varchar(100) unique not null

);

create table glassСolor

(

glassСolorId serial primary key not null,

glassСolorTitle varchar(100) unique not null

);

create table door

(

doorId serial primary key not null,

doorTitle varchar(100) not null,

doorModel varchar(100) not null,

doorCost money not null,

NumberOfDoors integer not null,

doorColor varchar(100) not null,

doorGlass bool,

vendorCodeDoor varchar(10) not null

);

create table pen

(

penId serial primary key not null,

penTitle varchar(100) not null,

penCost money not null,

NumberOfPen integer not null,

penColor varchar(100) not null,

vendorCodePen varchar(10) not null

);

create table sale

(

saleId serial primary key not null,

buyer varchar(100) not null,

salesman varchar(100) not null,

saleDoor varchar(100) not null,

saleDoorColor varchar(100) not null,

salePen varchar(100) not null,

salePenColor varchar(100) not null,

saleGlassColor varchar(100) not null,

salePayment varchar(100) not null,

saleCost money not null,

foreign key (buyer) references client (clientId),

foreign key (salesman) references employee (employeeFIO),

foreign key (saleDoor) references door (doorId),

foreign key (saleDoorColor) references doorСolor (doorColorId),

foreign key (salePen) references pen (penId),

foreign key (salePenColor) references penСolor (penСolorId),

foreign key (saleGlassColor) references glassСolor (glassСolorId),

foreign key (salePayment) references payment (paymentId)

);