Государственный Университет Молдовы

Факультет Математики и Информатики

Департамент Информатики

Индивидуальная работа

“Системы управления базами данных”

Проверил: Светлана Бодруг

Выполнил: Чобану Артём

Группа: i1902

Кишинев 2021

Задание – спроектировать базу данных в данной предметной области.

# **Страховая компания**

Описание предметной области

Вы работаете в страховой компании. Вашей задачей является отслеживание ее финансовой деятельности.

Компания имеет различные филиалы по всей стране. Каждый филиал характеризуется названием, адресом и телефоном.

Деятельность компании организована следующим образом: к вам обращаются различные лица с целью заключения договора о страховании.

В зависимости от принимаемых на страхование объектов и страхуемых рисков договор заключается по определенному виду страхования (например, страхование автотранспорта от угона, страхование домашнего имущества, добровольное медицинское страхование). При заключении договора вы фиксируете дату заключения, страховую сумму, вид страхования, тарифную ставку и филиал, в котором заключался договор.

**Таблицы:**

**Договоры** (Номер договора, Дата заключения, Страховая сумма, Тарифная ставка, Код филиала, Код вида страхования).

**Вид страхования** (Код вида страхования, Наименование).

**Филиал** (Код филиала, Наименование филиала, Адрес, Телефон).

**Развитие постановки задачи**

Нужно учесть, что договоры заключают страховые агенты. Помимо информации об агентах (фамилия, имя, отчество, адрес, телефон), нужно еще хранить филиал, в котором работают агенты. Кроме того, исходя из базы данных, нужно иметь возможность рассчитывать заработную плату агентам. Заработная плата составляет некоторый процент от страхового платежа (страховой платеж – это страховая сумма, умноженная на тарифную ставку). Процент зависит от вида страхования, по которому заключен договор.

Внести в структуру таблиц изменения, учитывающие эти факты, и изменить существующие запросы. Добавить новые запросы.

Создание таблиц:

create table Contracts  
(  
 ContractID int generated always as identity,  
 CreationDate DATE,  
 Sum int,  
 TariffRate int,  
 BranchID int,  
 InsuranceTypeID int,  
  
 constraint PK\_Contract PRIMARY KEY (ContractID),  
 constraint FK\_InsuranceTypeID FOREIGN KEY (InsuranceTypeID) REFERENCES InsuranceTypes (InsuranceTypeID),  
 constraint FK\_BranchID FOREIGN KEY (BranchID) REFERENCES Branches (BranchID)  
);  
  
create table InsuranceTypes  
(  
 InsuranceTypeID int generated always as identity,  
 Name varchar(50),  
  
 constraint PK\_InsuranceType PRIMARY KEY (InsuranceTypeID)  
);  
  
create table Branches  
(  
 BranchID int generated always as identity,  
 Name varchar(50),  
 Address varchar(50),  
 Phone varchar(10),  
  
 constraint PK\_Branch PRIMARY KEY (BranchID)  
);  
  
create or replace type InsuranceAgent as object  
(  
 FirstName varchar(50),  
 LastName varchar(50),  
 Patronymic varchar(50),  
 Address varchar(50),  
 Phone varchar(10),  
 BranchID int  
);  
  
create table InsuranceAgents of InsuranceAgent;

create or replace type InsuranceAgent as object  
(  
 FirstName varchar(50),  
 LastName varchar(50),  
 Patronymic varchar(50),  
 Address varchar(50),  
 Phone varchar(10)  
);  
drop table InsuranceAgents;  
create table InsuranceAgents  
(  
 AgentID int generated always as identity,  
 BranchID int,  
 AgentData InsuranceAgent NOT NULL,  
  
 constraint InsuranceAgent\_Phone check (regexp\_like(AgentData.PHONE, '^[[:digit:]]{9}$'))  
);

create function GetAgentSalary(agent\_ID int) return int  
 is  
 result int;  
begin  
 select Sum(C.TariffRate) \* Sum(C.Sum)  
 into result  
 from InsuranceAgents IA  
 join BRANCHES B on IA.BranchID = B.BRANCHID  
 join Contracts C on B.BranchID = C.BranchID  
 where IA.AgentID = agent\_ID  
 group by c.Sum, C.TariffRate;  
  
 return result;  
end;

insert into INSURANCETYPES(NAME)  
values ('AnotherInsuranceType');

insert into BRANCHES (NAME,  
 ADDRESS,  
 PHONE)  
values ('SomeBranch',  
 'SomeAddress',  
 '076738767');

insert into CONTRACTS(CreationDate,  
 Sum,  
 TariffRate,  
 BranchID,  
 InsuranceTypeID)  
values (TO\_DATE('02/05/2020', 'DD/MM/YYYY'),  
 50000,  
 8,  
 1,  
 2);

insert into INSURANCEAGENTS (BRANCHID,  
 AgentData)  
values (1,  
 new INSURANCEAGENT( 'Wasea', 'Pupkin', 'Ivanovi4', 'Derebasowskaya','098098098'));