**СОДЕРЖАНИЕ**

[1 ЦЕЛИ И ЗАДАЧИ ПРОЕКТА. ПРОЕКТИРОВАНИЕ БД 2](#_Toc136463069)

[Цели системы: 2](#_Toc136463070)

[Задачи системы: 2](#_Toc136463071)

[Концептуальная модель: объекты и их свойства 2](#_Toc136463072)

[Основные группы пользователей и их задачи 3](#_Toc136463073)

[Схема БД 4](#_Toc136463074)

[Ограничения по сущностям: 4](#_Toc136463075)

[Ограничения целостности, которые не указаны в таблицах отношений: 4](#_Toc136463076)

[Таблицы БД 5](#_Toc136463077)

[Описание групп пользователей и прав доступа 10](#_Toc136463078)

[2 СОЗДАНИЕ БАЗЫ ДАННЫХ 11](#_Toc136463079)

[Скрипты для создания БД 11](#_Toc136463080)

[Запросы к БД 16](#_Toc136463081)

[Индексы: 18](#_Toc136463082)

## ЦЕЛИ И ЗАДАЧИ ПРОЕКТА. ПРОЕКТИРОВАНИЕ БД

### Цели системы:

* Упрощение процесса оформления страховых полисов для клиентов, что может привести к увеличению числа продаж и улучшению отношений с клиентами.
* Автоматизация процесса рассмотрения и урегулирования страховых случаев, что позволит компании повысить эффективность работы и сократить время на обработку заявок.
* Разработка аналитических инструментов для оценки рисков и принятия решений по страхованию, что поможет компании предоставлять более точные и выгодные условия страхования для клиентов.

### Задачи системы:

1. Хранение и управление информацией о клиентах и их страховых полисах, что поможет компании легко отслеживать и обрабатывать заявки на страховые выплаты.
2. Управление финансовыми операциями, а именно выставление счетов и оплату страховых премий, что поможет компании следить за финансовым состоянием и управлять доходами и расходами.
3. Упрощение взаимодействия между различными отделами и сотрудниками компании, что позволит повысить координацию и эффективность работы.

### Концептуальная модель: объекты и их свойства

* **Страхователь**: Атрибуты: ФИО Страхователя, Серия, номер паспорта, кем и когда выдан, дата рождения, пол, место работы, контактные данные, Адрес Регистрации, серия, номер Полиса;
* **Страховщик**: Атрибуты: Наименование юр. лица, ИНН, БИК, Адрес;
* **Полис**: Атрибуты: Дата оформления, серия, номер полиса, срок действия (начало и конец срока действия), вид страхования, Статус (выплачена компенсация, пока не пригодился), ФИО представителя Страховой Компании, заключающего договор страхования; Данные страхователя; Данные застрахованного.
* **Застрахованный**: Атрибуты: ФИО застрахованного лица, Серия, номер паспорта, кем и когда выдан, дата рождения, пол, место работы, контактные данные, Адрес Регистрации, серия, номер Полиса, доля выплат;
* **Выгодоприобретатель**: тип (по дожитию, по смерти, иной случай), ФИО, Серия, номер паспорта, кем и когда выдан, дата рождения, место работы, контактные данные, Адрес Регистрации, доля страховых выплат, Серия, номер полиса;
* **Страховые расчеты**: Атрибуты: Страховая сумма, Страховой взнос, Страховые выплаты *(в случае смерти, инвалидности, болезни, несчастных случаев)*; серия, номер полиса, *которому принадлежат расчеты (в данном случае вклад может внести любой, а выплаты получить – застрахованный или выгодоприобретатель*);

### Основные группы пользователей и их задачи

Должностные лица СК:

* Оформление нового договора
* Получение сведений о текущих полисах
* Продление или расторжение договора (изменение срока действия в полисе)
* Получение сведений об оформленных полисах за определенный срок

Начальство СК:

* Получение сведений о текущих полисах
* Получение сведений об оформленных полисах за определенный срок
* Статистика финансовых расчетов

Страхователь

* Получение сведений об оформленном полисе

Застрахованный, Выгодоприобретатель

* Получение сведений об оформленном полисе

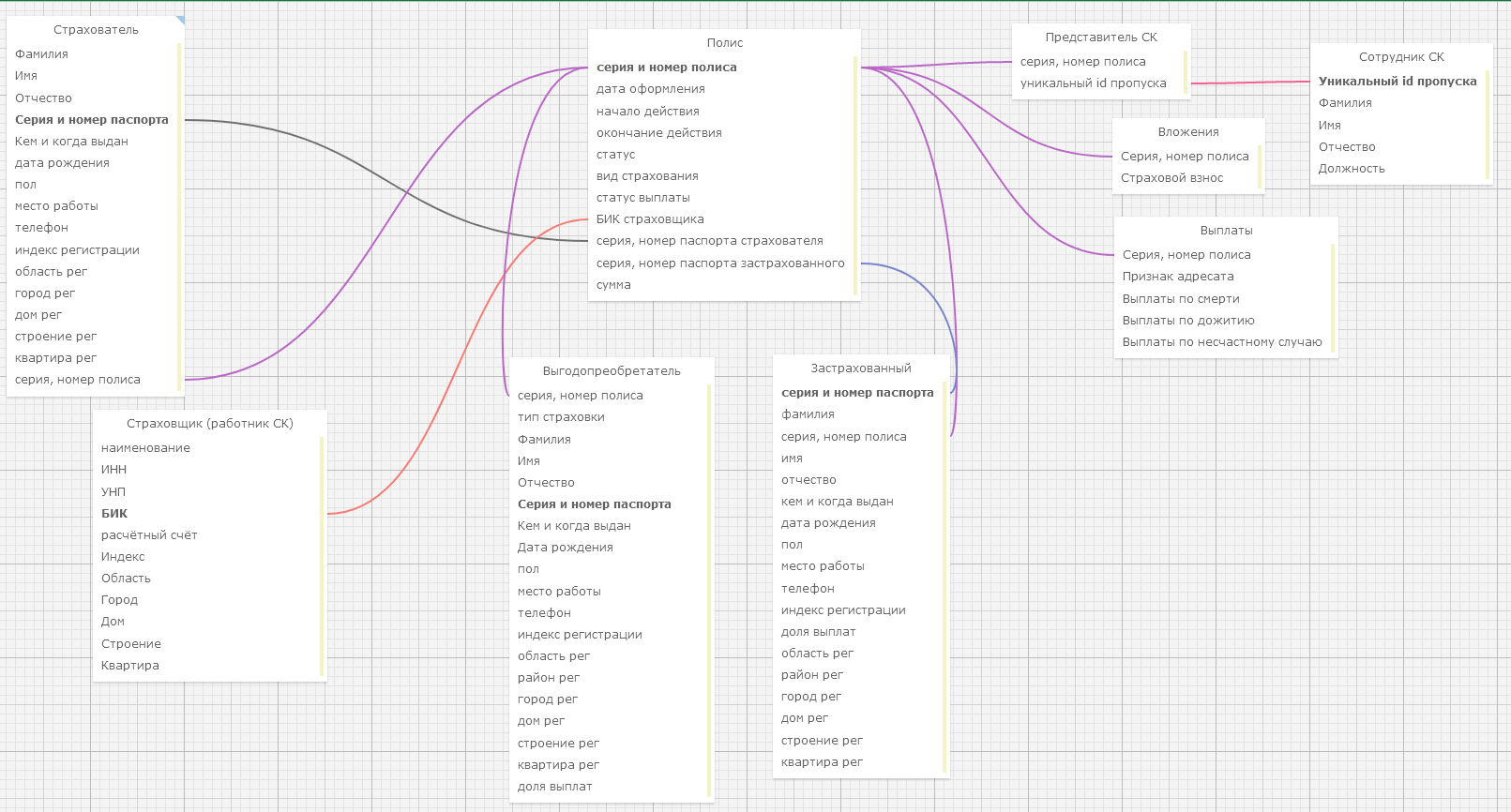
Бухгалтерия СК:

* Просмотр текущих Полисов
* Внесение изменений в таблицу страховых расчетов
* Статистика финансовых расчетов

Администратор БД

* Все права

### Схема БД



### Ограничения по сущностям:

1. Страховщик – он же Страховая компания (юридическое лицо) – может заключать несколько разных типов договоров Страхования со Страхователями (их может быть несколько)
2. В случае заключения Договора страхования Страховщик должен оформить **Полис**;
3. Каждый Страхователь может заключить Договор **на нескольких человек** (Застрахованных), Страхователей может быть несколько на 1 застрахованного и наоборот (если выбраны разные виды Страхования), но на один полис только один Страхователь;
4. Каждый Застрахованный вправе назначить Выгодоприобретателей (например, в случае смерти Застрахованного вся страховая сумма (или %) достается определенному Выгодоприобретателю, им же может служить Страхователь);
5. Различные полисы включают в себя Различные страховые суммы и выплачиваются Застрахованному в случае, описанном в Договоре Страхования;

### Ограничения целостности, которые не указаны в таблицах отношений:

Дата рождения людей > 1950 года

Дата оформления = текущий день

Начало действия полиса >= Текущий день + 1

Конец действия полиса >=Текущий день +6 (минимум 7 дней)

### Таблицы БД

Text – T, Count – C, Date – D.

**Таблица страховщик – InsurerOrg**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Наименование | Name | T | 30 | Обязательные поля |
| Идентификационный номер | IN | T | 14 |  |
| УНП (учётный номер плательщика) | UNP | T | 10-12 |  |
| БИК (банковский идентиф код) | BIK | C | 9 | Первичный ключ |
| Расчётный счет | Schet | T | 20 |  |
| Индекс | Index\_ad | C | 6 |  |
| Область | Area\_ad | T | 20 |  |
| Город | Town\_ad | T | 20 |  |
| Дом | House\_ad | T | 10 | Обязательные поля. Заполняется в зависимости от вида страхования. **Для строения и квартиры может быть NULL по умолчанию** |
| Строение | Building\_ad | T | 10 |
| Квартира | Flat\_ad | T | 10 |

**Таблица Страхователь – Insurer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Фамилия | Surname | t | 20 |  |
| Имя | FirstName | t | 15 |  |
| Отчество | Patronimic | t | 15 |  |
| Серия и номер паспорта | PassR | с | 10 | Первичный ключ |
| Кем и когда выдан | WhoPass | t | 50 |  |
| Дата рождения | Birthdate | d |  |  |
| Пол | Sex | c | 1 | М или Ж |
| Работа текущая | WorkCurr | T | 30 |  |
| Телефон | Phone | T | 20 |  |
| Индекс регистрации | Index\_reg | C | 6 |  |
| Область рег | Area\_reg | T | 20 |  |
| Город рег | Town\_reg | T | 20 |  |
| Дом рег | House\_reg | T | 10 |  |
| Строение рег | Building\_reg | T | 10 | По умолчанию может быть NULL |
| Квартира рег | Flat\_reg | T | 5 |
| Серия и номер полиса | Pol\_num | с | 25 | Внешний ключ |

**Застрахованный – Insured**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Фамилия | Surname | t | 20 |  |
| Имя | FirstName | t | 15 |  |
| Отчество | Patronimic | t | 15 |  |
| Серия и номер паспорта | PassD | с | 10 | Первичный ключ |
| Кем и когда выдан | WhoPass | t | 50 |  |
| Дата рождения | Birthdate | d |  |  |
| Пол | Sex | c | 1 | М или Ж |
| Работа текущая | WorkCurr | t | 30 |  |
| Телефон | Phone | t | 20 |  |
| Индекс регистрации | Index\_reg | c | 6 |  |
| Область рег | Area\_reg | t | 20 |  |
| Район рег | Region\_reg | T | 20 |  |
| Город рег | Town\_reg | T | 20 |  |
| Дом рег | House\_reg | T | 10 |  |
| Строение рег | Building\_reg | T | 10 | По умолчанию может быть NULL |
| Квартира рег | Flat\_reg | T | 5 |
| Серия и номер полиса | Pol\_num | с | 25 | Внешний ключ |

**Выгодоприобретатель – Benificiary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Фамилия | Surname | t | 20 |  |
| Имя | FirstName | t | 15 |  |
| Отчество | Patronimic | t | 15 |  |
| Серия и номер паспорта | Pass | с | 10 | Первичный ключ |
| Кем и когда выдан | WhoPass | t | 50 |  |
| Дата рождения | Birthdate | d |  |  |
| Пол | Sex | c | 1 | М или Ж |
| Работа текущая | WorkCurr | t | 30 |  |
| Телефон | Phone | t | 20 |  |
| Индекс регистрации | Index\_reg | c | 6 |  |
| Область рег | Area\_reg | t | 20 |  |
| Район рег | Region\_reg | T | 20 |  |
| Город рег | Town\_reg | T | 20 |  |
| Дом рег | House\_reg | T | 10 |  |
| Строение рег | Building\_reg | T | 10 | По умолчанию может быть NULL |
| Квартира рег | Flat\_reg | T | 5 |
| Доля выплат | Percent | Float unsigned | 5 | Обязательное поле |
| Серия и номер полиса | Pol\_num | с | 25 | Внешний ключ |

**Полис ­– Policy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Серия и номер полиса | Pol\_num | С | 25 | Первичный ключ |
| Дата оформления | DataAcceptation | D |  |  |
| Начало действия | EffectiveDate | D |  |  |
| Конец действия | ExpirationDate | D |  |  |
| Статус | Stat | T | 20 | Произвелась ли выплата  “использованы”  “пока не пригодились” |
| Вид/цель страхования | TypeIns | T | 15 | Travel, life. Student, home |
| БИК страховщика | BIK | C | 9 | **Внешний ключ** |
| Серия, номер паспорта **страхователя** | PassR | C | 10 | **Внешний ключ** Страхователь и застрахованные могут быть разными людьми (мама и ребёнок, дети страхуют родителей) |
| Серия и номер паспорта **застрахованного** | PassD | C | 10 |
| Сумма | SumPolicy | Float unsigned | 15 |  |

**Представитель СК(Агент) – Work\_Policy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Серия, номер полиса | Pol\_num | c | 25 | Внешний ключ |
| Уникальный Id пропуска | id | с | 16 | Внешний ключ |

**Сотрудник СК – worker\_SK**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Фамилия | Surname | T | 20 | Обязательные поля |
| Имя | FirstName | T | 15 |
| Отчество | Patronimic | T | 15 |
| Должность | Work\_dep | T | 30 |  |
| Уникальный пропуск | id | c | 16 | Первичный ключ |

**Вложения – put\_money**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Страховой взнос | Fee | Float unsigned | 20 | Обязательное поле |
| Серия, номер полиса | Pol\_num | c | 25 | Внешний ключ |

**Выплаты – Payment**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип | Длина поля | примечания |
| Серия, номер полиса | Pol\_num | c | 25 | Внешний ключ |
| Признак адресата | WhoPay | с | 1 | I(Insurance)/B(Bank) |
| Выплаты по смерти | PaymentDeath | Float unsigned | 10 | Обязательные поля  (не могут быть отрицательными) |
| Выплаты по дожитию (*окончание программы накопительного или инвестиционного страхования жизни)* | PaymentSurv | Float unsigned | 10 |
| Выплаты по несчастному случаю | PaymentCase | Float unsigned | 10 |

### Описание групп пользователей и прав доступа

s – чтение данных (select);

i – добавление данных (insert);

u – модификация данных (update);

d – удаление данных(delete).

|  |
| --- |
| Таблицы |
| Группы пользователей (роли) | | | | | |
| Должн. лица СК | Начальство СК | Страх. | Застрах. Выгодопре. | Бух. СК | Админ БД |
| Страховщик | S | SUID | S | S | S | SUID |
| Страхователь | SIUD | S | - | - | S | SUID |
| Застрахованный | SUID | S | - | - | S | SUID |
| Выгодоприобретатель | SUID | S | - | - | S | SUID |
| Представитель СК | SIU | SUID | - | - | S | SUID |
| Сотрудник СК | S | SUID | - | - | S | SUID |
| Вложения | S | S | - | - | SUID | SUID |
| Выплаты | S | S | - | - | SUID | SUID |
| Доля выплат | S | S | - | - | SUID | SUID |

## 2 СОЗДАНИЕ БАЗЫ ДАННЫХ

### Скрипты для создания БД

CREATE TABLE InsurerOrg

(

Name varchar(30) not null,

INN varchar(14) not null,

BIK char(9) not null primary key,

Schet varchar(20) not null,

Index\_ad char(6) not null,

Area\_ad varchar(20) not null,

Town\_ad varchar(20) not null,

House\_ad varchar(10) not null,

Building\_ad varchar(10) not null default 0,

Flat\_ad varchar(5) not null default 0

);

CREATE TABLE Insurer (

Surname varchar(15) not null,

FirstName varchar(15) not null,

Patronomic varchar(15) not null,

PassR char(10) primary key,

WhoPass varchar(50) not null,

Birthdate date not null check (Birthdate > '1949-12-31'),

Sex char(1) not null check (Sex IN ('w', 'm')),

WorkCurr varchar(50) not null,

Phone varchar(20) not null,

Index\_reg char(6) not null,

Area\_reg varchar(20) not null,

Town\_reg varchar(20) not null,

House\_reg varchar(10) not null,

Building\_reg varchar(10) default 'empty' not null,

Flat\_reg varchar(5) default 'empty' not null

);

CREATE TABLE Insured

(

Surname varchar(15) not null,

FirstName varchar(15) not null,

Patronomic varchar(15) not null,

PassD char(10) primary key not null,

WhoPass varchar(50) not null,

Birthdate date not null check (Birthdate> '1949-12-31'),

Sex char(1) not null check (Sex IN('w','m')),

WorkCurr varchar(100) not null,

Phone varchar(20) not null,

Index\_reg char(6) not null,

Area\_reg varchar(20) not null,

Town\_reg varchar(20) not null,

House\_reg varchar(10) not null,

Building\_reg varchar(10) not null default('empty'),

Flat\_reg varchar(5) not null default('empty')

);

CREATE TABLE Policy

(

Pol\_num char(25) not null,

DateAcceptation date not null default current\_date,

EffectiveDate date not null check (EffectiveDate > current\_date),

ExpirationDate date not null check (ExpirationDate > (current\_date + interval '5 days')),

Stat varchar(20) not null check (Stat IN ('payments used', 'payment until useful')),

TypeIns varchar(10) not null check (TypeIns IN ('travel', 'life', 'student', 'home')),

BIK char(9) references InsurerOrg,

PassR char(10) references Insurer on UPDATE cascade,

PassD char(10) references Insured on UPDATE cascade,

SumPolicy numeric(20,2) not null,

PRIMARY KEY (Pol\_num),

FOREIGN KEY (BIK) REFERENCES InsurerOrg(BIK) ON UPDATE cascade,

FOREIGN KEY (PassR) REFERENCES Insurer(PassR) ON UPDATE cascade,

FOREIGN KEY (PassD) REFERENCES Insured (PassD) ON UPDATE cascade

);

CREATE TABLE Beneficiary

(

Surname varchar(15) not null,

FirstName varchar(15) not null,

Patronomic varchar(15) not null,

Pass char(10) PRIMARY KEY not null,

WhoPass varchar(50) not null,

Birthdate date not null check (Birthdate>'1949-12-31'),

Sex char(1) not null check (sex IN('w','m')),

WorkCurr varchar(100) not null,

Phone varchar(20) not null,

Index\_reg char(6) not null,

Area\_reg varchar(20) not null,

Town\_reg varchar(20) not null,

House\_reg varchar(10) not null,

Building\_reg varchar(10) not null default('empty'),

Flat\_reg varchar(5) not null default('empty'),

percent float(6) not null,

Pol\_num char(25) references Policy

);

CREATE TABLE worker\_SK

(

Surname varchar(15) not null,

FirstName varchar(15) not null,

Patronimic varchar (15) not null,

Work\_dep varchar(30) not null,

Id char(16) PRIMARY KEY not null

);

CREATE TABLE Work\_policy

(

Id char(16),

Pol\_num char(25),

FOREIGN KEY (Id) REFERENCES worker\_SK (Id),

FOREIGN KEY (Pol\_num) REFERENCES Policy (Pol\_num),

PRIMARY KEY (Id, Pol\_num)

);

CREATE TABLE put\_money

(

Id serial PRIMARY KEY,

Fee numeric(20, 2) NOT NULL,

Pol\_num char(25) REFERENCES Policy (Pol\_num),

CONSTRAINT fk\_pol\_num FOREIGN KEY (Pol\_num) REFERENCES Policy (Pol\_num)

);

CREATE TABLE Payment

(

Pol\_num char(25) REFERENCES Policy (Pol\_num),

WhoPay char(1) DEFAULT 'I' CHECK (WhoPay IN ('I', 'B')),

PaymentDeath numeric(10, 2) NOT NULL,

PaymentSurv numeric(10, 2) NOT NULL,

PaymentCase numeric(10, 2) NOT NULL,

CONSTRAINT pk\_payment PRIMARY KEY (Pol\_num)

);

### Запросы к БД

1. Список действующих полисов:

CREATE VIEW curr\_policy AS

SELECT \*

FROM Policy

WHERE (EffectiveDate <= CURRENT\_DATE) AND (ExpirationDate >= CURRENT\_DATE);

2. Список оформленных полисов за последние 30 дней:

CREATE VIEW Days\_policy AS

SELECT \*

FROM Policy

WHERE (DateAcceptation <= CURRENT\_DATE) AND (DateAcceptation >= CURRENT\_DATE - INTERVAL '30' DAY);

3. Просмотр всех страховых взносов, внесенных по полисам за последние 30 дней:

CREATE VIEW Days\_Fee AS

SELECT p.Fee, i.Surname, i.FirstName, i.Patronimic, pol.DateAcceptation

FROM put\_money p

JOIN Policy pol ON p.NumberSer = pol.NumberSer

JOIN Insured i ON pol.PassD = i.Pass

WHERE (pol.DateAcceptation <= CURRENT\_DATE) AND (pol.DateAcceptation >= CURRENT\_DATE - INTERVAL '30' DAY);

4. Просмотр всех страховых выплат, рассчитанных за последние 30 дней:

CREATE VIEW Days\_Pay AS

SELECT p.PaymentDeath, p.PaymentSurv, i.Surname, i.FirstName, i.Patronimic, pol.DateAcceptation

FROM Payment p

JOIN Policy pol ON p.NumberSer = pol.NumberSer

JOIN Insured i ON pol.PassD = i.Pass

WHERE (pol.DateAcceptation <= CURRENT\_DATE) AND (pol.DateAcceptation >= CURRENT\_DATE - INTERVAL '30' DAY);

5. Список действующих полисов, по которым уже произведены выплаты:

CREATE VIEW Pays\_policy AS

SELECT \*

FROM Policy

WHERE Stat = 'payments used' AND (EffectiveDate <= CURRENT\_DATE) AND (ExpirationDate >= CURRENT\_DATE);

6. Сколько полисов оформили сотрудники за 30 дней:

CREATE VIEW Num\_Policy AS

SELECT w.Surname || ' ' || w.FirstName || ' ' || w.Patronimic AS Name, dep.Work\_dep, COUNT(CASE WHEN p.NumberSer=dep.SerNum THEN 1 ELSE NULL END) AS NumPol

FROM Work\_policy dep

JOIN worker\_SK w ON w.id = dep.id

JOIN Policy p ON p.NumberSer = dep.SerNum

GROUP BY dep.Work\_dep, w.Surname, w.FirstName, w.Patronimic;

7. Просмотр своего полиса (Страхователь, Застрахованный, Выгодоприобретатель):

CREATE VIEW My\_Policy AS

SELECT p.\*

FROM Policy p

LEFT JOIN Insurer r ON p.PassR = r.Pass

LEFT JOIN Insured d ON p.PassD = d.Pass

LEFT JOIN Beneficiary b ON p.NumberSer = b.NumberSer

WHERE (r.Pass = 'user' OR d.Pass = 'user' OR b.Pass = 'user') AND (EffectiveDate <= CURRENT\_DATE) AND (ExpirationDate >= CURRENT\_DATE);

8. Список полисов, по которым уплачены взносы № (действующих):

CREATE VIEW Sum\_FeeYes AS

SELECT p.NumberSer, p.SumPolicy, SUM(put.Fee)

FROM Policy p

JOIN put\_money put ON p.NumberSer = put.NumberSer

WHERE (p.EffectiveDate <= CURRENT\_DATE) AND (p.ExpirationDate >= CURRENT\_DATE) AND (SUM(put.Fee) = p.SumPolicy)

GROUP BY p.NumberSer, p.SumPolicy;

9. Список полисов, по которым НЕ уплачены взносы № (действующих):

CREATE VIEW Sum\_FeeNo AS

SELECT p.NumberSer, p.SumPolicy, SUM(put.Fee)

FROM Policy p

JOIN put\_money put ON p.NumberSer = put.NumberSer

WHERE (p.EffectiveDate <= CURRENT\_DATE) AND (p.ExpirationDate >= CURRENT\_DATE) AND (SUM(put.Fee) < p.SumPolicy)

GROUP BY p.NumberSer, p.SumPolicy;

### Индексы:

Индексы применяются в первую очередь для оптимизации производительности базы данных

Create index Insurer\_Pass on Policy (PassR);

Create index Insured\_Pass on Policy (PassD);

Create index NumberSer1 on put\_money (NumberSer);

Create index NumberSer2 on Payment (NumberSer);

Create index NumberSer3 on Beneficiary (NuberSer);

Create index NumberSer4 on Work\_policy (SerNum);