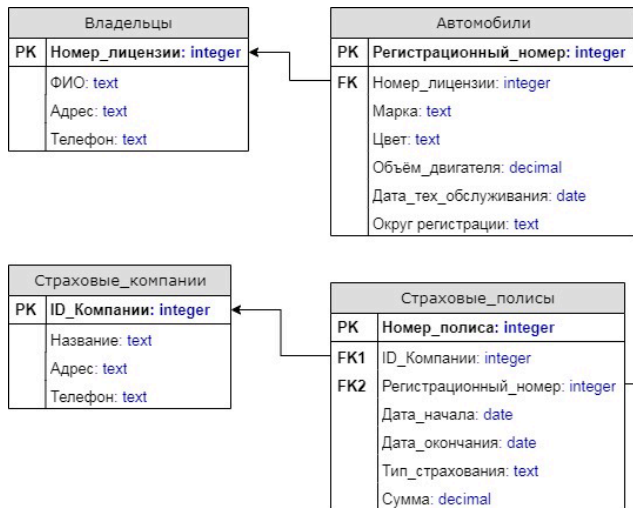


Вариант 3 (ГИБДД)



1. Написать команды создания таблиц заданной схемы с указанием необходимых ключей и ограничений. Должны быть установлены все ограничения первичного и внешних ключей. Все ограничения должны быть именованными (для первичных ключей имена должны начинаться с префикса «PK_», для вторичного ключа – «FK_», проверки - «CH_»). Все имена полей и типы данных должны полностью соответствовать схеме (до языка и регистра).

Ограничения: сумма страхового полиса не может быть отрицательной; объём двигателя не может быть отрицательным или больше 10000 литров; значение поля сумма не может быть отрицательным; значение null допустимо только в поле дата окончания.

Для каждой таблицы должна быть возможна вставка карточки без указания первичного ключа.

2. Заполнить созданные таблицы данными, 5-10 записей для каждой таблицы.

3. Написать запросы. Устранить дублирование только для тех случаев, где это потенциально возможно. Каждый запрос должен возвращать не менее одного карточка, при необходимости данные добавлять.

3.1. Вывести номер лицензии и автомобилей марки «Ford» с объёмом

двигателя больше 1.5 литра, страховка которых истекает в ближайшее две недели. Результат упорядочить по объёму двигателя. В выборке должны присутствовать только следующие атрибуты: Номер лицензии, марка автомобиля, объёмом двигателя, число полных дней до окончания страхового периода.

3.2. Для каждого владельца найти истёкшие страховки, оформленные на срок менее 2 недель. В выборке должны присутствовать только следующие атрибуты: Номер лицензии и ФИО владельца машины, название компании, количество дней страхования. Результат упорядочить по ФИО. Владелец, у которых нет таких страховок не выводить.

3.3. Сформировать статистику по маркам машин. В выборке должны присутствовать следующие атрибуты: марка машины, число машин, средняя цена страхового полиса, средний срок страхования в днях для страховок ограниченного срока действия.

3.4. Для каждой страховой компании найти среднюю сумму страхования, количество действующих страховок, количество застрахованных автомобилей, количество владельцев более чем одной страховки. В расчёт должны быть включены только действующие на данный момент страховки (страховок ограниченного срока действия). В выборке должны присутствовать только следующие атрибуты: Название компании, средняя сумма страхования, количество действующих страховок, количество застрахованных автомобилей, средний возраст владельцев страховок.

3.5. Выбрать действующие страховые полисы, у которых сумма страховке выше средней или ниже средней по марке автомобиля на 15%. В выборке должны присутствовать только следующие атрибуты: марка автомобиля, номер страхового полиса, цена страховки, средняя цена страховки по марке, отклонение суммы страховки от среднего.

3.6. Найти автомобили, на данный момент нигде не застрахованные. В выборке должны присутствовать только следующие атрибуты: регистрационный номер автомобиля, ФИО владельца.

4. Написать запросы на изменение данных.

4.1. Увеличить страховую сумму в 1,5 раза для автомобилей с объёмом двигателя более 2,5 литра

4.2. Удалить сведения о страховых компаниях, не пропавших ни одного страхового полиса.

4.3. Вынести цвета автомобилей в отдельный справочник.

5. Создать представления.

5.1. Оформить запросы 3.5 - 3.6 в виде представления.

5.2. Создать представление со следующими атрибутами: Номер лицензии, ФИО, телефон, марка машины, число дней с момента тех. осмотра, номер действующего полиса, тип и сумма страховки. Выборка должна включать только машины с действующими на данный момент полисами.

6. Написать процедуры и функции, согласно условиям. Все процедуры и функции при необходимости должны включать обработчики исключений. Названия функций: F_<имя>. Формат названий процедур: P_<имя>. Написать анонимные блоки или запросы для проверки работы процедур и функций.

6.1. Написать функцию, которая для заданной страховой компании возвращает количество автовладельцев, имеющих действующие полисы (id_компании и промежуток времени – аргументы функции). Если промежуток времени не указан, считается количество за всё время.

6.2. Написать функцию, которая для каждой страховой компании возвращает среднюю длительность страхового периода в днях. Значение может рассчитываться за конкретный период и/или для конкретной марки автомобиля. Принадлежность страхового периода указанному временному промежутку определяется по дате начала, бессрочные полисы в расчёте не учитываются. Функция имеет три аргумента: id_компании, начало периода (с точностью до дня), окончание периода (с точность до дня). Только первый аргумент является обязательным. Предусмотреть вариант вызова функции без обязательных аргументов.

6.3. Написать процедуру, которая формирует список автомобилистов с истёкшим сроком страхования в заданном округе регистрации (округ регистрации – аргумент процедуры). Обратите внимание, что на автомобиль может быть оформлено несколько полисов. Формат вывода:

Список автовладельцев для проверки на <число>:

1. <ФИО автовладельца 1>. Полис <номер полиса> на <марка> рег. номер <регистрационный номер> просрочен на <число> дня.
<и т. д.>

6.4. Написать процедуру, которая выполняет копирование всех данных об указанной страховой компании, включая страховые полисы. Аргумент процедуры - id_компании. Для скопированной записи ставится отметка «копия» в поле название.

6.5. Написать один или несколько сценариев (анонимных блока) демонстрирующий работу процедур и функций из п. 1-4.

Требование:

- Включение в запрос (для функций)

- Для каждой процедуры не менее 3-х примеров работы с различными значениями аргументов.

- Комментарии для каждого сценария описывающие суть примера и результат.

7. Создать триггеры, включить обработчики исключений. Написать скрипты для проверки. При необходимости снять ограничения (если ограничение мешает проверить работу триггера).

7.1. Написать триггер, активирующийся при изменении содержимого таблицы «Страховые_полисы» и проверяющий, чтобы дата начала была меньше даты окончания и поле «сумма» заполнялось автоматически в зависимости от типа страхования и объёма двигателя, в соответствии с таблицей (таблицу задать самостоятельно).

7.2. Написать триггер, сохраняющий статистику изменений таблицы «Автомобили» в таблице «Авто_Статистика», в которой хранится дата изменения, тип изменения (insert, update, delete). Триггер также выводит на экран сообщение с указанием количества дней прошедших со дня последнего изменения.

7.3. Написать триггер, активирующийся при вставке в таблицу «Страховые_компании» и проверяющий наличие компании с указанным наименованием. Если такая компания уже существует, вместо вставки обновляются значения полей адрес и телефон.