

Міністерство освіти і науки України

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
АВТОМОБІЛЬНО-ДОРОЖНІЙ УНІВЕРСИТЕТ**

Методичні вказівки до виконання самостійної роботи
**"Створення у середовищі MS Access та дослідження
бази даних "Система контролю безпеки руху"**
за дисципліною "Інформаційні системи і технології"

Укладач:

Симбірський Г.Д.

Харків
ХНАДУ
2017

Введение

Целью самостоятельной работы является закрепление знаний, полученных пользователями во время изучения дисциплины "Информационные системы и технологии" на втором курсе факультета транспортных систем ХНАДУ.

В процессе выполнения самостоятельной работы на основе анализа задачи, синтеза знаний, умений предметной области и информационных систем и технологий пользователи самостоятельно по индивидуальным заданиям проектируют базу данных, исследуют и реализуют ее средствами обработки данных. С этой целью пользователям необходимо выполнить следующее:

- 1) осуществить анализ предметной области;
- 2) определить модель базы данных и ее структуру;
- 3) реализовать БД в среде MS Access;
- 4) построить схему связей в базе данных;
- 5) создать и исследовать Формы ввода информации в БД;
- 6) создать запросы с условиями выбора;
- 7) создать запросы с вычислениями;
- 8) создать диаграммы по результатам вычислений;
- 9) создать на основе запросов отчеты.

Выполнение отмеченных заданий требует от пользователей достаточного уровня знаний и навыков, приобретенных при изучении дисциплины "Компьютерная техника и программирование" (первый курс ФТС) и "Информационные системы и технологии" (второй курс ФТС).

1. Анализ предпосылок предметной области

1.1. В настоящее время на автомобильных дорогах Украины наблюдается повышенная интенсивность движения, что является результатом диспропорции между ростом автомобильного парка и сетью автомобильных дорог. В свою очередь аварийность на транспорте является следствием диспропорции между уровнем подготовки, транспортной культурой участников движения и массовостью профессии водителя.

Рост интенсивности, изменение структуры и скоростных режимов транспортных потоков, неудовлетворительное состояние дорожного покрытия, недостаточная освещенность улиц вызывают рост требований к методам управления дорожным движением и, особенно, к мероприятиям обеспечения максимально возможной безопасности движения.

Управление движением транспорта в критических ситуациях заключается в оперативном влиянии на основные характеристики транспортного потока и его распределении по сетям параллельных маршрутов в зависимости от сложившихся условий движения. Одной из критических ситуаций считается возникновение дорожно-транспортных происшествий (ДТП) - нарушений процесса дорожного движения, вызвавшие ранение или гибель людей, повреждения транспортных средств или дорожных сооружений. Следствием ДТП является ежегодный рост числа человеческих жертв и материальных убытков. Для разработки мероприятий организации дорожного движения с целью снижения количества ДТП необходимо иметь полный объем информации о ДТП, обеспечить хранение, систематизацию, обработку, анализ и оперативную отчетность этой информации.

Учету подлежат все ДТП с участием хотя бы одного транспортного средства, которое находилось в движении и привело к гибели или ранениям людей, повреждениям транспортного средства, грузов, дороги, дорожных сооружений, имущества и т. п.

Основу разработки мероприятий по предупреждению ДТП составляют достоверные сведения о них. По существующим правилам учет ДТП выполняется Государственной автоинспекцией (ГАИ); при этом все обстоятельства происшествия регистрируются в учетных карточках ДТП, составленных на основании первичных документов, оформленных дежурной группой ГАИ на месте ДТП.

Статистика ДТП ведется местными органами ГАИ и автотранспортными предприятиями (АТП). В первичных документах ГАИ регистрирует характер ДТП, факторы и обстоятельства ДТП, условия их возникновения, состояние транспортного средства, количество пострадавших, послед-

ствия. В свою очередь, АТП должны хранить полную информацию о водителях и периодически ее возобновлять.

1.2. Развитие теоретических основ и технических средств регистрации информации, ее переработки и транспортировки создало предпосылки для разработки компьютерных информационных систем контроля, анализа и прогнозирования безопасности движения. Структурная схема такой информационной системы (системы контроля безопасности движения автотранспорта - СБД) на рис. 1.

Для решения указанной информационной задачи могут быть применены **системы управления базами данных (СУБД)**. На настоящее время наиболее эффективной является составляющая пакета Microsoft Office программа Access, предназначенная именно для создания и обслуживания баз данных, для обеспечения доступа к данным и для их обработки.

1.3. База данных является организованной структурой, используемой для хранения данных, то есть любых сведений о явлениях, процессах, действиях, и т. п. Данные отображают информацию о происшествиях, которые происходят в материальном мире, и, по сути, являются зарегистрированными сигналами, возникшими в результате этих происшествий. Данные становятся информацией после обработки их соответствующими методами. СУБД MS Access содержат в своих структурах не только данные, но и методы (программные коды) их обработки, и поэтому можно утверждать, что эта СУБД предлагает широкий диапазон средств для хранения информации и эффективного управления последней.

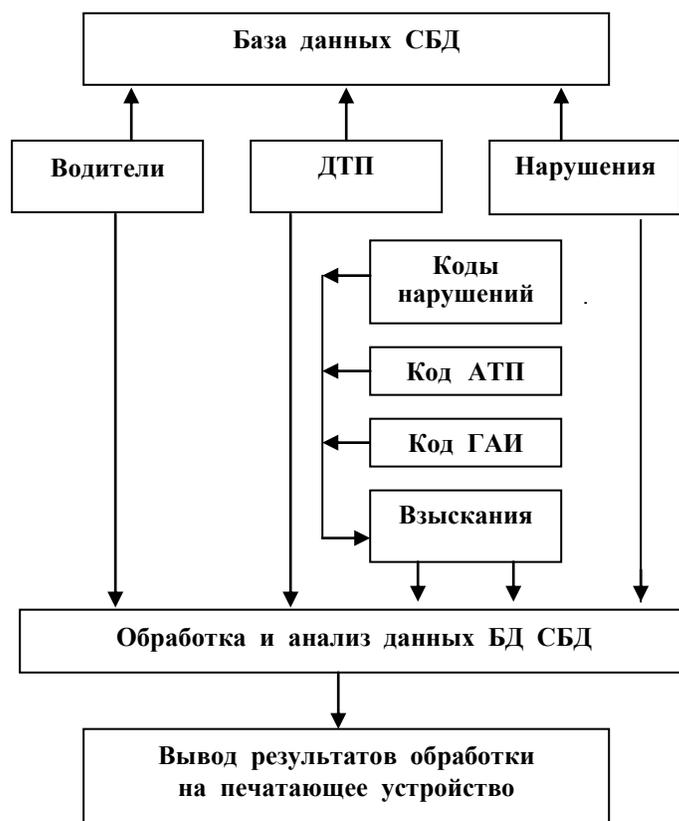


Рис. 1. Структурная схема базы данных СБД

1.4. Данные, состоящие в определенных отношениях, хранятся в двумерных таблицах; в терминах MS Access (далее - Access) столбцы названы **полями**, а строки - **записями**.

Записи имеют одинаковую структуру, и каждая запись содержит данные об одной **сущности** (например, о водителе, либо автомобиле, либо документе, и т. п.).

Поля таблицы содержат **атрибуты** (характеристики) этих сущностей (например, фамилии водителей, их адреса, марки и цены автомобилей). Каждое поле записи содержит одну характеристику

сущности и имеет единственный тип данных (например, либо текстовый, либо числовой, либо денежный и т. д.). Все записи таблицы имеют одинаковые поля, но содержат в полях **разные значения атрибутов**.

Такая модель хранения данных строится на взаимоотношениях между указанными ее составными частями и называется **реляционной** (от лат. **relatio** - отношения) **моделью данных**. Совокупность реляционных таблиц, между которыми могут быть образованы логические связи, составляет **логическую структуру реляционной базы данных**. **Сущности, атрибуты и связи** между ними представляют собой фундаментальные понятия **информационных систем**.

1.5. Для реляционной БД принципиальным является то, что каждый элемент данных нужно хранить только в одном экземпляре. В этом случае считается, что данные **нормализованы** - из определенной таблицы удалены все повторяемые данные и перенесены в новую таблицу для исключения повторений. Минимизация дублирования данных обеспечивает высокую эффективность БД, поддерживает ее в актуальном и непротиворечивом состоянии и способствует одноразовым (без повторений) вводу и коррекции данных.

Таким образом, реляционная модель данных обладает следующими свойствами:

- 1) каждая ячейка таблицы представляет собой элемент данных;
- 2) все поля в таблице однородны, то есть однотипны;
- 3) каждое поле имеет уникальное название;
- 4) одинаковые записи в таблице отсутствуют;
- 5) последовательность записей в таблице произвольная.

Основными свойствами полей таблиц БД являются такие: **название, тип и размер** (длина данных в конкретном поле).

1.6. Система управления базами данных Access позволяет работать в ее среде, как в индивидуальном режиме, так и в составе локальной или глобальной сетей. При этом Access является средой процедур, в которых предусмотрен ввод пользователем первичной информации с помощью клавиатуры, общения с персональным компьютером в диалоговом режиме, создания тех или иных запросов и получения на их основе вспомогательной информации. Реализованная в среде Access реляционная база данных **СБД** (рис. 1) содержит общую и служебную информацию о ДТП: данные о водителях, вид нарушения и взыскания, состояние здоровья участников, место, дату и виновных в ДТП, и т. п.

Система позволяет в диалоговом режиме выполнять просмотр информации, редактирование, ввод новых данных, удаление избыточных или устаревших. Среда Access предоставляет возможность анализа информации с помощью свойственных этой среде операций: разнообразных запросов по выбору, запросов по условиям, запросов с вычислениями, вывода любой информации на печатающее устройство в виде отчетов. Вся информация может быть упорядочена по любым признакам. На экран дисплея и на печатающее устройство выводятся запросы сведений с необходимым сочетанием данных. Введенная в базу данных информация может периодически обновляться (например, в конце года).

Сбор информации и ввод данных в компьютер выполняется в процессе производственного функционирования независимо от периодичности и очередности. Обмен информацией можно выполнять пересылкой файлов с помощью сети - локальной или корпоративной.

2. Структура базы данных

2.1. Во время выполнения своих служебных обязанностей работники ГАИ и АТП информацию о ДТП записывают на первичных носителях (бумажных, электронных, компьютерных) с целью последующей обработки информации. В дальнейшем эти записи о ДТП, а также сведения о водителях и другую информацию распределяют по определенным таблицам; в данном случае рассматриваются такие шесть таблиц:

- 1) **Информация** - учет водителей АТП;
- 2) **Нарушения** - данные о нарушениях водителей АТП;
- 3) **Учет ДТП** - информация о ДТП;

- 4) **Коды нарушений** - коды видов нарушений;
 5) **Коды АТП** - коды видов взысканий руководством АТП;
 6) **Коды ГАИ** - коды видов взысканий работниками ГАИ.

Указанные таблицы составляют базу данных СБД - организованную **структуру**, предназначенную для хранения и обработки информации с помощью СУБД Access. Последняя состоит из комплекса программных средств, предназначенных для создания компьютерной структуры базы данных, наполнения ее содержанием, редактирования, отображения данных, отобранных по определенному критерию, их упорядочения, оформления и последующего вывода на печатающее устройство или пересылку по каналам связи.

В таблицах 1 - 6 приведены примеры всех полей одной записи каждой из таблиц базы данных СБД. Эти примеры помогают пользователю найти, понять и реализовать собственный вариант.

В **первых** строках примеров предоставлена **сквозная нумерация** всех полей всех таблиц базы данных СБД (для ориентации при выборе пользователями собственных вариантов).

Во **вторых** строках приведены **названия** (заглавия) всех **полей** (столбцов) всех таблиц базы данных СБД.

В **третьих** строках отображены **записи** (содержание ячеек) каждого поля всех таблиц.

В **четвертых** строках расположены пояснения относительно содержания каждого поля каждой таблицы.

Таблица 1. "Информация"

1	2	3	4	5
№	ФИО	№_таб	№_ак	Д_рожд
1	Иванов Г. П.	151	1	27.06.1927
Табличный код водителя	Фамилия и инициалы водителя	Табельный номер	Номер автоколонны	Дата рождения

Таблица 1. "Информация" (продолжение)

6	7	8	9	10
Нац	Образ	Стаж	Кл	Д_раб
Украинец	Среднее	46	1	31.10.1955
Национальность	Образование	Общий стаж работы водителем	Класс водителя	Дата зачисления на работу

Таблица 1. "Информация" (продолжение)

11	12	13	14
№_удост	Кем_выд	Ав_шк.	Д_мед
ААД 809990 ВСД	ГАИ Кировограда	г. Одесса	10.02.1992
Номер удостоверения водителя	Кем выдано удостоверение	Место окончания автошколы (курсов)	Дата медицинского осмотра

Таблица 1. "Информация" (продолжение)

15	16	17	18	19
Д_экз	№_экз	Оц_экз	Адрес	ФИО_жены
11.10.1992	23	4	Харьков, ул. Чкалова, 36, кв. 52	Иванова О. Г.
Дата экзамена	Номер экзаменационного билета	Экзаменационная оценка	Домашний адрес	Фамилия и инициалы жены

Таблица 1. "Информация" (продолжение)

20	21	22	23
Раб_жены	Д_взыск	Д_аннул	ИО
База культторга	07.07.1993	07.10.1993	Георгий Петрович
Место работы жены	Дата наложения взыскания	Дата снятия взыскания	Имя и отчество водителя

Таблица 2. "Нарушения"

24	25	26	27	28
№_таб	Д_нар	Вр_нар	Тип_тс	№_тс
669	15.11.1992	22.45	ГАЗ 53	54-76
Табельный номер водителя	Дата нарушения	Время нарушения	Тип транспортного средства	Номер транспортного средства

Таблица 2. "Нарушения" (продолжение)

29	30	31	32	33
М_нар	Код_Д	Реестр	Код_А	Код_нар
г. Кировоград	3	ГАИ	3	00101
Географическое название места нарушения	Код взыскания ГАИ	Организация регистрации нарушения	Код взыскания АТП	Код вида нарушения

Таблица 3. "Учет ДТП"

34	35	36	37	38
№_таб	Зд_в	Д_дтп	Тип_тс	№_тс
669	Трезвый	09.01.1989	КАМАЗ	76-12м
Табельный номер водителя	Состояние здоровья водителя	Дата ДТП	Тип транспортного средства	Номер транспортного средства

Таблица 3. "Учет ДТП" (продолжение)

39	40	41	42
М_дтп	Код_дтп	Время_дтп	ДТП
Ул. Чкалова	00100	13.15	Наезд на пешехода
Название Места ДТП	Код вида ДТП	Время ДТП	Описание ДТП

Таблица 3. "Учет ДТП" (продолжение)

43	44	45	46	47
Повр_А	Ущерб (грн)	В_в	Код_Г	Код_А
2	0	Не виновен	0	3
Количество поврежденных автомобилей	Материальный ущерб	Виновность водителя	Код взыскания ГАИ	Код взыскания АТП

Таблица 4. "Коды нарушений"

48	49
Код_нар	Вид_нар
00101	Управление транспортом в нетрезвом состоянии

Таблица 5. "Коды взысканий АТП"

50	51
Код	АТП
0	Взыскание отсутствует

Таблица 6. "Коды взысканий ГАИ"

52	53
Код	ГАИ
0	Взыскание отсутствует

2.2. В таблице 7 выбрать и записать вариант пользователя в соответствии с двумя последними цифрами зачетной книжки. Также необходимо по примеру таблиц 8, 9 и 10 составить дополнительные таблицы с пояснениями сокращенных названий полей запросов своего варианта.

Таблица 7

Варианты заданий

Две последние цифры зачетной книжки	Первый запрос Номера полей таблицы 1: "Информация"	Второй запрос Номера полей таблиц 1 и 2: "Информация" и "Нарушения"	Третий запрос Номера полей таблиц 1 и 3: "Информация" и "Учет ДТП"
00	1, 2, 3, 23	1, 2, 3, 26	1, 2, 3, 46
01	1, 2, 3, 4	1, 2, 3, 25	1, 2, 3, 36
02	1, 2, 3, 5	1, 2, 3, 26	1, 2, 3, 37
03	1, 2, 3, 6	1, 2, 3, 27	1, 2, 3, 38
04	1, 2, 3, 7	1, 2, 3, 28	1, 2, 3, 39
05	1, 2, 3, 8	1, 2, 3, 29	1, 2, 3, 40
06	1, 2, 3, 9	1, 2, 3, 31	1, 2, 3, 42
07	1, 2, 3, 10	1, 2, 3, 33	1, 2, 3, 43
08	1, 2, 3, 11	1, 2, 3, 25	1, 2, 3, 44
09	1, 2, 3, 12	1, 2, 3, 26	1, 2, 3, 45
10	1, 2, 3, 13	1, 2, 3, 27	1, 2, 3, 46
11	1, 2, 3, 14	1, 2, 3, 28	1, 2, 3, 36
12	1, 2, 3, 15	1, 2, 3, 29	1, 2, 3, 37
13	1, 2, 3, 16	1, 2, 3, 31	1, 2, 3, 38
14	1, 2, 3, 17	1, 2, 3, 33	1, 2, 3, 39
15	1, 2, 3, 18	1, 2, 3, 25	1, 2, 3, 40
16	1, 2, 3, 19	1, 2, 3, 26	1, 2, 3, 42
17	1, 2, 3, 20	1, 2, 3, 27	1, 2, 3, 43
18	1, 2, 3, 21	1, 2, 3, 28	1, 2, 3, 44
19	1, 2, 3, 22	1, 2, 3, 29	1, 2, 3, 45
20	1, 2, 3, 23	1, 2, 3, 31	1, 2, 3, 46
21	1, 2, 3, 4	1, 2, 3, 33	1, 2, 3, 36
22	1, 2, 3, 5	1, 2, 3, 25	1, 2, 3, 37
23	1, 2, 3, 6	1, 2, 3, 26	1, 2, 3, 38
24	1, 2, 3, 7	1, 2, 3, 27	1, 2, 3, 39

25	1, 2, 3, 8	1, 2, 3, 28	1, 2, 3, 40
26	1, 2, 3, 9	1, 2, 3, 29	1, 2, 3, 42
27	1, 2, 3, 10	1, 2, 3, 31	1, 2, 3, 43
28	1, 2, 3, 11	1, 2, 3, 33	1, 2, 3, 44
29	1, 2, 3, 12	1, 2, 3, 25	1, 2, 3, 45
30	1, 2, 3, 13	1, 2, 3, 26	1, 2, 3, 46
31	1, 2, 3, 14	1, 2, 3, 27	1, 2, 3, 36
32	1, 2, 3, 15	1, 2, 3, 28	1, 2, 3, 37
33	1, 2, 3, 16	1, 2, 3, 29	1, 2, 3, 38
34	1, 2, 3, 17	1, 2, 3, 31	1, 2, 3, 39
35	1, 2, 3, 18	1, 2, 3, 33	1, 2, 3, 40
36	1, 2, 3, 19	1, 2, 3, 25	1, 2, 3, 42
37	1, 2, 3, 20	1, 2, 3, 26	1, 2, 3, 43
38	1, 2, 3, 21	1, 2, 3, 27	1, 2, 3, 44
39	1, 2, 3, 22	1, 2, 3, 28	1, 2, 3, 45
40	1, 2, 3, 23	1, 2, 3, 29	1, 2, 3, 46
41	1, 2, 3, 4	1, 2, 3, 31	1, 2, 3, 36
42	1, 2, 3, 5	1, 2, 3, 33	1, 2, 3, 37
43	1, 2, 3, 6	1, 2, 3, 25	1, 2, 3, 38
44	1, 2, 3, 7	1, 2, 3, 26	1, 2, 3, 39
45	1, 2, 3, 8	1, 2, 3, 27	1, 2, 3, 40
46	1, 2, 3, 9	1, 2, 3, 28	1, 2, 3, 42
47	1, 2, 3, 10	1, 2, 3, 29	1, 2, 3, 43
48	1, 2, 3, 11	1, 2, 3, 31	1, 2, 3, 44
49	1, 2, 3, 12	1, 2, 3, 33	1, 2, 3, 45
50	1, 2, 3, 13	1, 2, 3, 25	1, 2, 3, 46
51	1, 2, 3, 14	1, 2, 3, 26	1, 2, 3, 36
52	1, 2, 3, 15	1, 2, 3, 27	1, 2, 3, 37
53	1, 2, 3, 16	1, 2, 3, 28	1, 2, 3, 38
54	1, 2, 3, 17	1, 2, 3, 29	1, 2, 3, 39
55	1, 2, 3, 18	1, 2, 3, 31	1, 2, 3, 40
56	1, 2, 3, 19	1, 2, 3, 33	1, 2, 3, 42
57	1, 2, 3, 20	1, 2, 3, 25	1, 2, 3, 43
58	1, 2, 3, 21	1, 2, 3, 26	1, 2, 3, 44
59	1, 2, 3, 22	1, 2, 3, 27	1, 2, 3, 45
60	1, 2, 3, 23	1, 2, 3, 28	1, 2, 3, 46
61	1, 2, 3, 4	1, 2, 3, 29	1, 2, 3, 36
62	1, 2, 3, 5	1, 2, 3, 31	1, 2, 3, 37
63	1, 2, 3, 6	1, 2, 3, 33	1, 2, 3, 38
64	1, 2, 3, 7	1, 2, 3, 25	1, 2, 3, 39
65	1, 2, 3, 8	1, 2, 3, 26	1, 2, 3, 40
66	1, 2, 3, 9	1, 2, 3, 27	1, 2, 3, 42
67	1, 2, 3, 10	1, 2, 3, 28	1, 2, 3, 43
68	1, 2, 3, 11	1, 2, 3, 29	1, 2, 3, 44
69	1, 2, 3, 12	1, 2, 3, 31	1, 2, 3, 45
70	1, 2, 3, 13	1, 2, 3, 33	1, 2, 3, 46
71	1, 2, 3, 14	1, 2, 3, 25	1, 2, 3, 36
72	1, 2, 3, 15	1, 2, 3, 26	1, 2, 3, 37
73	1, 2, 3, 26	1, 2, 3, 27	1, 2, 3, 38
74	1, 2, 3, 17	1, 2, 3, 28	1, 2, 3, 39
75	1, 2, 3, 18	1, 2, 3, 29	1, 2, 3, 40
76	1, 2, 3, 19	1, 2, 3, 31	1, 2, 3, 42
77	1, 2, 3, 20	1, 2, 3, 33	1, 2, 3, 43
78	1, 2, 3, 21	1, 2, 3, 25	1, 2, 3, 44
79	1, 2, 3, 22	1, 2, 3, 26	1, 2, 3, 45
80	1, 2, 3, 23	1, 2, 3, 27	1, 2, 3, 46
81	1, 2, 3, 4	1, 2, 3, 28	1, 2, 3, 36

82	1, 2, 3, 5	1, 2, 3, 29	1, 2, 3, 37
83	1, 2, 3, 6	1, 2, 3, 31	1, 2, 3, 38
84	1, 2, 3, 7	1, 2, 3, 33	1, 2, 3, 39
85	1, 2, 3, 8	1, 2, 3, 25	1, 2, 3, 40
86	1, 2, 3, 9	1, 2, 3, 26	1, 2, 3, 42
87	1, 2, 3, 10	1, 2, 3, 27	1, 2, 3, 43
88	1, 2, 3, 11	1, 2, 3, 28	1, 2, 3, 44
89	1, 2, 3, 12	1, 2, 3, 29	1, 2, 3, 45
90	1, 2, 3, 13	1, 2, 3, 31	1, 2, 3, 46
91	1, 2, 3, 14	1, 2, 3, 33	1, 2, 3, 36
92	1, 2, 3, 15	1, 2, 3, 25	1, 2, 3, 37
93	1, 2, 3, 16	1, 2, 3, 26	1, 2, 3, 38
94	1, 2, 3, 17	1, 2, 3, 27	1, 2, 3, 39
95	1, 2, 3, 18	1, 2, 3, 28	1, 2, 3, 40
96	1, 2, 3, 19	1, 2, 3, 29	1, 2, 3, 42
97	1, 2, 3, 20	1, 2, 3, 31	1, 2, 3, 43
98	1, 2, 3, 21	1, 2, 3, 33	1, 2, 3, 44
99	1, 2, 3, 22	1, 2, 3, 25	1, 2, 3, 45

Таблица 8. Пример записи варианта задания

Две последние цифры номера зачетной книжки	Поля таблицы "Информация"	Поля таблиц "Информация" и "Нарушения"	Поля таблиц "Информация" и "УчетДТП"
	1-й запрос	2-й запрос	3-й запрос
51	1, 2, 3, 14	1, 2, 3, 26	1, 2, 3, 36, 44

Таблица 9. Пример записи пояснения названий персонифицированных полей

1	2	3
№	ФИО	№ таб
Табличный код водителя	Фамилия и инициалы водителя	Табельный номер

Таблица 10. Пример записи пояснения названий личных полей в запросах

14	25	36	44
№ ак	Д нар	Д дтп	Ущерб (грн)
Номер автоколонны	Дата нарушения	Дата ДТП	Материальные убытки

2.3. Таблицы являются основными и важнейшими **объектами** базы данных; **структуру** таблиц образуют **поля** (столбцы) и **записи** (строки).

Перед вызовом на экран составляющих любого объекта следует пометить этот объект (в данном случае **Таблицы**) нажатием на его названии ЛК мыши. Вызов на экран составляющей объекта **Таблицы**, например, таблицы **Информация**, выполняется в главном окне базы данных двойным нажатием ЛК мыши на названии таблицы.

Каждая таблица имеет следующие характеристики:

- 1) **название поля** - средство обращения к данным этого поля при автоматических операциях с базой (по умолчанию названия полей используются в качестве заголовков столбцов таблиц);
- 2) **тип поля** - тип данных, которые содержатся в данных полях;
- 3) **размер поля** - определяет количество символов в поле;
- 4) **формат поля** - способ форматирования данных в поле;
- 5) **маска ввода** - фиксированная форма, в которой вводятся данные в ячейку поля (средство автоматизации ввода данных).

2.4. Определенные поля отдельных таблиц базы данных могут быть соединены для образования связей между таблицами. Таблицы вместе со связями составляют **схему данных**. БД из связанных таблиц называются **реляционными базами данных**.

Существует несколько типов возможных связей между таблицами. Наиболее распространенными являются связи "**один ко многим**" и "**один к одному**". Связь между таблицами создается по **общим** полям. На стороне "**один**" поле должно быть **ключевым**, особенностью которого является единственное (без повторений) определение каждой записи в таблице; то есть на стороне "**один**" должно выступать **уникальное** ключевое поле. На стороне "**много**" может быть несколько записей с одинаковыми значениями в ключевых полях. За исключением связанного поля, данные ни в одном из других полей не должны повторяться. Подобная организация представления данных имеет такие преимущества:

- 1) **эффективность** - не нужно хранить повторяющиеся сведения, например, фамилию, имя и отчество;
- 2) **управление** - продуманная структура данных, которые не содержат повторяющихся сведений, что ускоряет обновление, удаление и добавление данных;
- 3) **точность** - отсутствие повторений снижает возможность возникновения ошибок; введенные один раз данные правильно отображаются во всей базе данных;
- 4) **сохранность данных** - добавление или удаление полей или записей в отдельных таблицах не вызывает изменений целостной структуры базы данных.

2.5. Формы являются средством ввода данных.

В формах можно разместить специальные элементы управления (счетчики, списки, переключатели, и т. п.) для автоматизации ввода информации. Преимущества форм раскрываются в том случае, когда происходит ввод данных с бумажных бланков. Для этого форму создают графически подобной бланку с целью оптимизации работы оператора: снижения его усталости и предотвращения появления печатных ошибок.

2.6. Запросы являются объектами для выбора данных из таблиц и представления их в удобном виде.

С помощью запросов выполняются поиск данных, их сортировка и фильтрация, преобразование данных по определенным алгоритмам, создание новых таблиц, их автоматическое наполнение импортированными из других источников данными, вычисления по табличным данным.

Особенностью запросов является то, что с их помощью выбирают данные из базовых таблиц и создают **результатирующую** таблицу, а данные базовых остаются неприкосновенными.

При сохранении информации в базовых таблицах упорядоченность записей отсутствует, а нумерация записей в поле **Счетчик** в первичных таблицах выполняется автоматически в очередности ввода записей. Если необходимо видеть данные, отсортированные или отфильтрованные по любым параметрам, можно использовать соответствующий запрос.

2.7. Отчеты являются структурами, которые предназначены для вывода данных на экран и на печатающее устройство.

В связи с этим в отчетах используются специальные мероприятия для группировки выводимых данных и специальные элементы оформления, характерные для печатных документов (верхний и нижний колонтитулы, номера страниц, и т. п.).

3. Проектирование базы данных

3.1. Студенту необходимо создать в среде Ms Access 2010 собственную базу данных с шестью таблицами, названия и структуры которых указаны в таблицах 1 - 6. Для этого нужно скачать с **диска** электронной почты файл **СБД.mdb**.

Продублируйте этот файл на случай возможных ошибок. Один из файлов переименуйте в файл со своей фамилией и номером группы, например, **СБД Петренко Т-21.mdb** и откройте его.

В первую очередь пользователю необходимо упорядочить записи. Для этого достаточно нажать ЛК мыши в любой клетке поля **Счетчик**, а затем нажать на панели инструментов **Стандарт-**

ная кнопку **Сортировка по возрастанию**.

3.2. Далее пользователю надлежит создать собственную базу данных контроля безопасности движения на АТП на **десять** водителей.

Первым в этом перечне будет водитель, порядковый номер (в поле **Счетчик**) которого отвечает двум последним цифрам зачетной книжки пользователя. Все предыдущие записи следует пометить, а затем удалить: пометить удаляемую запись (строку) следует с помощью нажатием ЛК мыши на крайней слева кнопке серого цвета и удалить помеченное последующим нажатием клавиши [**Delete**]. Для одновременного выделения нескольких записей (строк) следует пометить первую из них, нажать клавишу [**Shift**], удерживать ее и нажатием клавиши [**↓**] пометить удаляемые записи.

Запись с номером водителя, соответствующим двум последним цифрам зачетной книжки пользователя будет первой записью в создаваемой базе данных. Далее оставляется каждая **десятая(!)** запись, а промежуточные **девять** записей удаляются.

!!! В таблице **Информация** в итоге должно остаться десять записей с информацией о водителях. Оставшиеся после этого лишние записи необходимо также удалить.

К десяти записям водителей нужно добавить запись с фамилией и инициалами студента и произвольным вводом всех остальных параметров (по образцу записей о водителях).

Табельный номер в этом случае не вводится! Номер записи пользователя в поле **Счетчик** используется тот, который программа Access выводит самостоятельно. Изменению он не подлежит.

В созданной пользователем таблице **Информация** выполнить сортировку записей по **порядковым номерам**.

3.3. В соответствии с названием и назначением каждого поля созданной таблицы необходимо отредактировать (заполнить пустые и исправить ошибочные) содержание полей.

Рекомендуется принять к сведению следующие сведения о типах полей.

1) **Текстовый** - для текста, комбинации текста и чисел (например, адреса) или чисел, которые не требуют вычислений (например, номера телефонов, инвентарные номера, табельные номера, почтовые индексы). Длина текстового поля не превышает 255 символов. Для обширных текстов применяется поле типу **МЕМО**. В нем можно хранить свыше 65000 символов. Особенностью поля **МЕМО** является то, что данные хранятся не в самих полях, а в другом месте; в поле находится ссылка на место расположения текста.

2) **Числовой** - для числовых данных - целых или вещественных (десятичных) чисел. Для вещественных чисел необходимо указать количество символов десятичной части.

3) **Дата/время** - для ввода календарных и часовых данных.

4) **Логический** - для логических данных, имеющих только два значения - **Да** или **Нет (0 или 1)**.

5) **Денежный** - для денежных сумм. Их можно хранить и в числовых полях, но в денежном формате данные удобнее обрабатывать. В этом случае числа отображаются вместе с денежными единицами.

6) **Объект OLE** - для данных, позволяющих хранить не только числа и буквы, но и рисунки, музыкальные клипы, видеозаписи.

7) **Счетчик** - для нумерации записей; он является обычным числовым полем, в котором для очередного ввода записи выполняется автоматическая нумерация.

База данных имеет свойство запоминать все номера и самостоятельно присваивает новой записи очередной последний номер общего количества номеров. Новую запись нужно располагать последней. Не рекомендуется пытаться самостоятельно изменять данные этого типа.

8) **Гиперссылка** - определяет гиперссылку.

3.4. В таблицах **Нарушения** и **Учет ДТП** также нужно выполнить определенное редактирование с целью создания варианта пользователя.

Предлагается в обеих таблицах выбрать двадцать записей с табельными номерами, которые указаны в отредактированной пользователем таблице **Информация**; другие записи нужно удалить.

Внимание! При отсутствии в этих таблицах некоторых табельных номеров варианта пользова-

теля рекомендуется выбрать любую другую запись и заменить табельный номер на необходимый.

!!! Общее число записей в каждой из этих двух таблиц должно быть равным двадцати.

Принять к сведению тот факт, что в базе данных **не должно быть** в записях пустых ячеек. При отсутствии данных в некоторых полях в созданных по варианту пользователя таблицах, необходимо самостоятельно ввести отсутствующие данные в соответствии с назначением и типом поля.

В отредактированных таблицах **Нарушения** и **Учет ДТП** необходимо выполнить сортировку записей по **табельным** номерам. Сохранение отредактированных таблиц выполняется с помощью нажатия кнопки **Заккрыть** в строке заголовка каждой таблицы.

3.5. Как было отмечено выше, особенностью реляционных БД является отсутствие повторяемых данных в таблицах.

Для получения определенной информации возникает необходимость выбирать данные из нескольких таблиц. Для этого таблицы должны быть **связаны** между собой по **ключевому полю**, что позволяет однозначно **идентифицировать** каждую запись таблицы. **Связи** между таблицами дают возможность совместно использовать данные из разных таблиц.

В отредактированной таблице **Информация** базы данных пользователя таким уникальным ключевым полем следует считать поле **табельных номеров**. Именно в этом поле отсутствует дублирование записей, т. к. отсутствует дублирование табельных номеров. Значение табельного номера позволяет определять в соответствующих таблицах всю информацию, которая касается водителя с указанным табельным номером.

4. Создание связей между таблицами

4.1 Основные преимущества систем управления реляционными базами данных реализуются при работе не с отдельными таблицами, а с **группами взаимосвязанных** таблиц. Для создания связей между таблицами в среде Access существует специальный инструмент, называемый **Схемой данных**. Окно (рис. 5) этого инструмента открывается нажатием соответствующей кнопки на панели инструментов **Стандартная** или командой **Сервис→Схема данных**. Связи отображаются линиями с расположенными на них определенными метками.

В начале работы в этом окне отсутствуют какие либо элементы. Однако необходимо проверить наличие скрытых связанных таблиц. Они могли быть образованы самой программой Access по умолчанию и сохраниться в ее ядре.

Для проверки присутствия ошибочных связей нужно на панели инструментов **Стандартная** сначала нажать кнопку **Отобразить прямые связи**, а затем кнопку **Отобразить все связи**. Если в этих окнах выводятся уменьшенные копии (**иконки**) только с названиями полей некоторых таблиц с линиями связей, которые соединяют одноименные поля, необходимо все линии и иконки удалить для предотвращения искажений в создаваемой **Схеме данных**.

Для удаления линии связи нужно расположить указатель мыши на линии и нажать ПК мыши - на экран выводится список контекстных команд. Исполнить команду **Удалить** - выводится диалоговое окно с вопросом о сохранении изменений; нажать **Да** для завершения удаления.

После удаления всех связей на экране остаются иконки, в одной из которых строка заголовка активна (окрашена синим цветом). Если нажать [**Delete**], активная иконка удаляется, а активность автоматически переносится на другую. Аналогично удаляются все иконки.

4.2. Следующим шагом будет нажатие инструмента (кнопки) **Отобразить таблицу** на панели **Стандартная**. Как результат этого действия на экран выводится окно **Добавление таблицы**, предназначенное для выбора нужных таблиц и добавления их в **Схему данных** для построения межтабличных связей.

Сначала в поле **Схема данных** необходимо добавить символы всех таблиц БД. Для этого в окне **Добавление таблицы** следует пометить название таблицы **Информация** (возможно, она будет помечена по умолчанию); затем нажать левую клавишу [**Shift**], удерживать ее и нажатиями клавиши [**Insert**] пометить **все** названия таблиц. Нажатие кнопки **Добавить** в окне **Добавление таблицы** способствует выводу в окне **Схема данных** всех таблиц базы данных.

Далее необходимо расположить таблицы подобно изображению на рис. 5. При этом горизонтальные размеры должны оставаться такими, какими их предоставила среда Access; вертикальные размеры изменяют при необходимости.

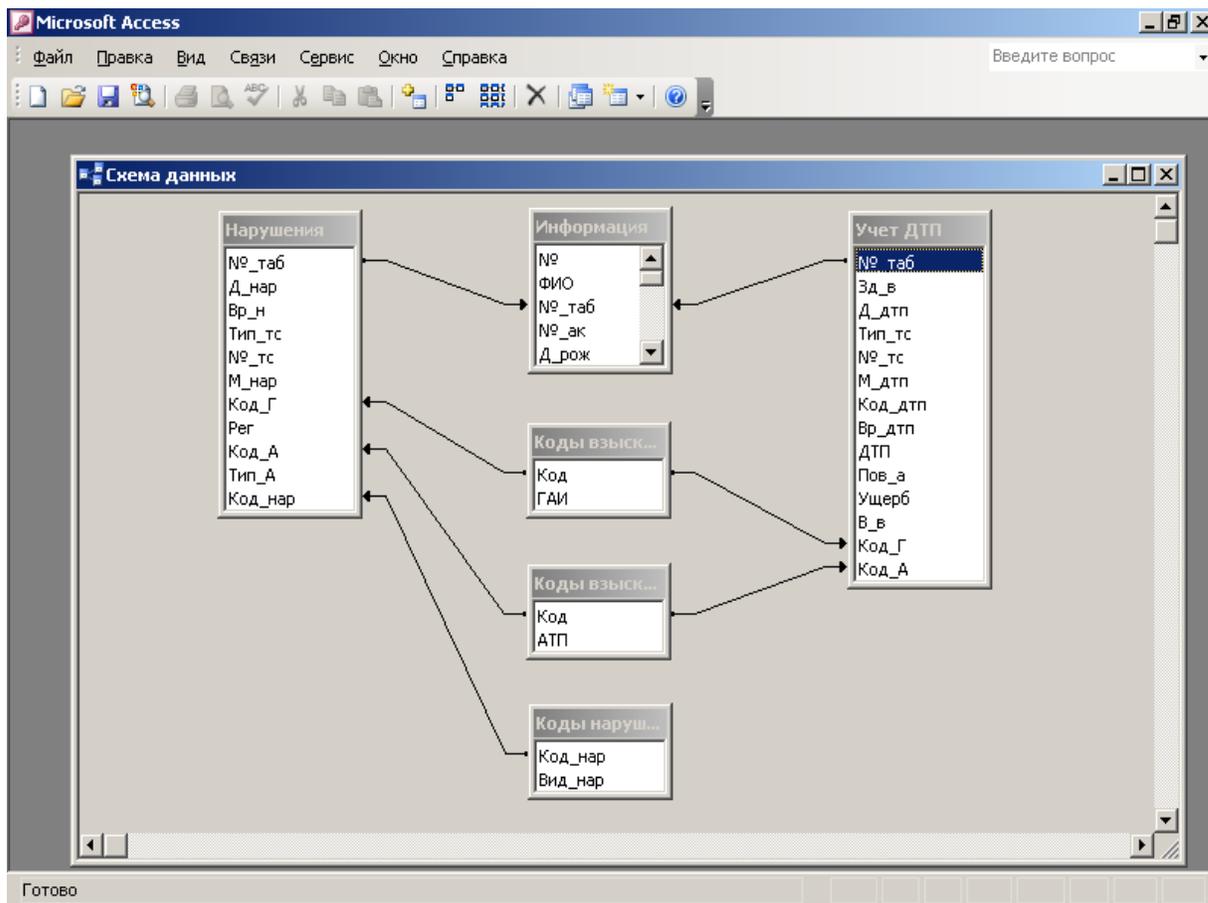


Рис. 2. Схема базы данных СБД.

4.3. Связи между таблицами образуются следующими действиями.

1) Расположить указатель мыши на ключевом поле таблицы, из которой ("откуда") поступает информация, и нажать ЛК мыши.

2) Удерживать нажатой ЛК мыши и перетянуть указатель на соответствующее поле в таблицу получения информации ("куда"); указателем мыши передвигается условное отображение названия поля.

Направления линий связей между таблицами базы данных СБД следующие (**Внимание!!! Соблюдайте направленность!!!**):

- от таблицы **Нарушения** к таблице **Информация** (по № таб);
- от таблицы **Учет ДТП** к таблице **Информация** (по № таб);
- от таблицы **Коды розысканий ГАИ** к таблице **Нарушения**;
- от таблицы **Коды розысканий АТП** к таблице **Нарушения**;
- от таблицы **Коды нарушений** к таблице **Нарушения**;
- от таблицы **Коды розысканий ГАИ** к таблице **Учет ДТП**;
- от таблицы **Коды розысканий АТП** к таблице **Учет ДТП**.

3) Отпустить ЛК мыши - на экран выводится окно **Изменение связей**, в котором кнопка **Создать** находится в активном состоянии. Обратит внимание на то, что открытием данного окна за-

вершается построение **каждой** линии связи.

4) Нажать кнопку **Объединение** - открывается очередное окно **Параметры объединения; пометить опцию с номером 2** и нажать кнопку **ОК**; происходит возвращение в окно **Изменение связей**.

5) В окне **Изменение связей** опция **Обеспечение целостности данных** пустая; установка в ней метки будет препятствовать удалению записей только в **одной** из связанных таблиц. Без такой защиты связанные (с удаляемыми) данные в других таблицах останутся без связи и будут засорять таблицы. В рассматриваемой базе данных СБД это условие не является существенным и поэтому опция **Обеспечение целостности данных** не помечается.

6) Нажать кнопку **Создать** - закрывается окно **Изменение связей**, а в окне **Схема данных** выводится линия, треугольник на конце которой отображает направление связи.

7) В соответствии с пунктами 1) - 6) и рис. 2 создать все линии связи для всех таблиц.

5. Формы и их информационные возможности

5.1. Форма является электронным бланком с областями (полями) для ввода данных. Пользователь вводит в эти поля данные, которые автоматически пересылаются в таблицы базы данных. Данные в таблицу можно вводить и без помощи форм, но существуют причины, которые делают формы необходимыми. Некоторые из них следующие.

1) Малоквалифицированному персоналу опасно предоставлять полный доступ к таблицам (самым ценным объектам в базе данных).

2) Разные сотрудники могут иметь разные права доступа к информации: один пользователь может иметь право вводить биографические и другие данные водителей, второй - данные ДТП, а третий - финансовые расчеты. Сговор между этими людьми должен быть исключен. Поэтому для ввода данных им предоставляют разные формы, но данные будут передаваться в одну таблицу, например, **Учет ДТП**.

3) Ввод данных в таблицу является достаточно напряженной работой. Уже после нескольких часов труда пользователь делает ошибки. Ввод данных в форму проще - здесь многое можно автоматизировать, а элементы управления формы настраивают так, чтобы данные при вводе проверялись.

4) Информация для БД поступает на бумажных носителях - бланках; экранную форму можно выполнить как точную копию таких бланков. Благодаря этому во много раз уменьшается количество ошибок при вводе в БД первичной информации и значительно снижается усталость персонала.

5.2. Создание форм состоит из следующих четырех этапов.

1) Выбор полей, данные для которых вводятся в форму.

В окне базы данных пометить объект **Формы**. В области справа выводятся ярлыки и названия двух способов создания форм. Дважды нажать ЛК мыши на ярлыке **Создание формы с помощью мастера** - открывается окно **Создание формы**.

В области **Таблицы и запросы** нажать кнопку со списком и пометить таблицу (например, **Информация**), которая будет источником данных для создаваемой формы.

При выборочном использовании полей в области **Доступные поля** пометить определенное поле и нажать кнопку ">" - поле пересылается в область **Выбранные поля**. Если необходимая обратная пересылка - используется кнопка "<".

Для пересылки **всех** полей в прямом или обратном направлениях нажимаются кнопки ">>" или "<<". При этом расположение полей сохраняется. Завершить работу в этом окне нажатием кнопки **Далее**.

2) Выбор внешнего вида формы.

В очередном окне выбора внешнего вида формы рекомендуется пометить опцию **в один столбец** и нажать кнопку **Далее**.

3) Выбор фоновой рисунка формы.

В следующем окне предлагается выбрать нужный **стиль** - пометить опцию выбранного стиля. Целесообразно для всех форм выбирать одинаковый стиль. Завершить работу в этом окне нажатием

кнопки **Далее**.

4) Указание названия формы.

В очередном окне в области **Задать имя формы** удалить название, предлагаемое средой Access, и ввести собственное название, состоящее из названия таблиц, по которым создается форма по варианту пользователя. Названия формы и всех других объектов базы данных должны начинаться заглавной буквой.

7) Завершить создание формы нажатием кнопки **Готово** - текущее окно закрывается и выводится окно с формой **Информация 1**. Для возвращения в главное окно базы данных следует нажать кнопку **Закреть** в верхнем правом углу окна формы.

8) Аналогично пунктам 1) - 7) создать формы для всех таблиц базы данных СБД.

5.3. Среда Access позволяет также создавать в режиме **Конструктор** формы с командными кнопками, с помощью которых выполняется переход к объектам БД и другие действия. Проектируется кнопочная форма следующими действиями (для тех студентов, у которых версия Access 2010 или Access 2003 поддерживает последующие действия).

1) В первичном окне БД СБД выполнить команды **Создание** и **Конструктор форм** - выводится пустая форма с названием **Form1** (рис. 3).

Одновременно открывается вкладка **Инструменты конструктора форм**, где во вкладке **Меню** находится кнопка **Элементы управления**, нажатие на которую открывает панель с инструментами для оформления конструируемой формы.

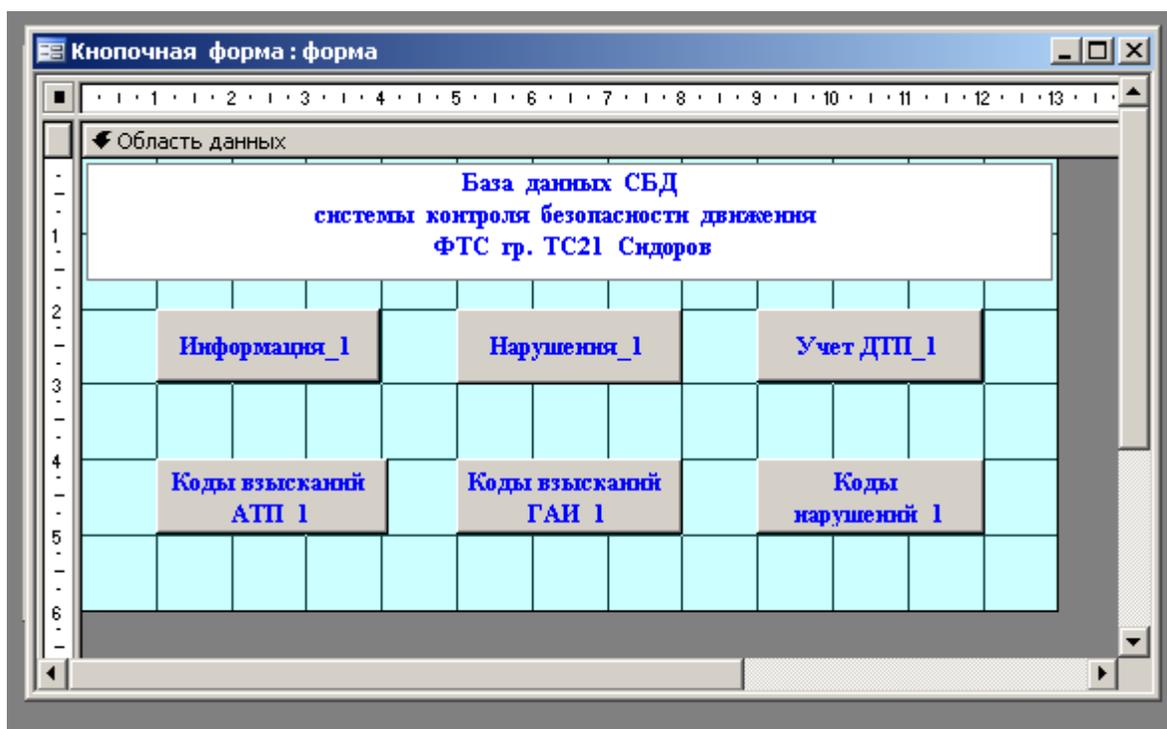


Рис. 3. Окно конструктора с кнопками управления.

2) С помощью маркеров растянуть форму указателем мыши на ширину **13** см и по высоте до **9** см. Озаглавить кнопочную форму с помощью элементов управления (кнопки **Аа (Надпись)**) и перевести указатель мыши в область данных окна **Конструктор** - указатель изменит свое изображение со стрелки на символ [+^A].

Переместить указатель в первую слева клетку первой строки и нажать ЛК мыши - выводится тоненькая полоска текстового поля. Начиная от отмеченной позиции, напечатать в первой строке заголовков формы **База данных системы контроля безопасности движения СБД ФТС гр. Т-21**) и фа-

милию пользователя (например, **Петренко**). Для того чтобы разделить надпись на две строки, следует расположить текстовый курсор перед буквами **ФТС**, нажатием **Backspace** удалить два пробела и нажать **[Shift] + [Enter]**.

Нажать **[Enter]** - на линиях поля выводятся маркеры, фон поля изменится на серый цвет, а под панелью главных команд среды Access выводится панель инструментов **Форматирования**.

Пометить для надписи размер шрифта **14**, а для выравнивания - **По центру**. Для исправления ширины поля надписи выполнить команду **Формат→Размер→По размеру данных**.

Пользователю рекомендуется самостоятельно указать цвета фона и шрифта с помощью инструментов **Цвет заливки** и **Цвет шрифта** панели **Форматирования**. Пользуясь мышью, растянуть надпись на всю ширину поля.

3) Для создания кнопок управления панели инструментов пометить элемент **Кнопка**, перевести указатель мыши в окно **Конструктор** и нажать ЛК мыши в области данных на месте расположения кнопки; это может быть любое место в поле, впоследствии кнопку можно будет переместить. В ответ выводится диалоговое окно **Создание кнопок**.

В области **Категории:** пометить Работа с формой, а в области **Действия:** пометить **Открыть форму** и нажать кнопку **Далее** - открывается окно **Создание кнопок**.

4) Пометить форму **Информация 1**, которая будет открываться этой кнопкой, и нажать кнопку **Далее**.

В очередном окне нажать кнопку **Далее**. В следующем окне пометить переключатель **Текст**; удалить текст **Открыть форму**, ввести название формы **Информация 1** и нажать кнопку **Далее**.

5) В последнем окне назначается название кнопки. Для этого следует удалить название, которое по умолчанию выводит программа, а напечатать название, соответствующее форме, для открытия которой создается кнопка, а именно - **Информация 1**.

Завершается проектирование нажатием кнопки **Готово** - программа возвращается в окно **Конструктор**, в котором отображена созданная кнопка (рис. 9). Размеры кнопки и ее расположение изменяют с помощью перетягивания маркеров мышью, ориентируясь на линии сетки.

6) Аналогично пунктам 1) – 5) данного раздела создать кнопки для всех форм СБД и равномерно расположить их в пределах поля сетки (рис. 3).

По окончании создания кнопок вернуться в окно **Кнопочная форма** с помощью кнопки **Закрыть** в заголовке окна. По умолчанию у первой из созданных кнопок название будет окружено точечной рамкой, что свидетельствует об активности кнопки. При нажатии другой кнопки обозначение активности переносится на название второй кнопки.

7) Для того, чтобы форма с кнопками управления автоматически выводилась на экран при обращении к базе данных СБД, следует во вкладке **Меню** выполнить команду **Сервис→ Параметры запуска**.

6. Создание запросов

6.1. Запрос является гибким и удобным способом доступа к данным. Для одной и той же таблицы можно создать множество разных запросов, каждый из которых сможет выбирать из таблицы необходимую часть информации. В результате работы запроса из общей исходной базы формируется результирующая таблица. Важным свойством запросов является то, что при создании результирующей таблицы можно не только выбирать информацию из таблиц, но и обрабатывать ее. При работе запроса данные могут сортироваться, фильтроваться, объединяться, разделяться, изменяться, вычисляться; при этом базовые таблицы остаются неприкосновенными. Наиболее используемыми являются запросы на выбор информации. Если в окне базы данных пометить объект **Запросы**, то в области справа выводятся ярлыки и названия двух способов создания запросов.

Применение режима **Мастера** пользователи исследовали при проектировании форм. Для конструирования запросов в среде Access пользователям необходимо использовать и исследовать режим **Конструктора**.

6.2. Все запросы в соответствии с вариантами заданий содержат три обязательных поля для идентификации водителей (табл. 8, поля 1, 2, 3), а четвертое поле - для дополнительного отбора информации.

Обратить особое внимание на то, что в вариантах запросов пользователей повторяются поля с номерами 1, 2 и 3. Это объясняется тем, что все запросы должны быть **персонифицированы**; четвертое поле в каждом запросе у каждого пользователя отличается.

С этой целью нужно создать в режиме **Конструктор** три полных (без условий) запроса. **Безусловные** запросы проектируются следующими действиями.

1) Пометить объект **Запросы** в окне базы данных и исполнить команду **Создание запроса в режиме конструктора**. На экран выводится окно конструктора, которое имеет название **Запрос1: запрос на выборку**. Вместе с ним на экран выводится окно **Добавление таблицы**. Следует обратить внимание на наличие в последнем окне трех страниц: **Таблицы, Запросы, Запросы и таблицы**. Они свидетельствуют о том, что запросы создаются не только по таблицам, но и по ранее созданным запросам.

2) В окне **Добавление таблицы** пометить таблицы, по которым создается запрос, и нажать кнопку **Добавить** - окно закрывается, а помеченные копии таблиц, на которых основан запрос, вместе со связями выводятся в верхней части бланка запроса. Строки нижней части бланка определяют структуру запросов, то есть структуру **результатирующей** таблицы запроса.

Каждое поле определенного запроса (по варианту пользователя) будет занимать один столбец бланка "запроса на выборку". Нажать ЛК мыши на названии поля соответствующей таблицы в верхней части бланка запроса и, удерживая ЛК мыши нажатой, передвинуть курсор вместе с символическим отображением названия в строку **Поле:** в очередном свободном столбце. Строка **Имя таблицы:** заполняется автоматически при перетаскивании названия поля.

Перетягивать названия полей в том порядке, какой указан в варианте пользователя.

Альтернативой перетаскиванию названий полей является двойное нажатие ЛК мыши на названии поля, используемого в запросе.

3) Нажать ЛК мыши в строке **Сортировка** на кнопке с треугольником - раскрывается список видов сортировки. Способ сортировки определяется пользователем самостоятельно и способствует сортировке данных в результирующей таблице запроса.

4) Завершить создание запроса нажатием кнопки **Заккрыть** - на экран выводится окно для ввода **названия** запроса, которое должно отвечать названию первичной таблицы (по которой образован запрос) и иметь очередной порядковый номер (например, **Информация 2**).

5) Аналогично пунктам 1) - 4) создать остальные запросы в соответствии с вариантом задания.

6.3. Повторить создание запросов аналогично пунктам 1) – 4) раздела 6.2. и присвоить им названия с цифрой **3** (например, **Информация 3**). Дублированные запросы предназначены для исследования влияния условий выбора. С этой целью в режиме **Конструктор** в поля запросов (кроме полей **1, 2, 3**) в строке **Условие выбора** вводятся условия выбора, которые пользователь определяет самостоятельно, используя сведения таблиц 11, 12.

Обращение к запросу и его открытие выполняется в главном окне базы данных нажатием ЛК мыши на названии запроса.

Таблица 11. **Примеры шаблонов**

Используемый символ	Образец	Результат
Символ вопроса (?) в качестве шаблона для любого символа	За?ор	Забор Затор
Звездочка (*) является шаблоном любой группы исходных символов	*й	128й Последний
Знак фунта (#) в качестве шаблона для любой цифры.	# - и	5 - и, 8 - и

Таблица 12. Примеры условных выражений

Условное выражение	Содержание выводимой записи
Киев	"Киев"
Not Киев	Без "Киев"
In (Киев, Ялта, Винница)	"Киев", "Ялта" или "Винница"
<M	От буквы "А" до буквы "Л"
>=M	От буквы "М" до буквы "Я"
100	"100"
<=20	Меньше или равно 20
Date()	Имеют значение текущей даты
>=01.01.94	После 01.01.94
Between 01.01.93 AND 31.12.93	В течение 1993 года
.02.	Месяц "Февраль"
Null	Содержат в поле пустое значение
Like "P*"	Начинаются с буквы "P"

7. Создание отчетов

7.1. Отчеты предназначены для вывода информации на печать и поэтому для расчета расположения данных на печатной странице необходимо сообщить программе тип используемого принтера. Отчеты состоят из разделов, а разделы могут содержать элементы управления.

7.2. В главном окне БД указать объект **Отчеты**, а затем нажать кнопку **Создать**. В новом окне **Новый отчет** указать **Автоотчет: в столбец**. В нижних полях указать названия таблиц или отчетов, по которым создается отчет; завершить создание отчета нажатием кнопки **ОК**.

7.3. В окне вопроса о сохранности отчета нажать кнопку **Да** - на экран выводится окно для ввода названия отчета, которое должно соответствовать названию запроса (на основе которого создан отчет) и иметь очередной порядковый номер (например, **Информация 4**).

7.4. Структуру отчета удобнее исследовать и совершенствовать после создания **Автоотчета** и его открытия в режиме **Конструктора**. Структура отчета состоит из пяти следующих разделов:

- 1) **заголовок** - нужен для печати общего заголовка отчета;
- 2) **верхний колонтитул** - используется для печати подзаголовков и номеров страниц;
- 3) **область данных** - содержит два элемента управления, связанных с содержимым полей таблиц базы данных; в эту область выводятся данные из таблиц для печати на принтере.
- 4) **нижний колонтитул** - используется для печати подзаголовков и номеров страниц;
- 5) **примечания** - содержит дополнительные сведения о параметрах отчета.

В **области данных** в **первом** элементе управления выводится текущая дата. Для этого используется встроенная в MS Access функция **Now()**. Она возвращает текущую дату и размещает в поле, а отчет воспроизводит ее при печати.

При обращении ко **второму** элементу управления выводится номер страницы и общее количество страниц. Для их определения используются встроенные функции **Page()** и **Pages()**. Текст, ограниченный кавычками, воспроизводится в соответствии с записью, а оператор **&** (оператор конкатенации) применяется для соединения текста в кавычках и возвращаемых функциями значений.

Разработайте отчеты к таблицам **Информация**, **Учет ДТП** и **Нарушения**. Стиль оформления выберите самостоятельно.

11. Контрольные вопросы

1. В каких объектах хранятся данные БД?
2. Чем отличаются поля и записи таблицы?
3. Какие существуют типы полей таблиц?
4. Содержит ли информация таблица, в которой нет записей?
5. Содержит ли информация таблица, в которой нет полей?
6. В чем заключается особенность поля типа Счетчик?
7. В чем заключается особенность поля типа Мемо?
8. Какое поле можно считать уникальным?
9. Какую БД называют реляционной?
10. Где создают связи между полями таблиц БД?
11. Для чего предназначены запросы?
12. Как называется запрос, предназначенный для выбора данных из таблиц БД?
13. Как называется область, заполнение которой создает структуру запроса?
14. Как называется поле результатов расчета по данным других полей?
15. Для выполнения итоговых вычислений записи в таблицах должны быть сгруппированными по каким-либо признакам. Как ввести в бланк запроса эту строку?
16. Как с помощью запроса выполнить сортировку записей в таблице по любому из полей?
17. Какие существуют два типа сортировки?
18. Для чего предназначены формы?
19. Какие существуют методы создания форм?
20. Какие существуют элементы управления форм?
21. Для чего предназначены отчеты?
22. Какое устройство осуществляет вывод данных отчета?

Список литературы

1. Симбирская Л. М., Симбирский Г. Д., Левтеров А. И. Информационные системы и технологии: Учебно-методическое пособие (лабораторный практикум). – Харьков: ХНАДУ, 2015. – 91 с.
2. Симбирская Л. М., Симбирский Г. Д., Левтеров А. И. Информационные системы и технологии: Учебно-методическое пособие для самостоятельного изучения. – Харьков: ХНАДУ, 2015. – 101 с.
3. Прокофьев В. А. Информационные технологии на транспорте: Учебное пособие. - СПб.: ГМА им. адм. С. О. Макарова, 2006.
4. И. А. Лапкина, С. П. Онищенко Информационные системы на транспорте: Учебное пособие: - Одесса: Феникс, 2006.
5. Мак-Федрис Пол. Формы, отчеты и запросы Microsoft Access 2003.: Пер. с англ. - М.: "Вильямс", 2005.
6. Система управления базами данных Microsoft Access для самостоятельной учебы: Учебное пособие / Составители: Н. В. Баловсяк, И. А. Григоришин, Л. В. Кулибаба. - К.: Дакор, КНТ, 2006.
7. Николаев А. Б. и др. Автоматизированная система обработки информации и управления на автомобильном транспорте: Учебник / - М.: "Академия", 2003.
8. Ярмуш О. В., Редько Г. Г. Информатика и компьютерная техника: Учебное пособие. - К.: Высшее образование. 2006.