**9. Опишите процесс задания ориентации фигур на странице, ориентация текста фигуры, позиционирования текста фигуры в MS Visio.**

Задание ориентации фигур на странице

При переносе фигур на страницу документа Visio они обычно уже имеют нужную ориентацию. Однако иногда нужно, чтобы фигуры отображались под другим углом. Visio предлагает несколько способов выполнить эту задачу.

Запустим программу Visio или, если она уже запущена, щелкнем на вкладке Файл (File), и выберем страницу Создать (New). В разделе Категории шаблонов (Template Categories) щелкнем на категории Общие (General), затем дважды на миниатюре Блок диаграмма (Block Diagram).

1. На вкладке Вид (View) в группе Масштаб (Zoom) щелкнем на кнопке По ширине страницы (Page Width), а затем расположим страницу так, чтобы было видно ее верхний край.
2. Перетащим фигуру Поле (Box) из набора элементов Блоки (Blocks) и сбросим ее так, чтобы ее центр находился приблизительно на 200 мм вверх от нижнего края и на 75 мм вправо от левого края страницы.

В версии программы с американской системой мер фигуры набора элементов Блок диаграмма (Block Diagramm) содержат слово текст (text), в то время как те же фигуры в версии с метрической системой мер текст не содержат. Помимо обычных маркеров выделения, окружающих поле, сверху по центру над полем имеется маркер поворота. При наведении указателя мыши на маркер поворота, указатель меняется со стрелки на круговую стрелку. Точка в центре прямоугольника показывает геометрический центр фигуры, то есть точку, вокруг которой фигура будет вращаться при перетаскивании маркера поворота.

1. Захватим маркер поворота и повернем фигуру на 90 гра­дусов влево (против часовой стрелки). Поворачиваются и фигура, и текст.
2. Перетащим маркер поворота против часовой стрелки еще на 90 градусов. Текст также перевернулся.
3. Перед тем как продолжить, вернем полю исходную ориентацию. Фигуру свободно можно вращать по дуге в 360 градусов.
4. Перетащим фигуру Круг (Circle) на страницу так, чтобы ее центр отстоял приблизительно на 75 мм вправо от центра поля (для выравнивания с полем используем динамическую сетку).
5. На вкладке Главная (Home) в группе Сервис (Tools) выберем инструмент Соединительная линия (Connector).
6. Нарисуем соединительную линию с правой стороны поля до левой стороны круга.
7. На вкладке Главная (Home) в группе Сервис (Tools) выберем инструмент Указатель (Pointer Tool).
8. Щелкнем на новой соединительной линии и введем Примерный текст. Затем нажмем клавишу Esc.
9. Нарисуем ограничивающую рамку вокруг всех трех фигур, чтобы выделить их, а затем щелкнем правой кнопкой мыши на любой из фигур.
10. На мини-панели инструментов зададим размер шрифта в 14 пт (14 pt), чтобы улучшить читаемость текста для последующих шагов.
11. Нарисуем ограничивающую рамку вокруг всех трех фигур, чтобы выделить их, а затем повернем весь выбранный фрагмент на 90 градусов по часовой стрелке.
12. Повернем выделенный фрагмент еще на 90 градусов по часовой стрелке, инвертировав тем самым фигуры.

Ориентация текста фигуры

Текст фигуры не всегда поворачивается при повороте фигуры, в которой он содержится. Это зависит от того, как эта фигура была создана.

1. Перетащим стрелку Поворотная односторонняя (2D single) из набора элементов Блоки (Blocks) на страницу документа и разместим ее под фигурой поля.
2. Не снимая выделение со стрелки, введем текст фигуры.
3. В окне Фигуры (Shapes) выберем набор элементов Рельефные блоки (Blocks Raised), перетащим фигуру Стрелка вправо (Right arrow) на страницу и поместим ее под стрелкой.
4. Не снимая выделение со стрелки, введем текст фигуры.
5. Нарисуем ограничивающую рамку вокруг обеих стрелок, чтобы выделить их, а затем изменим размер шрифта на 14 пт (14 pt).
6. Выделим верхнюю стрелку и с помощью маркера поворота повернем ее на 180 градусов.
7. Выделим нижнюю стрелку и с помощью маркера поворота повернем ее на 180 градусов.

Инструмент Блок текста (Text Block) позволяет изменять ориентацию текста практически в любой фигуре. Однако следует знать, что создатель фигуры может заблокировать текст в ней, запретив тем самым изменять или переориентировать его. Если щелкнуть инструментом Блок текста (Text Block) на фигуре с заблокированным текстом, маркеры будут иметь серый цвет.

1. На вкладке Главная (Home) в группе Сервис (Tools) щелкнем на кнопке Блок текста (Text Block).
2. Щелкнем один раз на верхней стрелке и обратите внимание на то, что маркеры выделения отображаются обычным образом.
3. Перетащим маркер поворота по часовой стрелке. Поскольку мы используем инструмент Блок текста (Text Block), а не Указатель (Pointer Tool), поворачивается только текст. Остановимся, когда повернем текст на 180 градусов.

Позиционирование текста фигуры

Текст на фигуре Visio находится в текстовом блоке. Изменять положение текста в рамках блока позволяют несколько команд на ленте. Также можно перемещать весь блок текста. Это делается с помощью инструмента Блок текста (Text Block).

1. Щелкнем один раз на прямоугольнике, который содержит текст.
2. На вкладке Главная (Home) в группе Абзац (Paragraph) щелкнем на любой кнопке, чтобы понаблюдать за эффектом.
3. На вкладке Главная (Home) в группе Сервис (Tools) щелкнем на кнопке Блок текста (Text Block), а затем щелкнем один раз на прямоугольнике, который содержит текст.
4. Щелкнем где-нибудь на границе текстового блока (синие пунктирные линии) и перетащим блок текста вверх и влево.
5. Когда активен инструмент Блок текста (Text Block), текстовый блок перемещается независимо от своей фигуры. Если использовать инструмент Указатель (Pointer Tool) и перемещать фигуру, то текстовый блок будет также перемещаться с сохранением своего положения относительно фигуры.
6. С помощью инструмента Блок текста (Text Block) также можно изменять размер текстового блока. Для этого нужно перетащить один из маркеров изменения размера. Хотя перемещение текстового блока на данной фигуре не имеет большого смысла, бывают ситуации, когда умение перемещать и вращать текст может пригодиться.
7. В окне Фигуры (Shapes) щелкнем на Дополнительные фигуры (More Shapes), перейдем в подменю Сеть (Network) и выберем набор Компьютеры и мониторы (Computers and Monitors). Открывается набор элементов Компьютеры и мониторы (Computers And Monitors).
8. Перетащим фигуру ЖК-монитор (LCD Monitor) на страницу, сбросим ее под фигурой круга и введем текст для фигуры.
9. На вкладке Главная (Home) в группе Сервис (Tools) щелкнем на кнопке Блок текста (Text Block). Переместим и повернем текстовый блок так, чтобы текст фигуры отображался на экране ЖК-монитора приблизительно под тем же углом, что и сам экран.
10. Перемещение и поворот текста в фигуре ЖК-монитор - это отличный пример ситуации, когда в программе Visio полезно использовать функцию масштабирования. Перед тем как пытаться переместить и повернуть блок текста, увеличьте масштаб, чтобы увеличить фигуру, и вы сможете выполнить эту задачу легче и точнее.

**46. Перечислите и опишите форматы данных.**

Существуют следующие форматы данных в приложении Microsoft Office Excel:

* общий,
* числовой,
* денежный,
* финансовый,
* дата,
* время,
* процентный,
* дробный,
* экспоненциальный,
* текстовый,
* дополнительный,

Общий

По умолчанию введенным данным присваивается так называемый общий формат,который подходит как для числовой, так и для текстовой информации. Однако нередко требуется работать с данными определенного типа, например с датами, или сделать так, чтобы все числа с дробной частью округлялись до целого значения. В таком случае нужно задать конкретные параметры.

Назначение формата данных осуществляется с помощью группы *Число* на вкладке *Главная* ленты инструментов. Сюда вынесены инструменты, которые используются в данной группе чаще всего. Для доступа к окну *Формат ячеек*, содержащему полный список форматов и их детальную настройку, нужно нажать соответствующую кнопку на ленте в разделе число ().

Числовой

Окно *Формат ячеек* содержит несколько вкладок. На вкладке *Число* (рис.1) в левой части окна находится область *Числовые форматы* со списком всех доступных форматов. Для выбора одного из них нужно щелкнуть на его названии левой кнопкой мыши и нажать ОК.

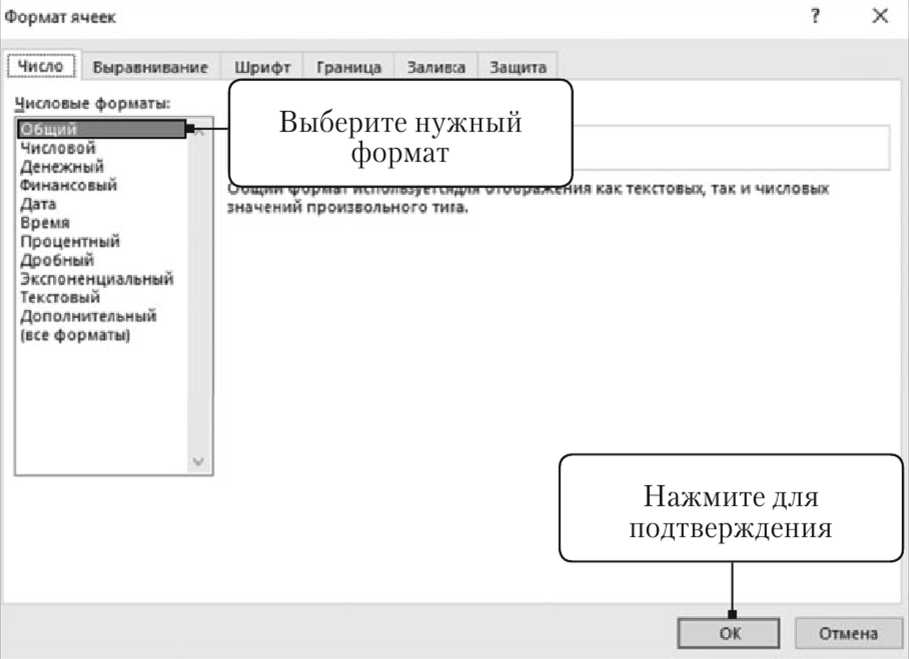


Рис. 1.Вкладка Число в окне Формат ячеек.

Для того чтобы назначить числовой формат, щелкнем левой кнопкой мыши на соответствующем названии в списке *Числовые форматы* в окне *Формат ячеек*. В правой части окна появятся инструменты настройки (рис. 2).

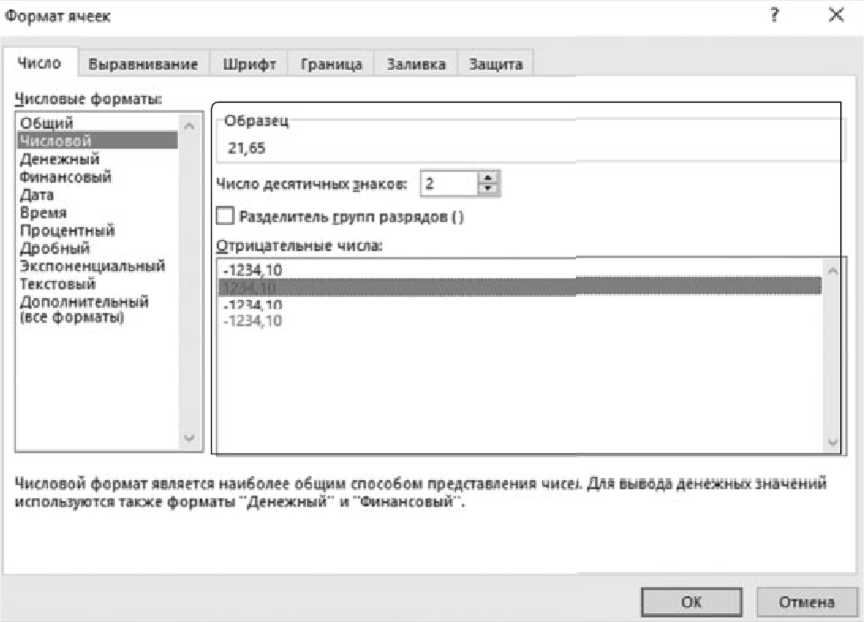


Рис.2.Инструменты настройки числового формата данных.

В поле *Число десятичных знаков* установим столько знаков после запятой, сколько необходимо (два знака (сотые доли), для отображения тысячных долей введите цифру 3 (рис. 3)). Нажмем ОК, чтобы сохранить изменения.

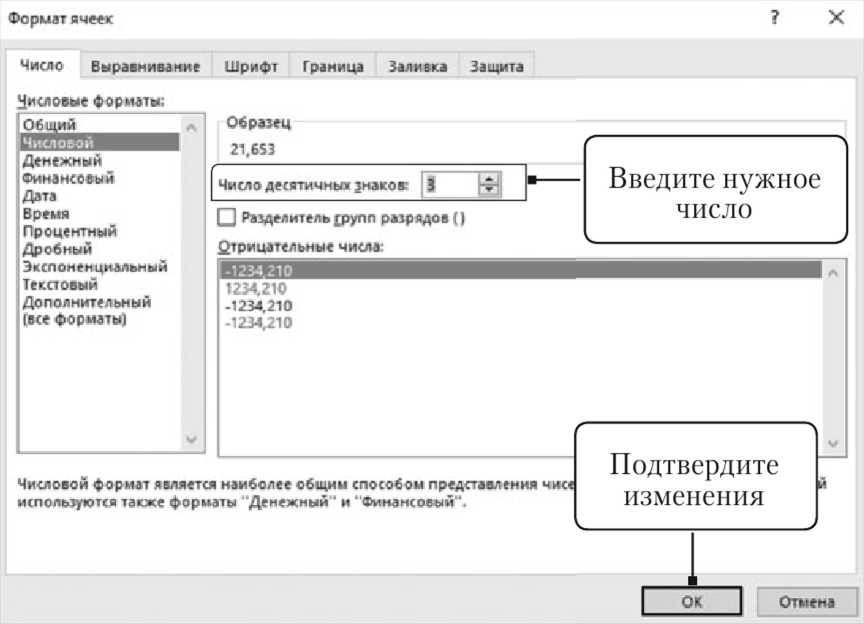


Рис. 3.Настройка отображения третьего знака после запятой.

После включения флажка *Разделитель групп разрядов* большие числа разделяются на триады (группы по три цифры), что улучшает удобство восприятия и работу с ними.

Денежный и финансовый

Чтобы задать приемлемый для цены формат, в списке окна *Формат ячеек* укажем *Денежный*. Затем в раскрывающемся списке *Обозначение* выберем желаемую валюту и нажмем ОК.

Значения в ячейках примут вид, как показано на рисунке 3.

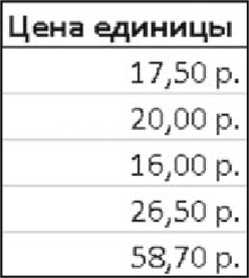


Рис. 3.Значения цен в денежном формате.

Дата и время

Форматы Дата и Время используются, когда необходимо, чтобы Excel воспринимал значение в ячейке именно как дату или время, а не как текст или обычное число. Такое представление значений позволяет совершать с ними большое количество специальных операций. В Excel существует несколько функций, предназначенных непосредственно для работы с датой и временем.

Вам наверняка знакома формулировка «дата в формате ДД.ММ.ГГГГ».

Как известно, у даты есть три составляющие — день (число месяца), месяц, год. В формулировках, подобных рассматриваемой, каждая из них обозначается первой буквой: Д — день, М — месяц, Г — год. Такое обозначение сродни аббревиатуре ФИО (Ф — фамилия, И — имя, О — отчество).

Количество одинаковых букв в нашей формулировке определяет количество символов, которое будет содержать каждая составляющая даты. Например, «ММ» говорит о том, что месяц указывается двумя символами: январь — 01, февраль — 02 и т.д. Если бы в определении формата стояла одна буква «М», то месяцы с января по сентябрь обозначались бы одной цифрой — 1, 2,..., 9.

Соответственно, обозначение «ДД» говорит о том, что день недели указывается двумя символами (01, 02, 07). «ГГГГ» — год записывается полностью (2015, 2016).

Реже в повседневной жизни встречается формат записи, в котором день и месяц записываются текстом. При этом в таблицах Excel он используется довольно часто. Например, самый длинный формат записи даты — «ДДДД.ММММ.ГГГГ». Четыре «Г» подразумевают запись года в виде четырехзначного числа. С днями и месяцами все несколько иначе. ММ ММ подразумевает запись месяца полным словом — Январь, Февраль и т. д. Соответственно, ДДДД говорит о длинной записи дня — понедельник, вторник.

Иногда можно встретить форматы «ДДД» и «МММ». Они подразумевают сокращенную словесную запись дней и месяцев — Пн, Вт, Ср, Янв, Фев, Мар и т. д.

Excel позволяет создавать пользовательские форматы данных, в том числе даты и времени. В записи не обязательно использовать сразу месяц, год и день. Нередко можно встретить формат вида ДД.ММ (01.01). Он очень удобен, если известно, о каком именно годе идет речь.

Excel позволяет задавать отличные от символа точки разделители частей даты. Например, используется запись вида ДД ММ ГГГГ.

Изложенную информацию можно применить и при записи времени, так как она строится по тому же принципу. Понимание формата позволяет максимально эффективно использовать способы записи. Ниже приведена таблица соотношений некоторых форматов с записью даты 1 января 2016 года.

|  |  |
| --- | --- |
| Формат | Пример записи даты |
| Д.М.Г | 1.1.16 |
| ДД.ММ.ГГГГ | 01.01.2016 |
| ДДДД | Среда |
| ММММ | Январь |
| ДД.ММ | 01.01 |
| ДДД, МММ.ГГ | Ср, Янв.16 |
| ММ.ГГГГ | 01.2016 |
| Д | 1 |

Процентный

Данный формат используется для работы с процентами. В этом случае знак процента вводится автоматически. Кроме того, единица в ячейках такого формата рассматривается как 100 %. Таким образом, если вы введете в такую ячейку число 0,5, оно заменится на 50 %, что при работе с процентами бывает весьма удобно.

Текстовый

Видя такой формат, Excel не обрабатывает данные, а интерпретирует их именно как текст. По умолчанию, если строка начинается со знака равенства, Excel воспринимает ее как формулу. Так, в общем формате после ввода текста = 2 + 2 в ячейке отобразится результат арифметического действия, то есть 4. Чтобы этого не произошло, ячейке следует присвоить текстовый формат.

Сначала не­обходимо присвоить формат, а только затем вво­дить текст. Иначе вы переведете в текстовый формат не само выражение, а полученный чис­ловой результат.

**84.** **Охарактеризуйте связи информационных объектов.**

Связи информационных объектов могут быть разного типа:

* одно-однозначные (1 : 1);
* одно-многозначные (1 : М);
* много-многозначные (М : N).

*Одно-однозначные связи* имеют место, когда каждому экземпляру первого объекта (А) соответствует только один экземпляр второго объекта (В) и наоборот, каждому экземпляру второго объекта (В) соответствует только один экземпляр первого объекта (А). Следует заметить, что такие объекты легко могут быть объединены в один, структура которого образуется объединением реквизитов обоих исходных объектов, а ключевым реквизитом может быть выбран любой из альтернативных ключей, т.е. ключей исходных объектов. Графическое изображение представлено на рисунке 1. Примерами одно-однозначных связей являются: группа – староста, фирма – расчетный счет в банке и т.п.

1:1

Объект B

Объект A

Рисунок 1. Графическое изображение одно-однозначных отношений объектов.

*Одно-многозначные связи (1:М)* – это такие связи, когда экземпляру одного объекта (А) может соответствовать несколько экземпляров другого объекта (В), а каждому экземпляра второго объекта (В) может соответствовать только один экземпляр первого объекта (А). Графическое изображение представлено на рисунке 2.

1:М

Объект A

Объект B

Подчиненный информационный объект

Главный информационный объект

Рисунок 2. Графическое изображение одно-многозначных отношений объектов.

В такой связи объект А является главным объектом, а объект В – подчиненным, т.е. имеет место иерархическая подчиненность объекта В объекту А. Примером одно-многозначных связей являются пары объектов: подразделения – сотрудники, кафедра – преподаватель, группа - студент и т.п.

*Много – многозначные связи (M:N)* – это когда, каждому экземпляру одного объекта (А) могут соответствовать несколько экземпляров второго объекта (В) и наоборот, каждому экземпляру второго объекта (В) может соответствовать тоже несколько экземпляров первого объекта (А). Графическое изображение представлено на рисунке 3.

М:N

Объект B

Объект A

Рисунок 3. Графическое изображение много-многозначных отношений объектов.

Много-многозначные связи не могут непосредственно реализовываться в реляционной базе данных. Поэтому в том случае, когда будут выявлены такие связи, может возникнуть необходимость их преобразования путем введения дополнительного объекта "связка". Исходные объекты будут связаны с этим объектом одно-многозначными связями. Таким образом, объект-связка является подчиненным в одно-многозначных связях по отношению к каждому из исходных объектов.

Объект-связка должен иметь идентификатор, образованный из идентификаторов исходных объектов Ka и Kb (рисунок 4).

М:N

Объект A

Объект B

Ключ Ka

Ключ Kb

1:N

1:М

Объект-Связка

Ключ Ka + Kb

Рисунок 4. Преобразование связи типа М : N с помощью объекта-связки

При рассмотренном подходе к выделению информационных объектов объект-связка, как правило, выявляется в результате рассмотренного ранее анализа функциональных зависимостей реквизитов. Много-многозначные связи в этом случае не требуют специальной реализации, т. к. осуществляются через объект, выполняющий роль объекта-связки. Примером много-многозначных связей является пара вида: поставщики-товары, если один поставщик поставляет разные наименования товаров, а товар одного наименования может поставляться несколькими поставщиками.

Определение связей между информационными объектами

Рассмотрим определение связей между информационными объектами и тип отношений, которыми они характеризуются, для предметной области Учебный процесс.

Связи между объектами ГРУППА  СТУДЕНТ характеризуются одно-многозначными отношениями (1:М), поскольку одна группа включает много студентов, а один студент входит только в одну группу. Связь между ними осуществляется по номеру группы, который является уникальным идентификатором главного объекта ГРУППА входит в составной идентификатор объекта СТУДЕНТ (см.табл.1)

Аналогично устанавливается связь между объектами КАФЕДРА ПРЕПОДАВАТЕЛЬ, которые также находятся в одно-многозначных отношениях. Связь между ними осуществляются по уникальному ключу главного объекта КАФЕДРА – коду кафедры, который в подчиненном объекте ПРЕПОДАВАТЕЛЬ является описательным.

Таблица 1. Объекты справочной информации о студентах, группах и предметах

|  |  |  |  |
| --- | --- | --- | --- |
| Информационный объект | Название реквизитов | Обозначение реквизита | Признак ключа |
| ГРУППА | Номер группы Количество студентов Средний балл в группе | НГ КОЛ ПБАЛ | Уникальный простой |
| СТУДЕНТ | Номер группы Номер студентов  Фамилия И.О. Год рождения Адрес Балл при поступлении | НГ НС ФИО ГОДР АДРЕС ПБАЛЛ | Уникальный составной |
| ПРЕДМЕТ | Код предмета Название предмета Всего часов Часов лекций Часов практики Число семестров Программа курса (большой текст) | КП НП ВЧ ЧАСЫ ЛЕК ПР ЧС ПРОГ | Уникальный простой |

Таблица 2. Группировка реквизитов по информационным объектам документа Список преподавателей кафедры

|  |  |  |  |
| --- | --- | --- | --- |
| Реквизиты объекта | Признак ключа | Имя информационного объекта | Семантика объекта |
| ТАБН ККАФ ФИО СТ ЗВ | П,У | ПРЕПОДАВАТЕЛЬ | Сведения о преподавателях кафедр |
| ККАФ НКАФ ТЕЛ ЗАВ | П,У | КАФЕДРА | Сведения о кафедрах |

В таблице приняты обозначения для ключа: П – простой, У – уникальный.

В каждой группе в течение семестра проводятся занятия по разным предметам (ОБЪЕКТ ИЗУЧЕНИЕ). С другой стороны, каждое занятие определенно для каждой группы. Поэтому имеет место связь типа один-ко-многим между объектами ПРЕДМЕТ  ИЗУЧЕНИЕ.

По каждому предмету проводится множество занятий в различных группах разными преподавателями. С другой стороны, каждое занятие проводится по конкретному предмету, что определяет отношения типа один-ко-многим между объектами ПРЕДМЕТ  ИЗУЧЕНИЕ.

Аналогично определяются отношения типа один-ко-многим между объектами ПРЕПОДАВАТЕЛЬ-ИЗУЧЕНИЕ.

Объект ИЗУЧЕНИЕ фактически играет роль объекта связки в много-многозначных отношениях объектов.

Объект УСПЕВАЕМОСТЬ содержит данные об успеваемости (оценку) конкретного студента по конкретному занятию. Поэтому он является связанным с объектом СТУДЕНТ и объектом ИЗУЧЕНИЕ. Один студент имеет оценки по нескольким занятиям, но каждая оценка всегда относится к одному конкретному студенту. Это означает, объект УСПЕВАЕМОСТЬ является подчиненным и находится в одно-многозначных отношениях с объектом СТУДЕНТ. Объект УСПЕВАЕМОСТЬ, а также является подчиненным и находится в одно – многозначных отношениях с объектом ИЗУЧЕНИЕ. Объект УСПЕВАЕМОСТЬ выполняет роль объект-связки много-многозначных отношениях объектов СТУДЕНТ и ИЗУЧЕНИЕ. Много-многозначные отношения между этими объектами определяются тем, что одному студенту соответствует много занятий, отображаемых объектом ИЗУЧЕНИЕ, а одно занятие проводится со многими студентами.

В табл.3 перечислены все одно-многозначные связи между объектами, указаны ключи, по которым должны устанавливаться связи, и определены главные и подчиненные информационные объекты в этих связях.

Таблица 3. Связи информационных объектов

|  |  |  |  |
| --- | --- | --- | --- |
| Главный объект | Подчиненный объект | Тип связи | Ключ связи |
| ГРУППА КАФЕДРА ГРУППА ПРЕДМЕТ ПРЕПОДАВАТЕЛЬ СТУДЕНТ ИЗУЧЕНИЕ | СТУДЕНТ ПРЕПОДАВАТЕЛЬ ИЗУЧЕНИЕ ИЗУЧЕНИЕ ИЗУЧЕНИЕ УСПЕВАЕМОСТЬ УСПЕВАЕМОСТЬ | 1:М 1:М 1:М 1:М 1:М 1:М 1:М | НГ К КАФ НГ КП ТАБН НГ+НС НГ+КП+ТАБН+ВИДЗ |

**94. Опишите операторы используемые при создании запросов на**

**выборку с параметрами.**

Запросы являются основным инструментом выборки, обновления и обработки данных в таблицах базы данных.

Access в соответствии с концепцией реляционных баз данных для выполнения запросов использует язык структурированных запросов SQL (Structured Query Language). С помощью инструкций языка SQL в Access реализуется любой запрос.

В то же время Access позволяет создавать запросы, не прибегая к записи инструкций языка SQL. Простейшие запросы могут быть созданы с помощью мастера, практически любой запрос можно создать в режиме графического конструктора.

При создании запроса этими средствами Access сам автоматически создает эквивалентную инструкцию SQL, которую можно увидеть, переключившись в режим SQL. Конструктор позволяет создавать запросы простым и удобным способом, а просмотр этих запросов в режиме SQL — понять и освоить синтаксис основных инструкций языка SQL, реализованного в Access.

Основным видом запроса является *запрос на выборку*. Результатом выполнения этого запроса становится новая таблица, которая существует до закрытия запроса. Структура такой таблицы определяется выбранными из одной или нескольких взаимосвязанных таблиц полями. Записи формируются путем объединения записей таблиц, на которых построен запрос. Способ объединения записей таблиц указывается при определении их связи в схеме данных или при создании запроса. Условия отбора, сформулированные в запросе, позволяют фильтровать записи, составляющие результат объединения таблиц. Запрос на выборку позволяет сформировать пользовательское представление о данных, не обязательно отвечающее требованиям нормализации.

С помощью запроса можно выполнить следующие виды обработки данных:

* включить в таблицу запроса выбранные пользователем поля таблицы;
* выбрать записи, удовлетворяющие условиям отбора;
* сформировать на основе объединения записей взаимосвязанных таблиц новую виртуальную таблицу;
* произвести вычисления в каждой из полученных записей;
* сгруппировать записи, которые имеют одинаковые значения в одном или нескольких полях, одновременно выполнить над другими полями группы статистические функции и в результат включить одну запись для каждой группы;
* создать новую таблицу базы данных, используя данные из существующих таблиц;
* произвести обновление полей в выбранном подмножестве записей;
* удалить выбранное подмножество записей из таблицы базы данных;
* добавить выбранное подмножество записей в другую таблицу.

Способы создания запросов:

1. Конструктор;

2. Мастер;

3. Режим SQL.

В режиме конструктора для создания запросов нужно указать следующие параметры:

|  |  |
| --- | --- |
| Поле | Имя поля таблицы, запроса, данные из которого будут располагаться в этом столбце. |
| Таблица | Имя таблицы или запроса, которому принадлежит дан­ное поле. |
| Сортировка | Способ вывода записей в данном столбце. Возможен вывод: по алфавиту (по возрастанию), по убыванию или отсутствует. |
| Вывод на экран | Определяет, будут ли значения этого поля выводиться на экран при выполнении запроса. Тип - логическое поле (имеет вид переключателя). |
| Условие отбора | Здесь в нескольких строках задаются условия отбора (ограничения на вывод информации) в данном поле. |
| Групповая операция | Используется в перекрёстных и итоговых запросах, позволяет обработать значения данного поля при помощи определенной функции. |

При задании условий отбора необходимо соблюдать правила:

1. Перед вводимым значением текстового поля должен стоять знак =.

2. Вводимая строка символов должна заключаться в кавычки.

В строке "Условие отбора" можно задавать несколько ус­ловий, соединенных союзами and и оr.

Выражения - синтаксическая единица, определяющая способ вы­числения некоторого значения. Выражения используются для проверки усло­вий или арифметических вычислений. Выражение должно содержать, по крайней, мере одну операцию и по крайней мере одну константу, идентифика­тор (имя объекта) или функцию.

В некоторых случаях мо­жет потребоваться, например, найти сумму значений по определенному полю. Такое поле можно создать с помощью запроса. Поле, содержание которого является результатом расчета по содержимому других полей, называется *вы­числяемым полем.*

При составлении выражений надо помнить несколько простых правил:

1) выражение всегда имеет логический тип, то есть его значение должно при­нимать либо Да (Истина), либо Нет (Ложь);

2) существуют некоторые требования к синтаксису выражения: а) символьные константы заключаются в кавычки, константы типа дата заключаются в #; б) имя объекта базы данных отделяется от имени поля вос­клицательным знаком, причем имена заключаются в квадратные скобки На­пример, можно прочесть следующее условие [Студенты]![Фамилия]="Андреев" как значение поля "Фамилия" таблицы "Студенты" должно быть "Андреев", проще говоря, показать данные о сту­денте Андрееве;

3) в бланке запроса обычно часть выражения, содержащая имя поля, опуска­ется тогда, когда это выражение задастся в той же колонке. Знак равенства также не пишется, он понимается по умолчанию. Например, условие отбора [Группа]=103 может быть записано в строке "Условия отбора" столбца "Группа" как числовая константа, то есть число 103.

В Access используются следующие операторы: арифметические, присваи­вания и сравнения, логические, идентификации, конкатенации и другие опе­раторы.

В таблице приведен список арифметические, присваивания и сравнения, логические операторов.

|  |  |  |
| --- | --- | --- |
| Знаки арифметических операций | Знаки операций сравнения | Логические операции |
| + сложение | > больше | And логическое И |
| - вычитание | < меньше | Or логическое ИЛИ |
| \* умножение | = равенство (присвоение) | Not логическое отрицание НЕ |
| / деление | >= больше или равно | Xor исключающее ИЛИ |
| ^ возведение в степень | <= меньше или равно |  |
| \ целочисленное деление | <> не равно |  |

В строке "Условие отбора" возможно использование конкатенации значе­ний символьных типов (&), а также следующие операторы, которые упроща­ют создание выражений для выборки записей в запросах;

Is - при использовании вместе с Null определяет, является ли значение Null или Not Null, например. Is Null означает значение поля пусто.

BETWEEN AND - позволяет задать диапазон значений. Например, усло­вие >= 10 and <=20 можно записать так BETWEEN 10 and 20

IN - позволяет задать используемый для сравнения список значений. На­пример, выражение ''хлеб"' оr '"вода" оr "земля" можно записать так IN("хлеб","вода","земля").

LIKE - используется для организации поиска по шаблону. В шаблоне можно использовать следующие символы:

|  |  |  |  |
| --- | --- | --- | --- |
| Символ | Шаблон обозначает | Условие | Access найдёт… |
| \* | Произвольное количество алфавитно-цифровых символов в произвольном месте | LIKE “\*Мар\*” | Ул. 8 Марта, 45-23; ул. Маршака, 180а-1; Мартин |
| ? | Любой одиночный алфавитно-цифровой символ в произвольном месте значения | LIKE“м?р”, “Т??я” | МУР, мир, мор, мер, Толя, Таня, Тоня, То1я, Тося, Тася |
| # | Любая одиночная цифра в произвольном месте значения | LIKE “23#45”, LIKE“52-5#-4#” | 23545, 23145, 23745, 23945 52-55-45? 52-59-41 |
| [ ] | Любой из перечисленных в скобках символ | LIKE “З[оаи]лин” | Зозулин, Зазулин, Зизулин. НО не найдёт Зезулин, Зузулин. |
| ! | Любой один символ, кроме, указанных в скобках | LIKE “ко[!рс]а” | Коза, кола, кома. НО не найдёт кора, коса. |
| - | Любого одиночного символа из указанного интервала | LIKE “ко[к-м]а” | Кола, кома |

Рассмотрим следующий пример. Пусть нам необходимо найти все научные работы, выполненные студентами вуза, по теме "Базы данных". Ситуация осложняется тем, что мы не можем точно знать как сформулированы темы этих работ. С уверенностью можно сказать только, что в названии работы есть словосочетания "база данных", "базы данных" или "базами данных" или "баз данных" или же сокращения от этих слов - "БД". Если задать две маски "\*баз\* данных\*" or "\*БД\*" для поиска в ноле "Название работы" (таблица "Научные работы Студентов"), получим список всех работ, в названии кото­рых входят эти слова независимо от того, стоят ли они в начале, в конце или середине названия работы.

В Access определено около 140 различных функций. Все функции можно сгруппировать по назначению: функции даты и времени; текстовые функции, функции преобразования типов данных; математические и тригонометриче­ские функции; финансовые функции и другие

*Рассмотрим математические и тригонометрические функции:*

Аbs (<число>) Возвращает абсолютное значение указанного числа.

Аtn (<число>) Возвращает арктангенс числа (в радианах).

Соs (<число>) Возвращает косинус угла (в радианах).

Ехр (<число>) Возвращает результат возведения числа е (основание натуральных логарифмов) в указанную степень.

Int (<число>) Fiх (<число>) Возвращают целую часть числа. Обе функции Int() и Fi.xQ отбрасывают дробную часть числа и возвращают целое значение. Различие между функциями JntQ и Fi\-() состоит в том, что для отрицательного значения аргумента число функция Int() возвращает ближайшее отрицательное целое число, ; меньшее либо равное указанному, a Fix() - ближайшее отрицательное целое число, большее либо равное указанному. На- : пример, Int{-8.4)=-9, a Fix(-8.4) =-8.

Log (<число>) Возвращает натуральный логарифм числа.

Rnd [(<число>)] Возвращает случайное число.

Sgn (<число>) Возвращает значение, соответствующее знаку числа. Если число больше нуля, то возвращает 1, если равно нулю, то 0, меньше нуля -1

Sqr (<Hoio>) Возвращает квадратный корень указанного числа.

Tan (<число>) Возвращает тангенс угла (в радианах).

*Рассмотрим некоторые функции по обработке значений типа дата.*

Date ( ) Возвращает текущее системное время и дату

Date$ () Возвращает как дату и как строку.

DateAdd (<интepвaл>;<количество>;<Дата2>) Возвращает дату, полученную добавлением к заданной дате числа интервалов.

DateDiff (<интервал>;<Дата1>;<Дата2>)озвращает интервал времени между указанными датами.

DatePert (<интервал>;<Дата>) Возвращает заданную часть даты.

Month (<Дата >); Определяет номер месяца (1 до 12) в дате.

Day (<Дата>); Определяет день (от 1 до 31) в дате.

Now () Возвращает дату и время системных часов компьютера.

Year (<Дата>); Определяет номер года (от 100 до 9999) в дате.

*Рассмотрим некоторые функции по обработке текстовых данных:*

Аsс (<строка>) Возвращает код символа для первого символа строки.

Сhr (<код Символа>) Возвращает символ, соответствующий указанному коду символа.

Str (<число>) Возвращает строковое представление числа.

Format (выражение[,формат[,…]]); Создание специальных строковых форматов. Возвращает значение, содержащее выражение, отформатированное согласно инструкциям, заданным в описании формата. Действуют все символы формата, рассмотренные в пункте свойство "Формат поля".

InStr ([<позиция>,] <строка1>, <строка2> [, способ сравнения]) Возвращает значение, указывающее позицию первого вхождения одной строки внутри дру­гой строки.

LCase (<cтpoкa>) Возвращает строку, преобразованную к ниж­нему регистру.

Left (<строка>, <количество>) Возвращает строку, содержащую указанное число первых символов строки.

Len (<строка>| <имяПеременной>) Возвращает значение, содержащее число сим­волов в строке или число байт, необходимое для размещения переменной.

LТrim (<строка>) RTrim(<строка>) Trim(<строка>) Возвращают строку, содержащую копию стро­ки, из которой удалены пробелы, находившие­ся в начале строки (LTrim), в конце строки (RTrim) или в начале и конце строки (Trim).

Mid ((<строкa>, <начпоз>[, <количество>]) Возвращает строку, содержащую указанное число символов строки с начальной позиции.

Right (<строкa>, <количество>) Возвращает строку, содержащую указанное; число последних символов строки.

Sрасе (<число>) Возвращает строку, содержащую указанное число пробелов.

String (<число>, <символ>) Возвращает строку, содержащую повторяющуюся строку указанной длины.

UCase (<строка>) Возвращает строку, преобразованную к верхнему регистру.

Vа1 (<строка>) Возвращает числа, содержащиеся в строке, как числовое значение соответствующего типа.

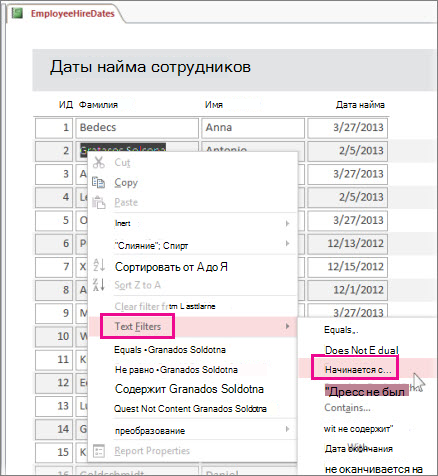
Функция iif (<выражение>, <выражение1>, <выражение2>) возвращает значение <выражение1>, если значение <выражение> равно True, и возвра­щает значение <выражение2>, если значение <выражение> равно False.

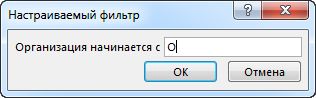
Функция Choose (<индекс>, <вариант1>, [<вариант2>, [<вариант n>]]) возвращает значение, соответствующее заданному положению в списке значе­ний.

**125. Опишите анализ данных отчета средствами фильтрации.**

При просмотре отчета Access на экране можно применить фильтры, чтобы отфильтровать нужные данные. После этого можно распечатать отчет только с нужными данными.

Чтобы отфильтровать данные в отчете, нужно открыть его в представлении отчета (щелкнуть его правой кнопкой мыши в области навигации и выбрать "Представление отчета"). Затем щелкнуть правой кнопкой мыши данные, которые нужно отфильтровать.

Например, в отчете со списком всех сотрудников можно ввести только тех сотрудников, фамилии которых начинаются с "О": щелкнуть правой кнопкой мыши любую фамилию и выбрать "Текстовые > начинается с…"  
  


Введите "О" в поле и нажмите кнопку "ОК".  
  


Access применит фильтр, и теперь можно распечатать отчет только с нужными данными.

Для того, чтобы при необходимости удалить и повторно прибавить фильтр на вкладке "Главная" нужно нажать кнопку “Фильтр”.

Если закрыть отчет без очистки фильтров, Access запомнит их и вы сможете снова нажать кнопку "Перевести фильтр", чтобы повторно применить их. Это возможно даже после закрытия и повторного открытия базы данных. Однако, если нажать кнопку > "Дополнительные > очистить все фильтры", Access полностью очистит фильтры, и в следующий раз их придется настраивать заново.

Если к отчету применено много фильтров, их можно сохранить в качестве запроса. Затем можно использовать запрос в качестве источника данных для текущего или нового отчета.

Применив фильтры, щелкните "Главная>"Расширенный > расширенный фильтр/сортировка".  
Access создаст запрос, который содержит все примененные фильтры. Если нужно, чтобы в результатах запроса выводились другие столбцы, кроме отфильтрованных, дважды щелкните их в таблицах, чтобы добавить их в сетку запроса. Нажмите кнопку "Сохранить" и введите имя запроса.