



**XXIV международная научно-практическая конференция
НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ**

Квалификация "Администратор баз данных" - выполнение Модуля 3 на платформе 1С

Серкова Екатерина Геннадьевна

Преподаватель, председатель П(Ц)К «Информационных технологий»

30.01.2024

—

31.01.2024

Модуль 1. Разработка, администрирование и защита баз данных

Задание 1 (ядро ДЭ)

Написать скрипт:

- создания нескольких пользователей, баз данных и настройки права доступа к базам данных;
- шифрования всех паролей из таблицы;
- дешифрования паролей из таблицы;
- резервного копирования базы данных;
- восстановления базы данных.



Модуль 1. Разработка, администрирование и защита баз данных

(БУ+ПУ)

- Спроектировать ER-диаграмму на основании описания предметной области.
 - Создать базу данных по спроектированной диаграмме, процедуру на проверку корректности данных, триггер на изменение.
-

Модуль 2. Соадминистрирование баз данных и серверов

(БУ+ПУ)

Написать скрипты, которые позволяют:

- найти информацию и провести вычисления;
 - удалить данные;
 - обновить данные.
-

Модуль 3: Разработка модулей программного обеспечения для компьютерных систем

- Создайте модуль программного обеспечения, который позволит анализировать информацию из созданной базы данных

Работа с заказами клиентов

Выберите клиента: Иванов И. И.

Введите строку поиска:

Выберите поле для сортировки

Клиент По возрастанию
Дата заказа По убыванию
Статус заказа

Клиент	Телефон	Электронная почта	Дата заказа	Статус заказа
Иванов И. И.	+762535589633	ivan@yandex.ru	25.02.2023	Выполнен
Иванов И. И.	+762535589633	ivan@yandex.ru	15.03.2023	В работе
Петров П. П.	+795632258966	petrov@mail.ru	15.03.2023	Выполнен
Смирнов В. П.	+792535165899	smirnov@mail.ru	15.05.2023	Новый

Первый шаг - структура

- Заказы клиентов (документ). Реквизиты: статус заказа;
- Клиенты (справочник). Реквизиты: телефон, адрес электронной почты
- Статусы заказов (перечисление). Значения: Новый, В работе, Выполнен

Второй шаг - форма

Работа с заказами клиентов

Выберите клиента: Выберите поле для сортировки

Введите строку поиска:

По возрастанию
 По убыванию

Клиент	Телефон	Электронная почта	Дата заказа	Статус заказа

Реквизит	Использовать всегда	Тип
ОтборКлиент	<input type="checkbox"/>	СправочникСсылка.Клиенты
ПоляСортировки	<input type="checkbox"/>	СписокЗначений
ПорядокСортировки	<input type="checkbox"/>	Строка
СписокЗаказовКлиентов	<input type="checkbox"/>	ДинамическийСписок
СтрокаПоиска	<input type="checkbox"/>	Строка

Третий шаг - программирование

Общий модуль

```
Общий модуль: РаботаСДинамическимиСписками: Модуль
// Отбор по клиенту в динамическом списке
Процедура ОтборПоКлиенту(Клиент, СписокОтбора) Экспорт
    СписокОтбора.Отбор.Элементы.Очистить();
    ЭлементОтбора = СписокОтбора.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));
    ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("Клиент");
    ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;
    ЭлементОтбора.ПравоеЗначение = Клиент.Ссылка;
    ЭлементОтбора.Использование = Истина;
КонечПроцедуры

// Изменить порядок сортировки динамического списка
Процедура ИзменитьПорядокСортировки(ПолеСортировки, СписокСортировки, ПорядокСортировки) Экспорт
    СписокСортировки.Порядок.Элементы.Очистить();
    ЭлементПорядка = СписокСортировки.Порядок.Элементы.Добавить(Тип("ЭлементПорядкаКомпоновкиДанных"));
    Если ПолеСортировки = "" Тогда
        ЭлементПорядка.Поле = Новый ПолеКомпоновкиДанных("Ссылка.СтатусЗаказа");
    Иначе
        ЭлементПорядка.Поле = Новый ПолеКомпоновкиДанных(ПолеСортировки);
    КонечЕсли;
    ЭлементПорядка.РежимОтображения = РежимОтображенияЭлементыНастройкиКомпоновкиДанных.Авто;
    Если ПорядокСортировки = "Возр" Тогда
        ЭлементПорядка.ТипУпорядочивания = НаправлениеСортировкиКомпоновкиДанных.Возр;
    Иначе
        ЭлементПорядка.ТипУпорядочивания = НаправлениеСортировкиКомпоновкиДанных.Убыв;
    КонечЕсли;
    ЭлементПорядка.Использование = Истина;
КонечПроцедуры

// Сбросить отбор в динамическом списке
Процедура СброситьОтбор(СписокОтбора) Экспорт
    СписокОтбора.Отбор.Элементы.Очистить();
КонечПроцедуры

// Отобразить поля по совпадению
Процедура ОтобразитьПоляПоСовпадению(ПолеОтбора, СписокОтбора, СтрокаПоиска) Экспорт
    ЭлементУсловногоОформления = СписокОтбора.УсловноеОформление.Элементы.Добавить();
    ЭлементОтбора = ЭлементУсловногоОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));
    Если ПолеОтбора.Имя = "СписокЗаказовКлиентовКлиент" Тогда
        ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("Клиент.Наименование");
    Иначе
        ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных(СтрРазделить(ПолеОтбора.ПутьКДанным, ".") [1]);
    КонечЕсли;
    ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Содержит;
    ЭлементОтбора.ПравоеЗначение = СтрокаПоиска;
    ПолеОформления = ЭлементУсловногоОформления.Поля.Элементы.Добавить();
    ПолеОформления.Поле = Новый ПолеКомпоновкиДанных(СтрРазделить(ПолеОтбора.ПутьКДанным, ".") [1]);
    ЭлементУсловногоОформления.Оформление.УстановитьЗначениеПараметра("Цветфона", WebЦвета.Аквамарин);
КонечПроцедуры
```


Третий шаг - программирование

Модуль формы

(При создании формы)

```
#Область ОбработчикиСобытийФормы
&НаСервере
Процедура ПриСозданииНаСервере (Отказ, СтандартнаяОбработка)
    ПоляСортировки.Добавить ("Клиент", "Клиент");
    ПоляСортировки.Добавить ("Дата", "Дата заказа");
    ПоляСортировки.Добавить ("", "Статус заказа");
    ПоляСортировки.Добавить ("Телефон", "Телефон");
    ПоляСортировки.Добавить ("ЭлектроннаяПочта", "Электронная почта");
КонецПроцедуры
#КонецОбласти
```

Третий шаг - программирование

Модуль формы (Сортировка)

```
#Область Сортировки
&НаКлиенте
Процедура ПорядокСортировкиПриИзменении (Элемент)
    Если ЗначениеЗаполнено (ПорядокСортировки) Тогда
        ПорядокСортировкиПриИзмененииНаСервере (Элементы.ПоляСортировки.ТекущиеДанные.Значение);
    КонецЕсли;
КонецПроцедуры
&НаКлиенте
Процедура ПоляСортировкиПриИзменении (Элемент)
    ПоляСортировкиПриИзмененииНаСервере (Элементы.ПоляСортировки.ТекущиеДанные.Значение);
КонецПроцедуры
&НаСервере
Процедура ПорядокСортировкиПриИзмененииНаСервере (ПолеСортировки)
    РаботаСДинамическимиСписками.ИзменитьПорядокСортировки (ПолеСортировки, СписокЗаказовКлиентов, ПорядокСортировки);
КонецПроцедуры
&НаСервере
Процедура ПоляСортировкиПриИзмененииНаСервере (ПолеСортировки)
    РаботаСДинамическимиСписками.ИзменитьПорядокСортировки (ПолеСортировки, ПоляСортировки, ПорядокСортировки);
КонецПроцедуры
#КонецОбласти
```

Третий шаг - программирование

Модуль формы (Фильтрация по клиенту)

```
#Область_фильтрацииПоКлиенту
&НаКлиенте
Процедура_фильтроватьПоКлиенту(Команда)
    Если НЕ ЗначениеЗаполнено(ОтборКлиент) Тогда
        Сообщение = Новый СообщениеПользователю;
        Сообщение.Текст = "Выберите клиента из списка.";
        Сообщение.Поле = "ОтборКлиент";
        Сообщение.Сообщить();
        Возврат;
    КонецЕсли;
    фильтроватьПоКлиентуНаСервере();
- КонецПроцедуры
&НаКлиенте
Процедура_СброситьФильтрПоКлиенту(Команда)
    Если СписокЗаказовКлиентов.Отбор.Элементы.Количество() = 0 Тогда
        Сообщить("Все записи уже показаны.", СтатусСообщения.Информация);
        Возврат;
    КонецЕсли;
    СброситьФильтрПоКлиентуНаСервере();
- КонецПроцедуры
&НаСервере
Процедура_фильтроватьПоКлиентуНаСервере()
    РаботаСДинамическимиСписками.ОтборПоКлиенту(ОтборКлиент, СписокЗаказовКлиентов);
- КонецПроцедуры

&НаСервере
Процедура_СброситьФильтрПоКлиентуНаСервере()
    РаботаСДинамическимиСписками.СброситьОтбор(СписокЗаказовКлиентов);
- КонецПроцедуры
#КонецОбласти
```

Третий шаг - программирование

Модуль формы (Поиск в списке)

```

$НаСервере
функция ЗначениеПоискаНайдено ()
    Перец ПоискПоСтрокам, ПоискПоПеречислениям;
    Результат = Ложь;
    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        |   ЗаказПокупателя.Клиент КАК Клиент,
        |   ПРЕДСТАВЛЕНИЕ (ЗаказПокупателя.СтатусЗаказа) КАК СтатусЗаказа,
        |   ЗаказПокупателя.Клиент.ЭлектроннаяПочта КАК ЭлектроннаяПочта,
        |   ЗаказПокупателя.Клиент.Телефон КАК Телефон,
        |   ЗаказПокупателя.Дата КАК ДатаЗаказа
        |ИЗ
        |   Документ.ЗаказПокупателя КАК ЗаказПокупателя
        |ГДЕ
        |   (ЗаказПокупателя.Клиент.Телефон ПОДОБНО &СтрокаПоиска
        |       ИЛИ ЗаказПокупателя.Клиент.ЭлектроннаяПочта ПОДОБНО &СтрокаПоиска
        |       ИЛИ ЗаказПокупателя.Клиент.Наименование ПОДОБНО &СтрокаПоиска
        |       );";

    Запрос.УстановитьПараметр ("СтрокаПоиска", "%" + СтрокаПоиска + "%");
    РезультатЗапроса = Запрос.Выполнить ().Выбрать ();
    Если РезультатЗапроса.Количество () > 0 Тогда
        ПоискПоСтрокам = Истина;
    КонечЕсли;
    Для каждого ЗначениеПеречисления Из Перечисления.СтатусЗаказа Цикл
        Если СтрЧислоВхождений (НРег (Строка (ЗначениеПеречисления)), НРег (СтрокаПоиска)) > 0 Тогда
            ПоискПоПеречислениям = Истина;
        КонечЕсли;
    КонечЦикла;
    Если ПоискПоСтрокам = Истина ИЛИ ПоискПоПеречислениям = Истина Тогда
        Результат = Истина;
    КонечЕсли;
    Возврат Результат;
Конечфункции

$НаСервере
Процедура ОтборПоСтрокеПоискаНаСервере (ПолеОтбора)
    РаботаСДинамическимиСписками.ОтобратьПоляПоСовпадению (ПолеОтбора, СписокЗаказовКлиентов, СтрокаПоиска);
КонечПроцедуры
#КонецОбласти

```

торите команду.";

Варианты решения

С помощью общего модуля

<https://disk.yandex.ru/d/p5atCFafNxrEQ>



Без использования общего модуля

<https://disk.yandex.ru/d/UzAkl2AknozDpg>





**СПАСИБО
ЗА ВНИМАНИЕ!**