

персонажа. Кроме этого существует онлайн-чат, в котором пользователи могут обсудить игру, объединиться, или же просто поделиться своим опытом.

Принцип взаимодействия заключается в том, что пользователь может вводить данные как вручную, так и с помощью кнопок бота. Все зависит от сцены действия. Сам бот имеет юзернейм @ShelterGame_bot.

Практическая задача:

- В телеграме создание любого бота происходит с помощью одного официального бота – BotFather. Поэтому с помощью него создаем нового бота, указывая его имя, описание и юзернейм. Далее от BotFather TOKEN и с помощью него подключаем нашего бота к Telegram
- Было принято решение выбрать язык программирования Python, так как он достаточно мощный и имеет предельно простой синтаксис. Поэтому подключаем к проекту библиотеку для работы с API – PyTelegramBotAPI, и с помощью неё создаём обработчик сообщений, который реагирует на сообщения пользователя. Для исключения ошибок распознавания сообщений и удобства пользователя мы подключаем специальные кнопки для управления игровым процессом.
- Создаём логику для бота, чтобы он правильно реагировал на обращения к нему пользователя.
- Подключаем к проекту базу данных, в которой будут храниться все данные о пользователе. В качестве базы данных использовалась SQLite, потому что она легко встраивается и очень мощна благодаря технологиям обслуживающих библиотек.
- Переносим нашего бота на удалённый VDS (Virtual Dedicated Server, виртуальный выделенный сервер), так как бот должен оставаться включенным 24/7, а на территории РФ подключение к Telegram блокируется.
- Вносим сценарий и делаем визуальное оформление.

В итоге, затратив небольшое количество времени и сил, получилось создать вполне функционального бота для развлекательных целей.

Список использованных источников

1. Telegram APIs. - URL: <https://core.telegram.org> (дата обращения: 21.10.2018).
2. pyTelegramBotAPI. - URL: <https://github.com/eternnoir/pyTelegramBotAPI/#logging> (дата обращения: 21.10.2018).
3. SQLite Documentation. - URL: <https://www.sqlite.org/docs.html> (дата обращения: 21.10.2018).

20.51.23

ОПТИМИЗАЦИЯ ПРОГРАММЫ 1С, ИСПОЛЬЗУЕМОЙ ДЛЯ ПРИЁМА И ЗАКУПКИ ТОВАРОВ

Автор: Макарова Анастасия Владимировна, студентка третьего курса Филиала «Протвино» государственного университета «Дубна»

Научный руководитель: к.т.н., доцент Астафьева Марина Петровна

Аннотация

Работа посвящена оптимизации работы с товарами. В статье рассматриваются плюсы использования в программных решениях внешних объектов, конкретно отчётов и обработок, также перечисляются способы работы в 1С: Предприятия. Задача –

создать обработку, позволяющую читать информацию о поступающих товарах из файла Excel, и отчёт, отображающий необходимость закупки товаров.

Annotation

The article is devoted to the optimization of work with the goods. The article discusses the benefits of using in software solutions of external objects, specifically reports and processing, also lists the ways of working in 1C: Enterprise. The task is to create a processing that allows you to read information about incoming goods from the Excel file, and a report showing the need for purchasing goods.

Ключевые слова: 1C: Предприятие, файловый режим работы, внешняя обработка, внешний отчёт, структура компоновки данных (СКД).

Keywords: 1C: Enterprise, file mode, external processing, external report, data composition structure (DCS).

Значимость информационных технологий с каждым годом растёт. Причиной этому является технологический прогресс. Он позволяет, по мере возможностей, автоматизировать многие процессы, иными словами, заменять человеческую деятельность различными программами или оборудованием. Примером могут служить базы данных.

Базы данных облегчают хранение, поиск и удаление различной информации. По этой причине почти каждая организация в наше время имеет свою базу данных. В ней хранятся вся информация об организации, начиная от личных данных сотрудников и заканчивая статистикой по работе за определённый период. Вся информация хранится в виде таблиц, которых может быть сколь угодно много, и уже с помощью различных запросов к этим таблицам можно получить и структурировать любую информацию.

В настоящее время существует множество программ для работы с базами данных: MySQL, Oracle, PostgreSQL и многие другие. Помимо общих программ существует те, работа которых более специализирована. Примером такой программы может служить 1C: Предприятие. Оно предназначено для автоматизации деятельности на предприятии.

В данной работе описана технология создания внешней обработки и внешнего отчета, позволяющих оптимизировать учёт и приём товара.

Работа с 1C следует начать с того, чтобы решить, какой способ хранения информационной базы будет использоваться. В данном случае использовался файловый вариант информационной базы, поскольку она позволяет быстро и легко устанавливать систему и осуществлять работы с ней, но данный вариант предоставляет не все возможности работы с данными и их защиты, в отличие от клиент – серверного варианта. На рисунках 1 и 2 представлены схемы работы с файловой и серверной базой.

Зачастую в документах реализации или приёма номенклатура вводится посредством подбора, на основании других документов или вручную. Но их можно ввести, используя файл Excel. Именно с этой целью разрабатывалась внешняя обработка.

Внешние обработки представляют собой обработки, не входящие в состав прикладного решения и хранящиеся в файлах с разрешением *.erf. Их плюс заключается в том, что их можно включить в любое прикладное решение, но при этом не меняя его код, тем самым не требуется постоянное сохранение конфигурации при проектировании или отладке.

Разработанная обработка содержит в себе форму для ввода информации о файле и программный код, позволяющий считывать из него информацию. Посредством формы пользователем вводится информация о файле, листе, с которого считывается

информация, и информация о колонках, в которых хранятся данные. На рисунке 3 представлен интерфейс формы.

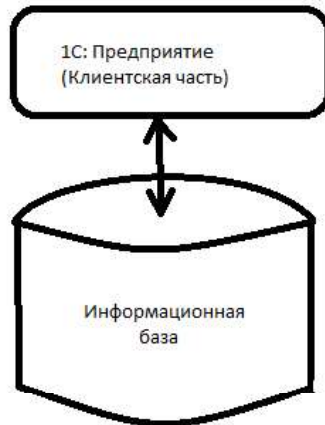


Рисунок 1. Файловый вариант работы



Рисунок 2: Серверный вариант работы

Имя файла: ...

Лист для загрузки: ▾

Загружать строки с: по:

Настройки

Колонка "Код товара": ▴ ▾

Колонка "Наименование товара": ▴ ▾

Колонка "Ед.изм.": ▴ ▾

Колонка "Количество": ▴ ▾

Колонка "Цена": ▴ ▾

Колонка "Стоимость без НДС": ▴ ▾

Колонка "НДС": ▴ ▾

Колонка "Сумма НДС": ▴ ▾

Колонка "Стоимость всего": ▴ ▾

Колонка "Код производителя": ▴ ▾

Рисунок 3: Пример интерфейса формы

Программный код создается в модуле формы и модуле объекта самой обработки. Структура кодов состоит из процедур и функций, их отличие в том, что вторые возвращают ту или иную информацию, а первые нет. Основная программа писалась в модуле формы. Одни из главных элементов формы – кнопка «Загрузить». После её нажатия пользователь либо увидит заполненный документ реализации, либо, если отметил поле «Закрыть после загрузки», перечень всех документов реализации. Для удобства использования, если при загрузке будут ошибки или вводиться новая информация, которой нет в информационной базе, пользователю будут выводиться сообщения с информацией. Например, если поступила новая номенклатура, которой ещё нет в справочнике «Номенклатура», хранящем всю информацию о когда-либо поступающих товар, то пользователь увидит сообщение: «Создана новая номенклатура:». После двоеточия будет выводиться название товара. Ниже, на рисунке 4, приведён фрагмент кода для кнопки «Загрузить».

Кроме внешней обработки был создан внешний отчёт, позволяющий видеть, сколько товара осталось на складе, и когда его меньше, чем должно быть, количество выделяется красным цветом. Информация о том сколько его должно быть вводиться пользователем при добавлении новой номенклатуры. Отчет создавался посредством схемы компоновки данных. Запроса, для создания отчёта, представлен на рисунке 5.

Основное преимущество внешних обработок и отчётов заключается в том, что они могут создаваться при работающем приложении, из чего следует, что они не требуют постоянного обновления конфигурации. К ещё одному преимуществу можно отнести, их относительную простоту и удобство включения в почти любое программное решение.

В заключении следует отметить, что в дальнейшем планируется доработка внешней обработки, чтобы её можно было использовать не только при приёмке товара, но и при продаже или списании.

```

Для Стр = Объект.НачСтрока По КонСтр Цикл
    СтруктураСтроки = ПодготовитьСтроку (ЛистФайла, Стр) ;

    Если ПустаяСтрока (СтруктураСтроки.КодТовара) = Ложь ИЛИ
ПустаяСтрока (СтруктураСтроки.НаименованиеТовара) = Ложь Тогда
        НовСтрока = ДокОбъект [ИмяТЧ].Добавить () ;

        Если ТипЭнч (Объект.Документ) =
Тип ("ДокументСсылка.РеализацияТоваровУслуг") Тогда
            ПараметрыНоменклатуры = СоздатьНайтиНоменклатуру (СтруктураСтроки) ;
        Иначе
            ПараметрыНоменклатуры = СоздатьНайтиНоменклатуру (СтруктураСтроки,
Истина) ;

        КонецЕсли;

        Номенклатура = ПараметрыНоменклатуры.Номенклатура;

        Если Номенклатура.Пустая() Тогда
            Возврат;
        КонецЕсли;

```

Рисунок 4: Фрагмент кода для кнопки "Загрузить"

```

ВЫБРАТЬ
    ТоварыНаСкладахОстатки.Номенклатура,
    СУММА(ТоварыНаСкладахОстатки.ВНаличииОстаток) КАК ВНаличииОстаток,
    спрНоменклатура.МинКол
ИЗ
    РегистрНакопления.ТоварыНаСкладах.Остатки КАК ТоварыНаСкладахОстатки
    ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Номенклатура КАК спрНоменклатура
    ПО ТоварыНаСкладахОстатки.Номенклатура = спрНоменклатура.Ссылка

СГРУППИРОВАТЬ ПО
    ТоварыНаСкладахОстатки.Номенклатура,
    спрНоменклатура.МинКол

```

Рисунок 5: Запрос, для создания отчёта

Список использованных источников

1. Радченко М.Г. Хрусталева Е.Ю. 1С: Предприятие 8.2 Практическое пособие разработчика. Примеры и типовые приемы. – М.: ООО «1С-Публишинг», 2009.-874с.
2. Хрусталева Е.Ю. Разработка сложных отчетов в 1С. Предприятии 8. Система компоновки данных. – М.: ООО «1С-Публишинг», 2008.-515с.
3. Внешняя обработка – URL: http://v8.1c.ru/overview/Term_000000601.htm (дата обращения: 20.10.2018)

20.53.19

РАЗРАБОТКА ФУНКЦИЙ УЧЁТА РАБОЧЕГО ВРЕМЕНИ С ПРИМЕНЕНИЕМ СРЕДСТВ T-SQL

Автор: Макарова Анастасия Владимировна, студентка третьего курса Филиала «Протвино» государственного университета «Дубна»

Научный руководитель: к.т.н., доцент кафедры Кульман Татьяна Николаевна