**Министерство образования и молодежной политики СК**

**ГБПОУ «Лермонтовский региональный многопрофильный колледж»**

**Работа допущена к защите**

Зав.кафедрой «Программирование

в компьютерных системах»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Г.Д.Томулевич

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2015 г.

**КУРСОВАЯ РАБОТА**

###### Пояснительная записка

на тему:*\_\_\_\_\_\_*\_\_\_\_\_\_\_*\_\_\_\_\_\_\_\_\_\_*Ресторан Calypso\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ПМ.02 Разработка и администрирование баз данных**

**специальность 09.02.03 «Программирование в компьютерных системах»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент гр. | | ***ПР4*** |  |  | | **И.В.Шведов** |
|  | | группа |  | подпись | | Ф.И.О. студента |
| Проверил |  | | | | **А.А.Светличный** | |
|  | оценка и подпись | | | | Ф.И.О. преподавателя | |

Лермонтов, 2015

Содержание

стр.

[Ведение 3](#_Toc437962523)

[1. ПЛАНИРОВАНИЕ СОЗДАНИЯ Базы данных «Ресторан Calypso» 4](#_Toc437962524)

[1.1. Основные понятия управления проектами 4](#_Toc437962525)

[2. Составление требований заказчика 7](#_Toc437962526)

[3. Проектирование Базы данных «Ресторан Calypso» 9](#_Toc437962527)

[4. Разработка Базы данных «Ресторан Calypso» 11](#_Toc437962528)

[4.1. Создание таблиц 11](#_Toc437962529)

[4.2. Создание запросов 13](#_Toc437962530)

[4.3. Создание форм 19](#_Toc437962531)

[4.4. Создание отчетов 24](#_Toc437962532)

[5. Тестирование Базы данных «Ресторан Calypso» 26](#_Toc437962533)

[6. Сопровождение Базы данных «Ресторан Calypso» 27](#_Toc437962534)

[Заключение 28](#_Toc437962535)

[Список литературы 29](#_Toc437962536)

# Ведение

База данных – это единое, вместительное хранилище разнообразных данных и описаний их структур, которое после своего определения, осуществляемого отдельно и независимо от приложений, используется одновременно многими приложениями.

Кроме данных база данных может содержать средства, позволяющие каждому из пользователей оперировать только теми данными, которые входят в его компетенцию. В результате взаимодействия данных, содержащихся в базе, с методами, доступными конкретным пользователям, образуется информация, которую они потребляют и на основании которой в пределах собственной компетенции производят ввод и редактирование данных

При создании баз данных необходимо уделить особое внимание тому, чтобы данные можно было широко использовать в различного рода приложениях и чтобы способы исполь­зования данных можно было легко и быстро изменять. До появления электронных баз данных было чрезвычайно трудно изменить способ организации используемых данных. Различные про­граммисты по-разному представляли данные и постоянно стремились их модифицировать по мере возникновения новых задач. Эти модификации вызывали значительные изменения существующих программ, и поэтому их выполнение обходилось дорого.

Таким образом, для конечного пользователя важнейшей задачей является правильно расставить приоритеты в процессе создания БД и четко для себя установить, что требуется от его БД, которую в дальнейшем реализует программист.

# 1. ПЛАНИРОВАНИЕ СОЗДАНИЯ Базы данных «Ресторан Calypso»

### 1.1. Основные понятия управления проектами

Проектом[[1]](#footnote-1) называется совокупность распределенных во времени мероприятий или работ, направленных на достижение поставленной цели. Примерами проектов являются строительство зданий, комплексов, предприятий, освоение выпуска нового вида продукции, проведение модернизации производства, разработка программного продукта и т.д.

Проект обладает определенными свойствами:

1. Проект всегда имеет четко определенную цель, которая выражается в получении некоторого результата. Достижение этого результата означает успешное завершение и окончание проекта. Например, для проекта строительства здания результатом является само здание, принятое в эксплуатацию;
2. Проект имеет четко очерченное начало, которое совпадает с началом первой работы, направленной на достижение поставленной цели. Начало может задаваться директивно, либо рассчитываться в результате составления плана работ по проекту;
3. Проект имеет четко очерченный конец, который совпадает с концом последней работы, направленной на получение заданного результата. Как и начало, конец проекта может задаваться директивно, или рассчитываться при составлении плана работ. Например, для проекта строительства здания конец проекта совпадает с датой акта сдачи/приемки его в эксплуатацию;
4. Проект исполняется командой, в состав которой входит руководитель проекта, менеджеры, исполнители. Помимо основной команды в нем могут участвовать сторонние исполнители, команды и организации, которые привлекаются на временной основе для выполнения отдельных работ;
5. При реализации проекта используются материальные ресурсы. Их номенклатура и количество определяются характером проекта и входящих в него работ. Так при строительстве дома используются песок, щебень, цемент, кирпич и т.п;
6. Проект имеет бюджет. Стоимость проекта складывается из стоимости израсходованных материальных ресурсов, затрат по оплате труда реализующей его команды и прочих расходов, связанных с особенностями конкретных видов работ;

Проект имеет ограничения трех видов:

1. Ограничения по бюджету устанавливают предельную стоимость всего проекта или отдельных видов работ;
2. Ограничения по времени задают предельные сроки окончания либо всего проекта, либо некоторых работ. Например, тестовые испытания должны проводиться в присутствии представителя заказчика, который будет присутствовать в заданный период времени;
3. Ограничения по ресурсам определяются ограниченным составом команды или графиками поступления материальных ресурсов.

Жизненный цикл проекта – это промежуток времени между моментами его начала и завершения. Он делится на четыре фазы:

1. Концептуальная фаза. Включает формулирование целей, анализ инвестиционных возможностей, обоснование осуществимости (технико-экономическое обоснование) и планирование проекта;
2. Фаза разработки проекта. Включает определение структуры работ и исполнителей, построение календарных графиков работ, бюджета проекта, разработку проектно-сметной документации, переговоры и заключение контрактов с подрядчиками и поставщиками;
3. Фаза выполнения проекта. Включает работы по реализации проекта, в том числе строительство, маркетинг, обучение персонала и т.п.;
4. Фаза завершения проекта. Включает в общем случае приемочные испытания, опытную эксплуатацию и сдачу проекта в эксплуатацию;

Результат проекта – это некоторая продукция или полезный эффект, создаваемые в ходе реализации проекта. В качестве результата, в зависимости от цели проекта, могут выступать: научная разработка, новый технологический процесс, программное средство, строительный объект, реализованная учебная программа, реструктурированная компания, сертифицированная система качества и т.д. Об успешности проекта судят по тому, насколько его результат соответствует по своим затратным, доходным, инновационным, качественным, временным, социальным, экологическим и другим характеристикам запланированному уровню.

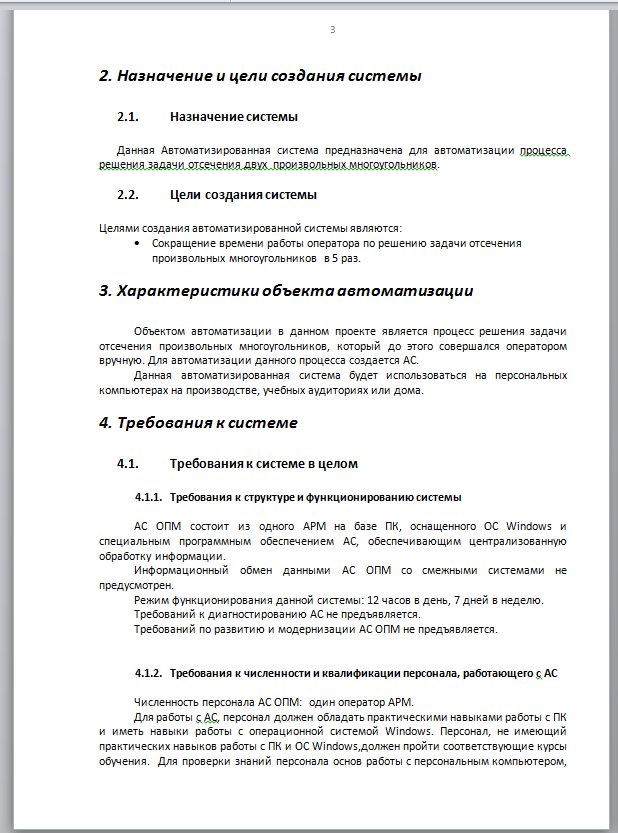
# 2. Составление требований заказчика

Существуют два диаметрально противоположных подхода к разработке технического задания (ТЗ). Первый подход заключается в том, чтобы:

* собрать исходные требования заказчика;
* отшлифовать их формулировки;
* сгруппировать требования;
* заполнить ими разделы ТЗ;
* согласовать этот текст с заказчиком.

Далее приступить к разработке системы, отдав ТЗ непосредственно разработчикам. Всё что происходит потом известно всем. Разработчики начинают задавать вопросы, ответы на которые, разработавшие ТЗ специалисты, могут найти только у специалистов заказчика.

В итоге понимание того, что же из себя должна представлять будущая система приходит не в ходе процесса разработки ТЗ, а в ходе процесса разработки системы, а в особо выдающихся случаях – в ходе приемки так называемой «готовой» системы заказчиком. Это затратный и неэффективный, но распространенный подход к созданию системы. Многие компании предпочитают его, поскольку он якобы быстрее позволяет перейти к процессу разработки системы. Это самообман дилетантов, считающих, что лучше семь раз быстро отрезать, чем один раз отмерить. Этот подход и приводит впоследствии к лоскутным, а не целостным решениям, появлению «костылей» и «заплаток» и высоким временным и материальным затратам на сопровождение и модернизацию. Схематично этот подход отражен на риc. 1.



1. Схема составления ТЗ

Изумленные новыми вопросами специалисты заказчика формируют ответы, из которых постепенно становиться ясно, что собранные требования нуждаются в изменениях. Внесение изменений приводит к новым вопросам и уточнениям требований. И так несколько итераций.

Основная проблема такого подхода в том, что при составлении ТЗ хорошо сформулированные требования заказчика просто группируются в заданных стандартом разделах ТЗ. Они не представляют собой систему требований, поскольку не связаны между собой никакими отношениями, как, впрочем, и сами разделы любого стандарта на ТЗ. Система требований – одна из моделей будущей системы, - не определяется структурой какого-либо документа, а определяется внутренними связями между требованиями.

# 3. Проектирование Базы данных «Ресторан Calypso»

Проектирование программного обеспечения – этап жизненного цикла программного обеспечения, во время которого исследуется структура и взаимосвязи элементов разрабатываемой системы. Результатом этого этапа является проект, содержащий достаточное количество информации для реализации системы. Различают проектирование архитектуры системы и детальное проектирование программных модулей.

Основная задача любого успешного проекта заключается в том, чтобы на момент запуска системы и в течение всего срока ее эксплуатации можно было обеспечить:

* требуемую функциональность системы и степень адаптации к изменяющимся условиям ее функционирования;
* требуемую пропускную способность системы;
* требуемое время реакции системы на запрос;
* безотказную работу системы в требуемом режиме, то есть: готовность и доступность системы для обработки запросов пользователей;
* простоту эксплуатации и поддержки системы;
* требуемую безопасность.

Производительность является главным фактором, который определяет эффективность системы. Хорошее проектное решение – основа высокопроизводительной системы.

В реальных условиях проектирование информационных систем – это поиск способа, который обеспечивает необходимую функциональность системы средствами имеющихся технологий с учетом заданных ограничений.

Под методологией разработки подразумевается набор методов и критериев оценки, которые используются для постановки задачи, планирования, контроля и в конечном итоге – для достижения поставленной цели. Сам процесс разработки описывается моделью, которая определяет последовательность наиболее общих этапов и получаемых результатов.

Долгое время процесс разработки ПО осуществлялся в соответствии с методиками, наработанными в инженерной области, стандартная практика поэтапного создания продукта, начиная с составления спецификаций и заканчивая поставкой заказчику. Существуют различные стандарты:

* ГОСТ (Россия);
* ISO (Европа, Россия);
* CMM (Capability Maturity Model – распространен в США);

Известны несколько основных моделей жизненного цикла ПО.

1. Каскадная модель – переход на следующий этап означает полное завершение работ на предыдущем этапе;
2. Поэтапная модель с промежуточным контролем – разработка ПО ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют уменьшить трудоемкость процесса разработки по сравнению с каскадной моделью, но время жизни каждого из этапов растягивается на весь период разработки;
3. Спиральная модель – особое внимание уделяется начальным этапам разработки: выработке стратегии, анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования). Каждый виток спирали предполагает создание некой версии продукта или какого либо его компонента; при этом уточняются характеристики и цели проекта, определяется его качество и планируются работы следующего витка спирали;
4. Активное программирование и его клоны – наиболее популярным для данной модели стало экстремальное программирование (eXtreme Programming, XP)[[2]](#footnote-2). Отцом-идеологом XP считают Кента Бека. XP является довольно молодой методологией, оценки которой весьма противоречивы: от восторженных до резко негативных. Основными принципами являются простота решений и интенсивная разработка малыми группами, активное общение в группе и обратная связь с клиентом, фактически вовлеченным в процесс разработки, а также известная доля куража.

# 4. Разработка Базы данных «Ресторан Calypso»

# 4.1. Создание таблиц

Таблица – это набор данных по конкретной теме, такой как поставки или поставщики. Использование отдельной таблицы для каждой темы означает, что соответствующие данные сохранены только один раз, что делает базу данных более эффективной и уменьшает число ошибок при вводе данных. Создание таблиц заключается в задании её полей и назначении их свойств.

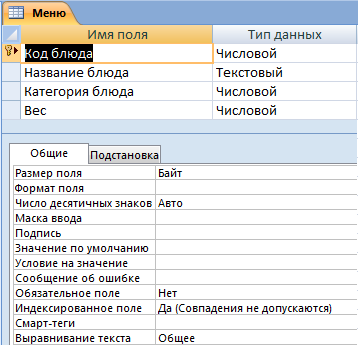
Для создания таблиц достаточно воспользоваться конструктором таблиц, с помощью которого можно определить имена полей таблицы и типы данных полей. Кроме этого, для создания наиболее часто используемых в базе данных таблиц, можно воспользоваться режимом мастера.

После, того как необходимые поля будут определены, можно определить первичный ключ для таблицы, содержащий одно или несколько полей.

Первичный ключ используется для связывания таблицы с вторичными ключами в других таблицах. Это одно или несколько полей (столбцов), комбинация значений которых однозначно определяет каждую запись в таблице. Первичный ключ не допускает значений 0 и всегда должен иметь уникальный индекс.

На примере таблицы *Меню* (рис. 2) продемонстрируем структуру создаваемых таблиц.В этой таблицы находятся поля:

1. *Код блюда* – это ключевое поле, у которого тип данных счетчик, а размер поля – байт;
2. *Название блюда* – это текстовое поле, размер поля 30 символов
3. *Категория блюда* – это поле внешнего ключа, т.е. из выпадающего списка можно выбрать значение которые хранятся в другой таблице;
4. *Вес* – это числовое поле в котором указывается вес блюда

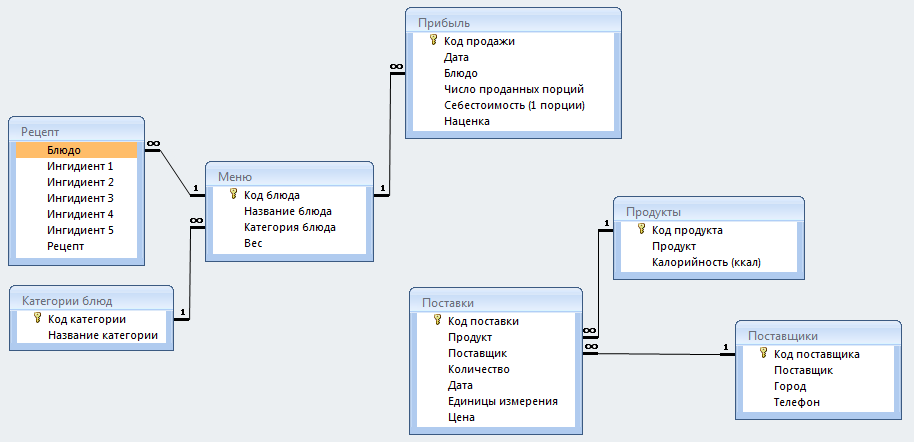


1. Структура таблицы Меню

Связь между таблицами устанавливает отношения между совпадающими значениями в ключевых полях, обычно между полями, имеющими одинаковые имена в обеих таблицах. В большинстве случаев ключевым полем одной таблицы, являющимся уникальным идентификатором каждой записи, связывается внешний ключ другой таблицы.

Связь между таблицами создается путём добавления связываемых таблиц в окно «Схема данных» с последующим перетаскиванием ключевого поля из одной таблицы в другую.

На рис. 3 изображена схема данных таблиц БД ресторана *Calypso.*



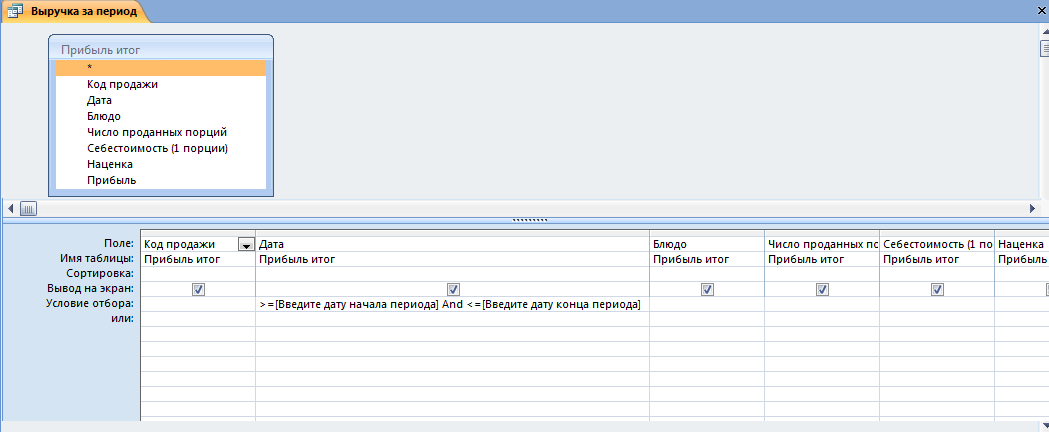
1. Схема данных таблиц БД ресторана Calypso.

# 4.2. Создание запросов

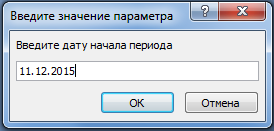
Запросы используются для просмотра, изменения и анализа данных различными способами. Запросы также можно использовать в качестве источников записей для форм, отчётов и страниц доступа к данным.

Для одной и той же таблицы можно создать множество разных запросов, каждый из которых сможет извлекать из таблицы лишь малую часть информации, но именно ту часть, которая необходима. В результате работы запроса из общей исходной базы формируется результирующая таблица, содержащая часть общей информации, соответствующей запросу. Ещё одним ценным свойством запросов является их способность выполнять итоговые вычисления.

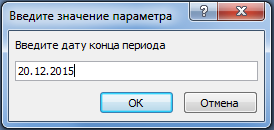
На рис. 4 изображен запрос на выборку данных за период в режиме конструктора. При выполнении данного запроса появляется всплывающее сообщение с просьбой вести начальную дату (рис. 5). После этого появляется сообщение с просьбой ввести конечную дату (рис. 6). По результату введенных дат происходит выборка данных за определенный период (рис. 7).



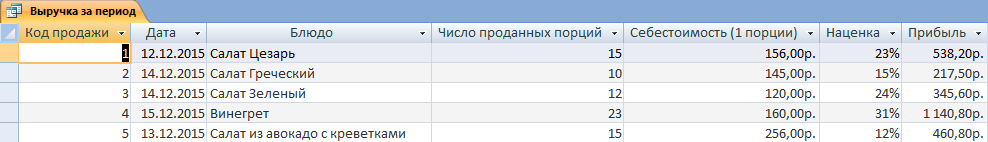
1. Запрос на выборку данных за период в режиме конструктора.



1. Всплывающее сообщение с просьбой вести начальную дату

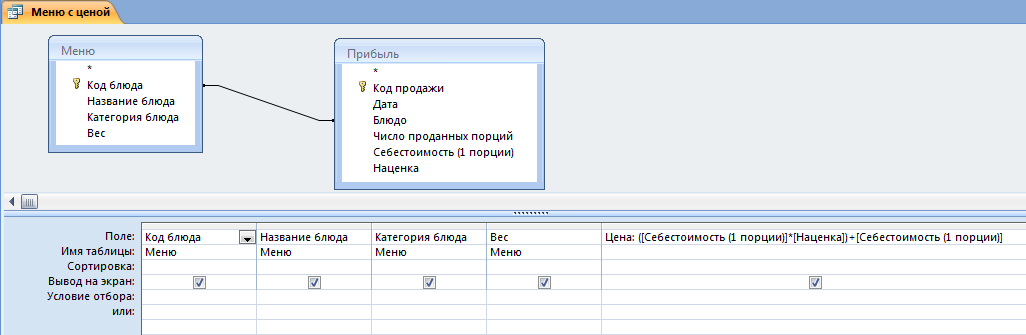


1. Всплывающее сообщение с просьбой вести конечную дату



1. Результат выполнения запроса на выборку записей за период

На рис. 8 изображен запрос в режиме конструктора на подсчет стоимости 1 порции блюда из меню. На рис 9 изображено выполнение данного запроса.

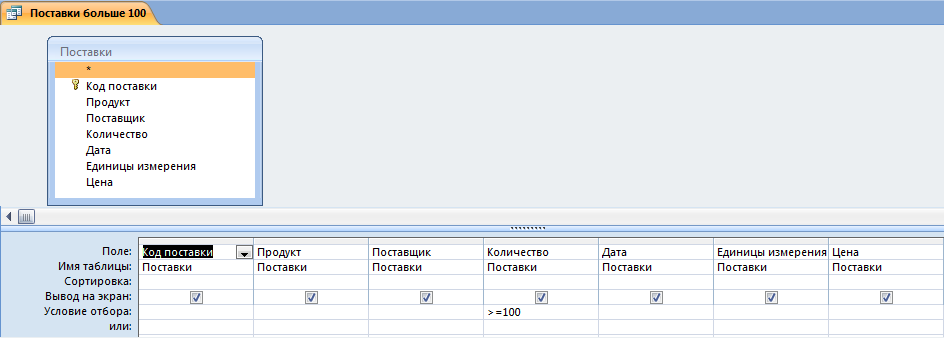


1. Запрос в режиме конструктора на подсчет стоимости 1 порции блюда из меню

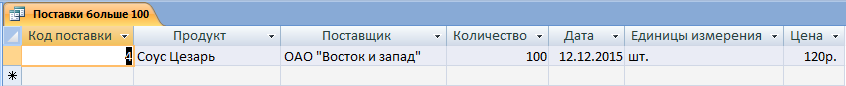


1. Пример выполнения запроса по подсчету стоимости блюда

На рис. 10 изображен запрос в режиме конструктора на выборку записей на количество поставляемых продуктов больше 100. На рис 11 изображено выполнение данного запроса.

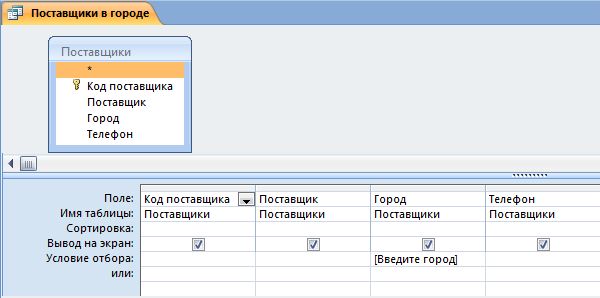


1. Запрос в режиме конструктора на выборку записей на количество поставляемых продуктов больше 100

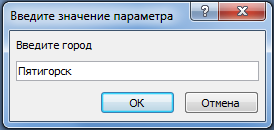


1. Пример выполнения запроса на выборку записей на количество поставляемых продуктов больше 100

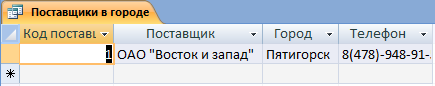
На рис. 12 изображен запрос на выборку поставщиков определенном городе в режиме конструктора. При выполнении данного запроса появляется всплывающее сообщение с просьбой вести город поставщика (рис. 13).. По результату введенных данных происходит выборка данных в определенном городе (рис. 14).



1. Запрос на выборку поставщиков определенном городе в режиме конструктора

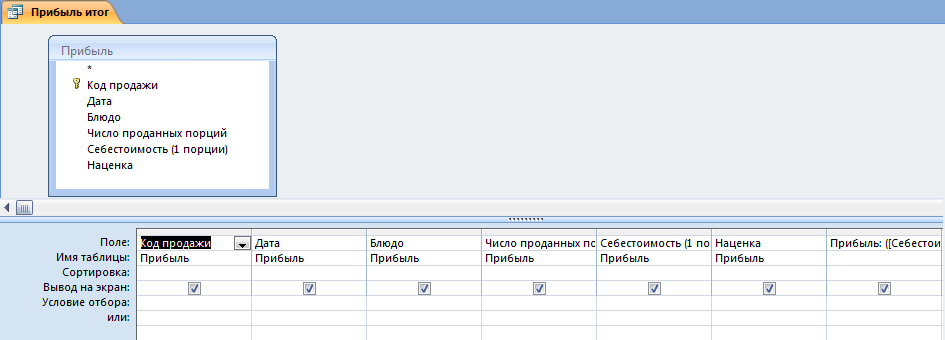


1. Сообщение с просьбой вести город поставщика

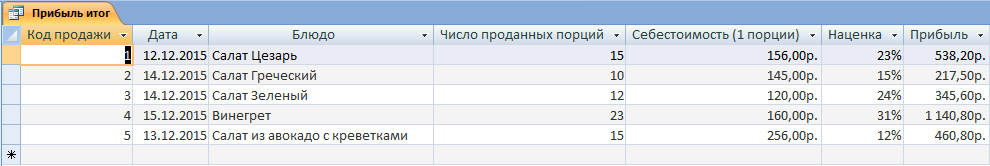


1. Пример выполнения запроса на выборку поставщиков в указанном городе

На рис. 15 изображен запрос в режиме конструктора, в котором подсчитывается сумма выручки за определенное блюдо. На рис 16 изображено выполнение данного запроса.

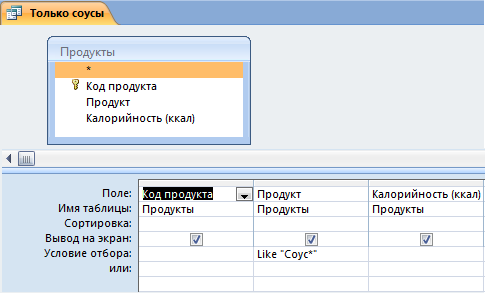


1. запрос в режиме конструктора, в котором подсчитывается сумма выручки за определенное блюдо



1. Пример выполнения запроса, в котором подсчитывается сумма выручки за определенное блюдо

На рис. 17 изображен запрос в режиме конструктора, в котором выбираются только соусы. На рис 18 изображено выполнение данного запроса.



1. запрос в режиме конструктора, в котором выбираются только соусы.



1. Пример выполнения запроса, в котором выбираются только соусы

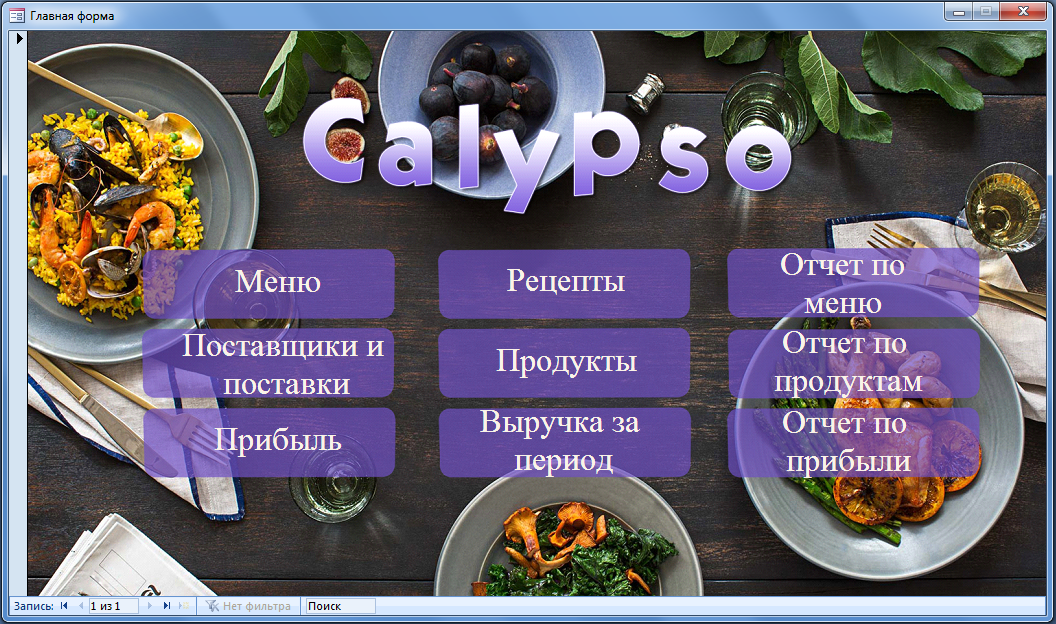
# 4.3. Создание форм

Форма – средство отображения данных на экране и управления ими. В форме можно разместить специальные элементы управления (счётчики, раскрывающиеся списки, переключатели, флажки и прочее) для автоматизации ввода. С помощью формы данные можно не только вводить, но и отображать. Запросы тоже отображают данные, но делают это виде результирующей таблицы, не имеющей почти никаких средств оформления. При выводе данных с помощью форм можно применять специальные средства оформления.

Наиболее удобными способами создания форм являются создание формы с помощью конструктора и с помощью мастера.

Мастер задаёт подобные вопросы об источниках записей, полях, макете, требуемых формах и создаёт форму на основании полученных ответов. С помощью мастера наиболее удобно создавать формы, предназначенные для ввода и редактирования данных.

На рис. 19 изображена главная форма БД *ресторан Calypso*. На ней находятся кнопки перехода к другим формам, запросам и отчетам.

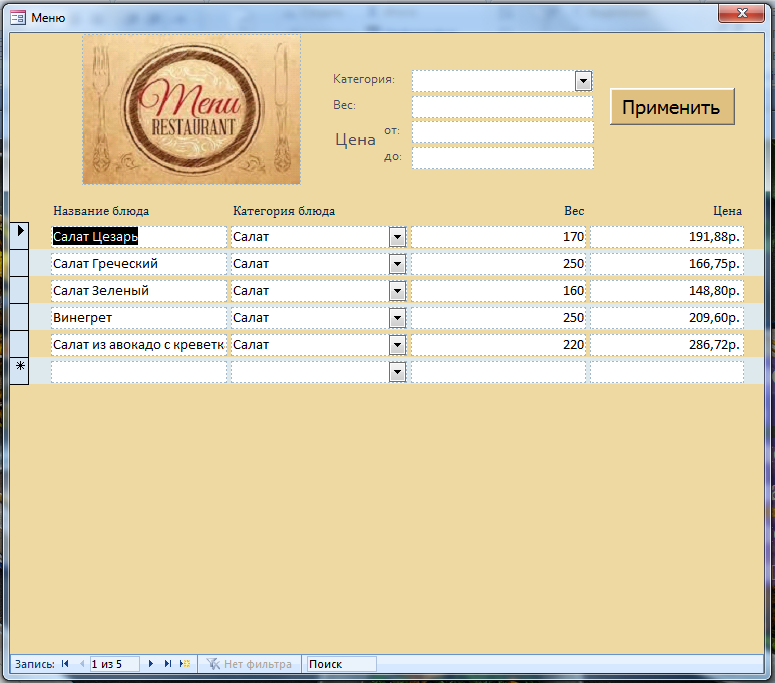


1. Главная форма БД ресторан Calypso

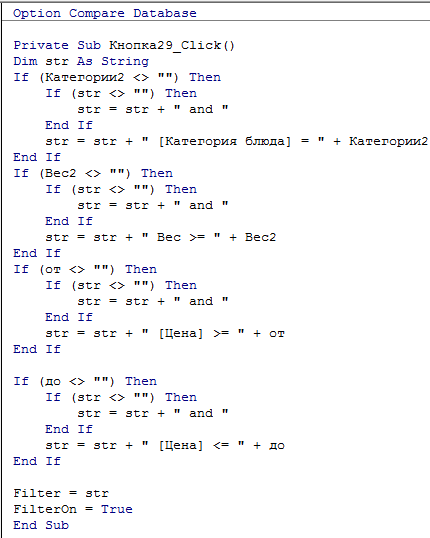
На рис. 20 изображена форма *Меню.* В ней перечислены блюда *ресторана Calypso.* В этой форме продемонстрирован способ использования VBA[[3]](#footnote-3). Для применения фильтра нужно выполнить одно из условий:

1. Выбрать категорию блюда;
2. Ввести вес блюда;
3. Указать цену;
4. Нажать на кнопку *Применить* ;

Текст программы приведен на рис. 21.

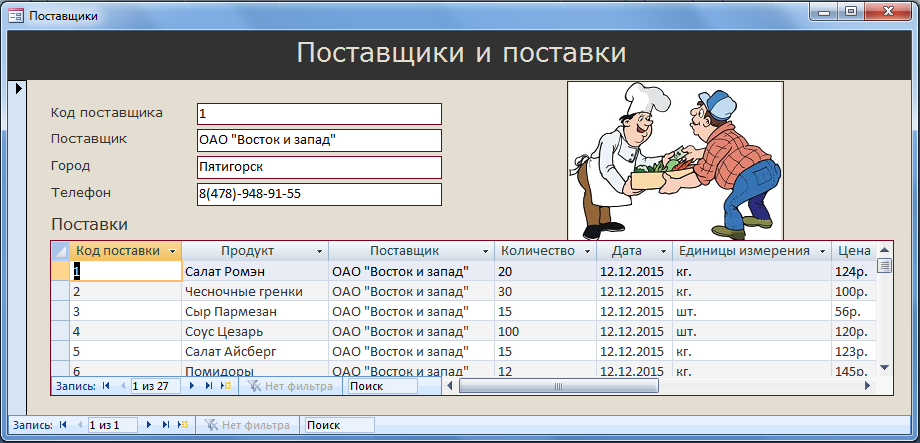


1. Форма Меню БД ресторан Calypso



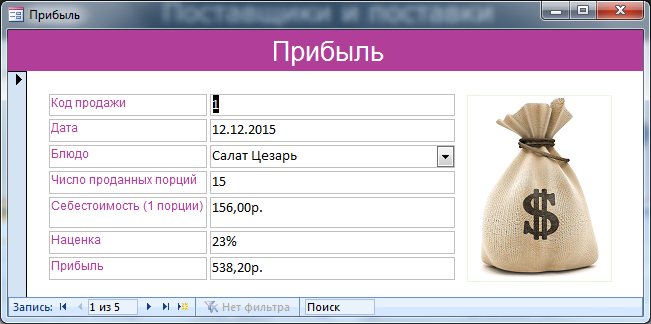
1. Листинг программы обработки нажатия кнопки применить

На рис. 22 изображена форма Поставщики и поставки. В ней перечислены поставщики и поставки в ресторан Calypso.



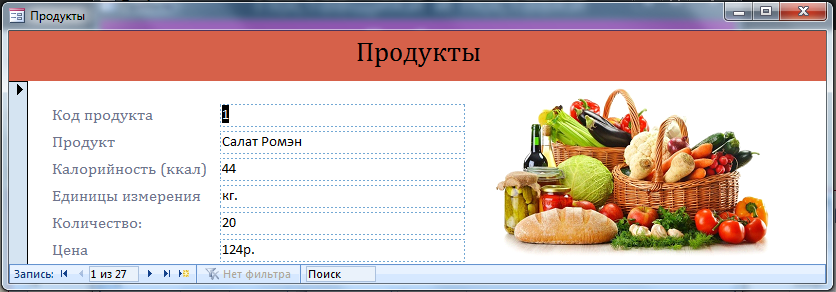
1. Форма Поставщики и поставки БД ресторан Calypso

На рис. 23 изображена форма *Прибыль*. В ней перечислены прибыль ресторана Calypso по определенному блюду.



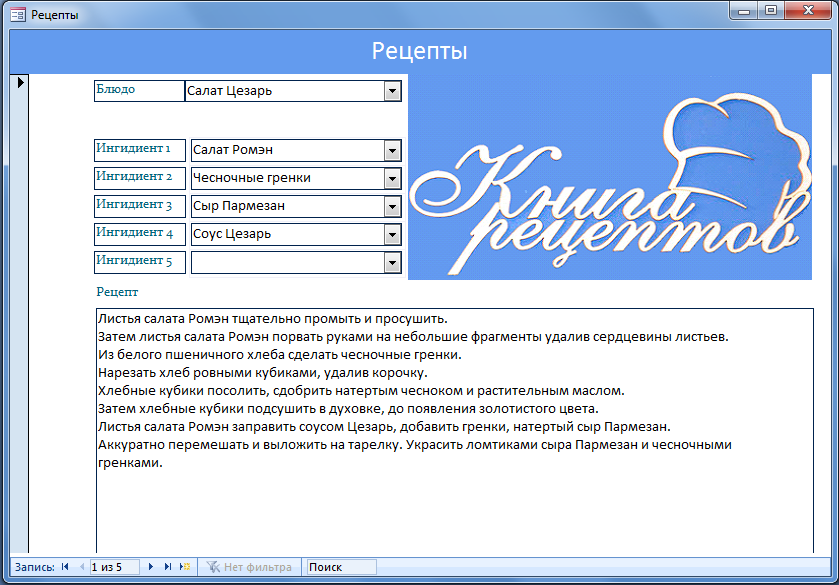
1. Форма Прибыль БД ресторана Calypso

На рис. 24 изображена форма *Продукты*. В ней перечислены продукты ресторана Calypso.



1. Форма Продукты БД ресторана Calypso

На рис. 25 изображена форма *Рецепты*. В ней перечислены рецепты ресторана Calypso.

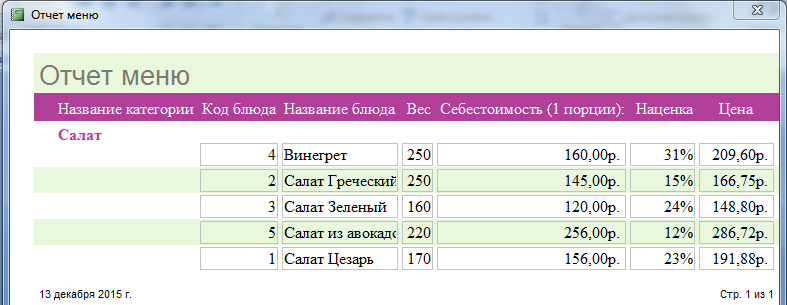


1. Форма Рецепты БД ресторана Calypso

# 4.4. Создание отчетов

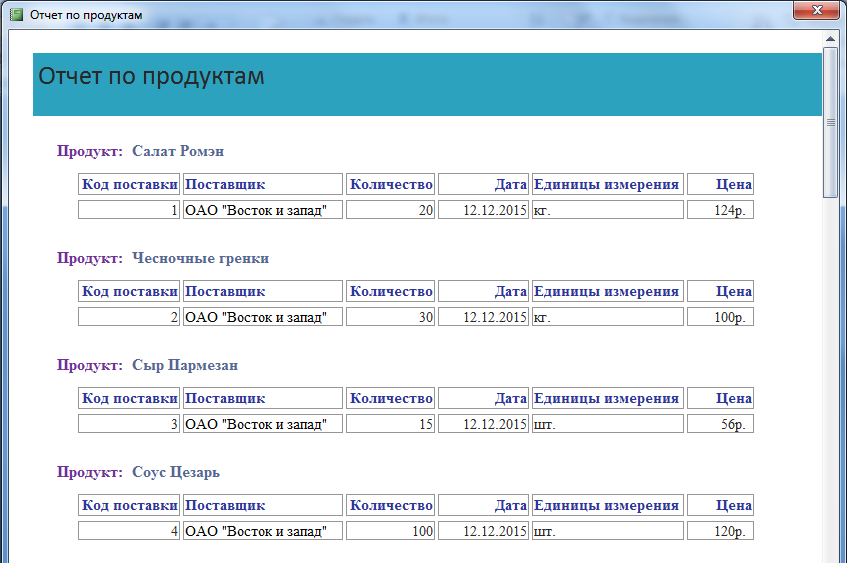
Отчёт представляет собой эффективный способ представления данных в печатном формате. Имея возможность управлять размером и внешним видом всех элементов отчёта, пользователь может отобразить сведения желаемым образом. Большая часть сведений в отчёте поступает из базовой таблицы, запроса. По своим свойствам и структуре отчёты во многом похожи на формы, но предназначены только для вывода данных, причём для вывода не на экран, а на печатающее устройство. В связи с тем отчёты отличаются тем, что в них приняты специальные меры для группирования выводимых данных и для ввода специальных элементов оформления характерных для печатных документов. Наиболее удобным способом создания отчётов является создание отчёта с помощью мастера.

На рис. 26 изображен фрагмент отчета *Меню.* В нем отображены блюда ресторана Calypso.



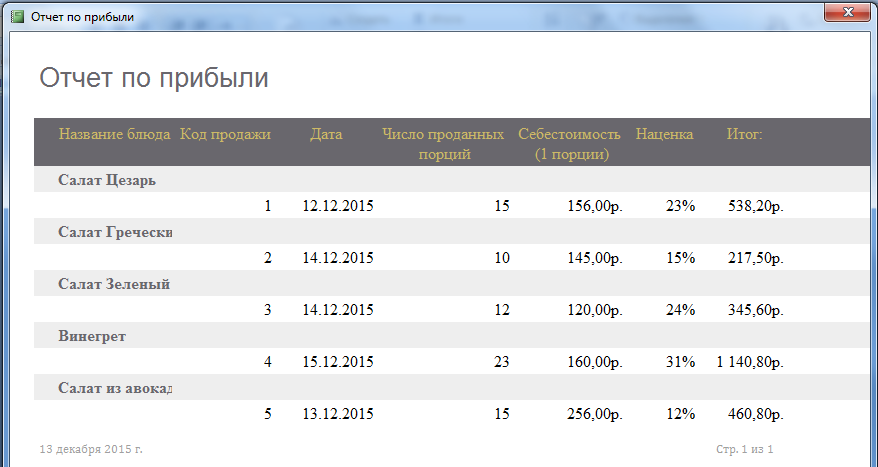
1. Фрагмент отчета Меню

На рис. 27 изображен фрагмент отчета *Продукты.* В нем отображены продукты ресторана Calypso.



1. Фрагмент отчета Продукты

На рис. 28 изображен фрагмент отчета *Прибыль.* В нем отображены прибыль ресторана Calypso.



1. Фрагмент отчета Прибыль

# 5. Тестирование Базы данных «Ресторан Calypso»

Тестирование – это любая деятельность, направленная на обнаружение ошибок в программном продукте. Тестирование проводится для того, чтобы найти ошибки в программе и тем самым повысить ее надежность, а следовательно, ценность. Если мы тестируем программу, то нам нужно окупить затраты на тестирование, каким-либо образом повысив стоимость программы.

Это можно сделать только повысив надежность программы, ради чего тестирование и проводится. Повысить надежность можно только исправлением ошибок, внесенных в процессе разработки. После тестирования нельзя гарантировать отсутствие ошибок, можно лишь говорить о некотором уровне уверенности в правильности работы системы.

Каждую ошибку следует внимательно изучить, чтобы понять, почему она возникла, что должно было быть сделано, чтобы ее предотвратить или обнаружить раньше. Удачным считается тест, который обнаружил ошибку. Если ни одна ошибка не была обнаружена, то тест считается неудачным.

Цель тестирования – показать, что программа корректно выполняет предусмотренные функции, т.е. программа соответствует спецификации. Или, более детально, цель тестирования – показать, в каких ситуациях программа не соответствует спецификации, в то время как тестовые данные используются в соответствии со спецификацией программы.

# 6. Сопровождение Базы данных «Ресторан Calypso»

Сопровождение программного обеспечения (ПО) может подразумевать как постоянное, так и периодическое обслуживание (по запросу). Постоянный вариант поддержки и сопровождения ПО больше подходит для высоконагруженных систем, второй вариант необходимо применять на проектах, которые могут содержать большой функционал или проекты, на которых необходимо отслеживать всевозможные действия пользователей, которые периодически приводят к некорректной работе (будь то неверное построение отчетов или статистики, некорректно выставленные статусы какой-либо заявке или товару и т.д.).

Техническое сопровождение включает в себя консультирование (по телефону, электронной почте, факсу и др.), мониторинг работы программно-аппаратного комплекса и удаленную диагностику критических ситуаций, поставку пакетов обновлений и обновленной документации, а также выезд инженера на место эксплуатации программно-аппаратного комплекса в случае возникновения аппаратной проблемы, которую невозможно решить удаленно. В рамках сопровождения могут быть также инициированы доработки ПО. Техническая поддержка оборудования включает в себя профилактические работы, текущий ремонт и внеплановое техническое обслуживание. Важное место в процессе оказания услуг занимает технология обратной связи с Клиентом.

# Заключение

В современном мире часто приходится работать с данными из разных источников, каждый из которых связан с определённым видом деятельности. Для координации всех этих данных необходимы определённые знания и организационные навыки. MS Access объединяет сведения из разных источников в одной реляционной базе данных. Удобный интерфейс позволяет легко ориентироваться в ней, привлекая тем самым многих разработчиков и пользователей баз данных.

Чтобы сделать любую базу в MS Access необходимо изучить предметную область и составить таблицы, в которых будет отображаться эта предметная область. В результате было сделано автоматизированное рабочее место в виде набора связанных экранных форм и отчётов, позволяющие просматривать данные по центральной больнице. MS Access позволяет управлять информацией из одного файла базы данных. В рамках этого файла данные можно разделить на отдельные таблицы; просматривать, добавлять и удалять данные в таблицах; находить и извлекать только нужные данные с помощью запросов, а также анализировать или печатать данные в заданном макете с помощью отчётов. Создание главной формы к данным позволяет пользователям легко просматривать, обновлять или анализировать данные.

Разработанная база данных проста в применении и может быть использована в любом ресторане.

# Список литературы

1. Бекаревич Ю. Пушкина Н. Самоучитель Microsoft Access. СПб. 2014. 560 с.
2. Глушков С.В. Ломотько Д.В. Базы данных: Учебный курс. Харьков. «Издательство ACT». М. 2013. 420 с.
3. Карпова Т.С. Базы данных: модели, разработка, реализация. СПб Питер. 2012. 350 с.
4. Крёнке Д. Теория и практика построения баз данных. СПб. Питер. 2013. 360 с.
5. А. Горев, С. Макашарипов, Р. Ахаян. Эффективная работа с СУБД. СПб. Питер. 2015. 704с.
6. Стандарты по библиотечному делу: Сборник / Сост. Захарчук Т.В. Петрова Л.И. Завадовская Т.А. Зусьман О.М. СПб. "Профессия". 2015. 512 с.
7. А. В. Кузин, С. В. Левонисова. Использование СУБД Access. Академия. 2013. 320 с.

1. http://www.intuit.ru/studies/courses/496/352/lecture/8387 [↑](#footnote-ref-1)
2. http://www.informicus.ru/default.aspx?SECTION=6&id=95 [↑](#footnote-ref-2)
3. http://www.askit.ru/custom/vba\_office/m1/01\_02\_basis\_vba.htm [↑](#footnote-ref-3)