

РЕФЕРАТ

Потеренко А.Г. Разработка Интернет-магазина: Исследовательская работа:
ДИТУД – Димитровград, 2007. – 41 с.

Объект исследования – Интернет-магазины.

Цель работы – Разработка Интернет-магазина, работающего на различных платформах. В процессе работы был создан программный продукт, позволяющий в короткие сроки развернуть работающую систему “Интернет-магазин” в сети Интернет.

Полученные результаты: Разработан Интернет-магазин.

Выводы: Разработано решение, позволяющее решать проблемы электронной торговли в сети Интернет.

Область применения: торговые организации, которые хотели бы перевести часть своего бизнеса для работы в глобальных сетях.

					КР-ДИТУД-230105-013-07			
Изм.	Лист	№ докум.	Подпись	Дата	Разработка Интернет-магазина на языке Java	Лит.	Лист	Листов
Разраб.		Потеренко А.Г.						
Провер.		Нестеркин В.Н.					1	41
Реценз.						ВТ-41		
Н. Контр.								
Утверд.								

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1. ПРОГРАММНАЯ ДОКУМЕНТАЦИЯ.....	4
1.1. Обзорная часть.....	4
1.2. Стадии и сроки разработки.....	5
1.3. Функциональная спецификация.....	5
1.4. Нефункциональная спецификация.....	7
1.5. Программные средства реализации.....	9
1.6. Описание программы.....	12
1.7. Описание применяемых алгоритмов.....	20
1.8. Структура базы данных.....	21
1.9. Тестирование.....	25
1.10. Вид рабочей программы.....	35
2. ЭКСПЛУАТАЦИОННАЯ ДОКУМЕНТАЦИЯ.....	36
2.1. Руководство программиста.....	36
ЗАКЛЮЧЕНИЕ.....	40
ИСТОЧНИКИ ИНФОРМАЦИИ.....	41

ПРИЛОЖЕНИЕ 1. Листинги программ.

ПРИЛОЖЕНИЕ 2. Видео-ролики, демонстрирующие работу магазина

ПРИЛОЖЕНИЕ 3. Дополнительная документация прикладного программиста для обращения к функциям и процедурам API

ВВЕДЕНИЕ

Данная курсовая работа представляет собой взаимосвязь таких важных понятий как электронная коммерция и электронный бизнес. Процесс их объединения с современными технологиями является главной задачей, поставленной перед автором.

Электронная коммерция – технология, позволяющая заказчикам оформлять и оплачивать заказы на Web-узле предприятия. Предприятия используют свой Web-узел не только как постоянно обновляемый рекламный проспект, но дают возможность заказчикам совершать сделки на этом Web-узле и могут даже дополнительно предоставлять услуги и поддержку в оперативном режиме. Для этого обычно применяются те или иные формы защищенных транзакций, основанных на использовании технологий, которые будут использоваться при написании данной работы. Это позволяет предприятию проводить свои деловые операции круглый год, 24 часа в сутки, что может привести к расширению торгового оборота, уменьшить затраты на сбыт и обслуживание клиентов, а также повысить качество обслуживания заказчиков.

Электронный бизнес – полная интеграция технологии Internet в экономическую инфраструктуру предприятия. Предприятия применяют технологию Internet во многих областях своей деятельности. Для управления всеми внутренними и внешними процессами применяются сети, а сбыт, обслуживание и реклама осуществляются исключительно через Web. Кроме прочих потенциальных преимуществ, на предприятии ускоряется обмен данными, деловые процессы становятся проще, эффективнее, а производительность возрастает.

Умение управлять процессом внедрения современных технологий в инфраструктуру предприятия является главной задачей программистов, работающих на данном предприятии. Перед ними стоит очень сложная задача, решить которую они обязаны качественно и в относительно короткий срок.

Создать Интернет-магазин для большого предприятия требует очень больших капиталовложений. Современное программное обеспечение стоит очень дорого, поэтому задача программиста – найти золотую середину между выбором используемого программного обеспечения в процессе разработки проекта и созданием успешного проекта в кратчайшие сроки. Автор постарался выбрать именно те программные продукты, эффективность которых доказана уже не одним успешным проектом в мире электронного бизнеса.

При решении поставленной задачи автор попытался создать наиболее удобный в использовании и легкий в установке и настройке программный продукт. Поэтому установить его не составит труда даже неопытному пользователю.

Данная система в основном разрабатывалась для предприятий, у которых в штате присутствуют опытные системные администраторы и администраторы баз данных.

1. ПРОГРАММНАЯ ДОКУМЕНТАЦИЯ

1.1. Обзорная часть

Развернутая постановка задачи

1. Создание информационной системы, реализующей логику Интернет-магазина.
2. Создание удобного интерфейса пользователя через браузер для управления всей системой ИМ в целом.
3. Выбор программных средств для реализации поставленной задачи.
4. Программная логика должна быть реализована для трех пользователей системы: клиента, персонала магазина, администратора магазина.
5. Система должна быть многопользовательской, как того подразумевает название проекта.

Web-доступ подразумевает использование Web-интерфейса для доступа к СУБД. В дальнейшем разработанные модули понадобятся предприятию для полного перевода всей интерфейсной логики на уровень Web.

У автора на выбор предоставляется много технологий программирования и систем управления базами данных. Поэтому второстепенная задача – провести анализ существующих средств и правильно сделать выбор.

Анализ аналогичных разработок

В Интернете можно найти много компаний, которые специализируются на создании разработок, призванные решить поставленную задачу:

1. Компания "WEB-студия ВВ2". Программисты этой фирмы давно себя зарекомендовали с хорошей стороны. Их разработки удовлетворяют современным требованиям построения интерфейса для пользователя. Но есть у них два больших недостатка. Первый – цена (\$2500) за каждый заказ (рекомендуется примерно от 500 до 1000 \$). Второй – они используют технологии создания приложений, о которых заказчик может только догадываться. На мой взгляд, заказчик должен быть уверен в безопасности, масштабируемости разрабатываемой системы и должен заранее знать о том, как конфиденциальные данные будут передаваться по сети (в зашифрованном, незашифрованном виде), как, и где будут храниться данные (в какой базе), на какой технологии будет работать сервер приложений и т.п. Вся эта информация закрыта для заказчика.

2. "1С: Аркадия Интернет-магазин". Хотя фирма 1С зарекомендовала себя с лучшей стороны в области бухгалтерского учета и торговли, для крупномасштабных проектов она непригодна. Для выполнения длительных транзакций используется SQL-версия продукта 1С. Это уже большие накладные расходы на сервер MSSQL. Без него в сети с большим числом пользователей (каким является Интернет) использование данной разработки не имеет смысла. К тому же версия 1С не многоплатформенная, так как она базируется на ОС семейства Windows, как известно – самой небезопасной системы с большим числом ошибок, которые легко эксплуатируют хакеры с помощью эксплоитов.

Первый вариант подходит предприятиям, не имеющих собственного хос-

тинга, соответственно собственной сети с сервером приложений и СУБД.

Второй вариант накладывает ограничение на платформу и зависимость от небезызвестной фирмы Microsoft.

1.2. Стадии и сроки разработки

Предполагаемые сроки разработки колеблются от 2,5 до 3 месяцев в зависимости от решения второстепенных задач, помимо главных задач.

Начало разработки: 1.03.07. Конец разработки: 1.06.07 (и раньше).

Разработка будет проведена в три стадии:

1. Разработка технического задания.
2. Рабочее проектирование и разработка.
3. Внедрение.

На стадии разработки технического задания будет выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования и разработки будут выполнены перечисленные ниже этапы работ:

1. Разработка программы.
2. Разработка программной документации.
3. Испытания программы.

На стадии внедрения будет выполнен этап разработки – подготовка и передача программы.

1.3. Функциональная спецификация

Описания внешней информационной среды

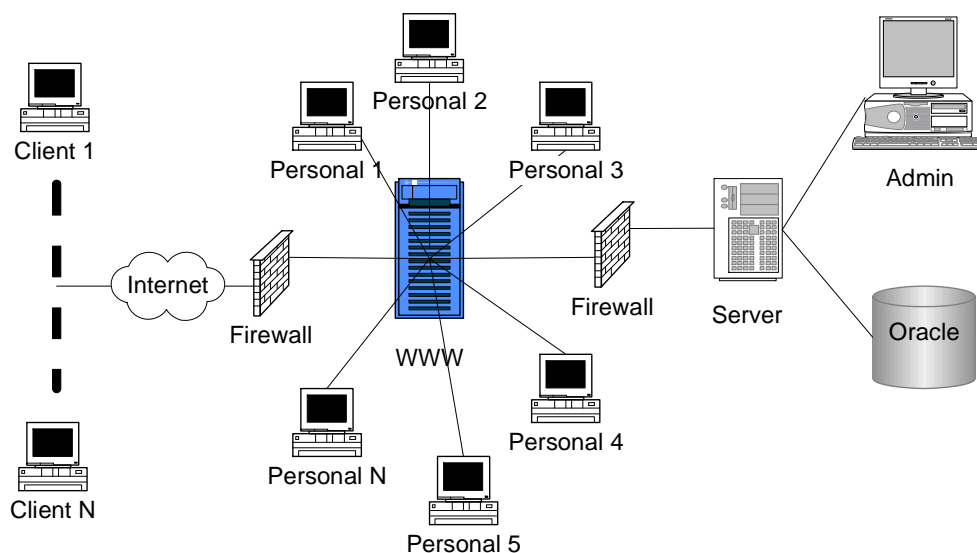


Рис. 1. Концептуальная схема внешней информационной системы

На рисунке 1 на концептуальном уровне определены все используемые каналы ввода и вывода и все информационные объекты, к которым будет применяться разрабатываемое ПС, а также существенные связи между этими информационными объектами.

Администратор работает с системой по архитектуре “Клиент-сервер”, а все остальные (персонал и клиенты) по “Трехуровневой архитектуре”.

Определение функций ПС

Так как разрабатываемая система условно делится на три части, рассмотрим функциональную спецификацию каждой части.

Административная часть будет обладать следующей функциональностью:

1. Регистрировать учетные записи персонала магазина. Администратор должен ввести логин и пароль пользователя, затем программа добавит их в базу данных.

2. Удаление учетных записей персонала магазина. Удаление происходит по введенному логину.

3. Создание секретного симметричного ключа шифрования. Этот ключ в дальнейшем будет нужен для шифрования пароля.

4. Просмотр всех учетных записей персонала. Выводятся сведения об идентификаторе пользователя, его логине, пароле и времени последнего входа.

5. Дополнительно должна быть реализована функция для перекодировки JSP файлов из кодировки Win-1251 в кодировку ISO-8859-1.

6. Авторизованный вход в систему по логину и паролю.

Пользовательская часть будет обладать следующей функциональностью:

1. Регистрация склада. При этом пользователь должен ввести данные о складе: название, страна, область, город, улица, дом.

2. Накладные прихода товаров. При этом пользователь должен выбрать ID склада и ID товара, ввести данные о прибывшем количестве товара.

3. Статистика по товарам и складам. Здесь пользователь может просмотреть данные о: суммарном количестве каждого товара по складам, суммарном количестве товаров на каждом складе, пустующих складах, закончившихся товаров.

4. Регистрировать новую категорию товаров. При этом пользователь должен ввести лишь название категории.

5. Регистрировать нового поставщика товаров. При этом пользователь должен ввести название поставщика.

6. Регистрировать новый способ оплаты заказов. При этом пользователь должен ввести название способа.

7. Регистрировать новый способ доставки заказов. При этом пользователь должен ввести название способа.

8. Регистрировать новый товар на складе. При этом пользователь должен ввести название товара, выбрать идентификатор категории товара, заполнить характеристики товара, выбрать бренд товара, заполнить описание товара, ввести гарантию на товар, выбрать идентификатор поставщика товаров, ввести цену товара, выбрать локальный файл с изображением товара.

9. Редактировать свойства товара. При этом пользователь должен выбрать идентификатор товара и продолжить редактирование.

10. Просмотреть товары по определенным критериям. Критерии включают в себя: идентификатор категории, идентификатор поставщика, бренд, гаран-

тийный период, цена.

11. Просмотреть свойства отдельного товара.
12. Обрабатывать заказы пользователей.
13. Авторизованный вход в систему по логину и паролю.

Клиентская часть будет обладать следующей функциональностью:

1. Авторизованный и неавторизованный режим работы в системе.
2. Регистрация клиента.
3. Поиск товаров по названию.
4. Просмотр товаров.
5. Работа с электронной корзиной: добавление, удаление товаров из корзины, редактирование количества товаров в корзине.
6. Личная страница зарегистрированного клиента с историей заказов, статусом заказов и личными данными.

Описание исключительных ситуаций

В разрабатываемой системе могут возникнуть, по меньшей мере, две исключительные ситуации по вине персонала системы, при которых адекватность данных может не соответствовать действительности:

1. Нельзя выполнять одновременно вставку нового товара и регистрировать новый склад. Может возникнуть ситуация, когда новый товар не сможет быть оформлен на регистрируемом складе. При регистрации нового склада необходимо перевести БД в оффлайн-режим и внести необходимые изменения.

2. При регистрации нового товара возможна ситуация, когда данные о товаре будут занесены в БД, а изображение товара не будет вставлено по причине того, что графического файла может не быть на съемном носителе или сетевом диске. Это может произойти потому, что процедура вставки товара происходит в одной транзакции, а его изображения – в другой. При этом выдается сообщение об ошибке в окне статуса, хотя товар был вставлен. Необходимо проверить выполнение операции и удостовериться, что товар имеет свое изображение.

1.4. Нефункциональная спецификация

Перечень достаточно элементарных свойств, которые требуется обеспечить в разрабатываемом ПС и которые в совокупности образуют приемлемое для пользователя качество ПС:

1. Независимость от устройств.

ПС будет разработано для работы на разнообразном аппаратном и программном обеспечении.

2. Автономность.

ПС будет зависеть от многих компонентов, поэтому оно не будет обладать свойством автономности.

3. Модифицируемость.

ПС будет разработана как система из взаимодействующих модулей (клас-

сов). Жесткая иерархия классов позволяет легко дописывать, изменять компоненты.

4. Устойчивость.

ПС будет обладать свойством устойчивости, и продолжать корректное функционирование, несмотря на ошибочные входные данные от пользователей. Система будет функционировать, но ожидаемые результаты естественно будут неправильными.

5. Эффективность.

Временная эффективность будет зависеть от загруженности серверов баз данных и Web-серверов.

ПС не будет являться эффективным по памяти, так как для высокого быстродействия системы необходимо довольно большой объем оперативной памяти.

Оно также не будет эффективным по устройствам, так как для успешного функционирования необходимо дорогое аппаратное обеспечение.

6. Точность.

ПС будет выдавать 100% правильные результаты своей работы с точки зрения предполагаемого использования. Величина погрешности результатов сводится к нулю, благодаря использованию качественного ПО при разработке и использовании его ПС.

7. Завершенность.

ПС будет обладать всеми необходимыми функциями для решения поставленных задач.

8. Защищенность.

ПС будет обладать данным свойством, так как основные функции системы недоступны пользователям напрямую и выполняются автономно, независимо от действий пользователя.

9. Легкость применения.

ПС будет поставляться с обучающими видео-роликами и документацией по применению.

10. Сопровождаемость.

ПС будет обладать данным свойством, потому что оно поставляется со справкой API-интерфейса всей системы. Другими словами, любой сторонний программист может получить доступ к описанию любой функции и к ее параметрам.

11. Расширяемость.

ПС обладает данным свойством, так как будет базироваться на программном обеспечении, которое, как заверяют его разработчики, имеет способность к использованию большего объема памяти для хранения данных или расширению функциональных возможностей отдельных его компонент.

12. Восстанавливаемость.

ПС обладает данным свойством, так как при некорректном завершении работы все начатые транзакции откатываются, и при следующем запуске система переходит в нормальный режим работы, а последствия краха остаются

лишь в системных журналах.

1.5. Программные средства реализации

Технология программирования

Для программирования для сети Интернет разработано много Web-языков программирования. В числе них наиболее известные С#, PHP, Java.

С# – творение Microsoft. Язык, наряду с технологией ASP .Net позволяет создавать код, генерирующий динамические страницы. Об этой платформе можно рассказывать бесконечно. На мой взгляд, самая удачная разработка этой корпорации. Microsoft .Net – это платформа, включающая серверы, клиенты и сервисы. Она содержит набор приложений, таких, как Visual Studio .Net, Tablet PC, и .Net My Services. Microsoft .Net была спроектирована таким образом, чтобы удовлетворять текущие и будущие требования заказчиков к созданию, внедрению и управлению приложениями. Фундаментальным принципом Microsoft .Net является изменение процесса разработки и внедрения программного обеспечения – в частности, переход от разработки собственных программ к покупке, настройке и интеграции готового программного обеспечения. Microsoft .Net была разработана для интеграции приложений с помощью Web-служб, используя протоколы и форматы, такие, как SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) и UDDI (Universal Description, Discovery, and Integration). Многоязыковая поддержка также является большим плюсом для этой платформы.

Можно сказать только одно про эту платформу. Новизна этой технологии должна зарекомендовать себя временем. Продукт создан одной корпорацией и учитывает не все требования, которые выдвигают множества мелких компаний, поэтому не все стремятся переходить на .NET.

Язык PHP – сравнительно молодой, но в то же время удивительно удобный и гибкий язык для программирования Web. С помощью него можно написать 99% программ, которые обычно требуются в Интернете. Для оставшегося 1% придется использовать С или Perl. Главный недостаток на мой взгляд – язык очень расположен к ошибкам, допускаемых программистами низкого и среднего уровня. Если и разрабатывать на PHP, то необходимо знать в корне все его особенности, иначе приложение просто опасно использовать. Не счесть случаев использования глубоких знаний этого языка в корыстных целях, которые преследуют хакеры. Существуют много реальных примеров взлома Интернет-магазинов, построенных на технологии PHP. Автор не рискнул его использовать в качестве технологии для своего проекта.

Язык Java – творение Sun Microsystems. Отличительная особенность Java – средства безопасности. Как язык, так и сама платформа с самого начала разрабатывались с учетом безопасности. Java-платформа позволяет пользователям загружать по сети непроверенный код и запускать его в безопасной среде, в которой он не сможет нанести какой-либо ущерб. Непроверенный код не сможет заразить базовую систему каким-либо вирусом, не сможет считывать или записывать файлы на жесткий диск и т. п. Одни эти возможности делают

Java уникальной платформой.

В Java любой Java-код – апплет, сервлет, JavaBeans-компонент или все Java-приложение – можно запускать с ограниченными правами доступа, что предотвратит возможность нанесения какого-либо ущерба системе.

Вопросы безопасности, связанные с языком и платформой Java, тщательно рассматривались специалистами во всем мире. На ранних этапах существования Java ошибки в системе безопасности – некоторые из них были достаточно серьезные – не раз обнаруживались, но в дальнейшем были исправлены. Java предоставляет надежные гарантии, связанные с безопасностью, и поэтому случаи, когда находится новая ошибка в системе безопасности, становятся сенсацией. Ни одна другая распространенная платформа не предоставляет таких серьезных гарантий безопасности, как Java. Никто не утверждает, что в будущем не будут обнаружены новые прорехи в системе безопасности Java, но даже если безопасность Java не является совершенной, эта платформа на практике показала себя достаточно надежной в повседневном использовании. Безусловно, она лучше всех других альтернатив.

Технология Java поддерживается и совершенствуется с помощью так называемого процесса Java Community Process (JCP), представляющего собой взаимодействие более 400 компаний, организаций и частных лиц в целях создания платформы для сервисов и приложений, которая может работать на системах любого типа.

Java приложения могут выполняться на любых операционных системах: системах уровня предприятия, таких, как Unix, Linux, OS/390, Windows 2000, или HP-UX; домашних операционных системах, таких как Mac OS, Windows или Linux; а также операционных системах для мобильных устройств, таких как Palm OS или Symbian EPOC

Можно также отметить, что технология Java является открытой и построенная на внутриотраслевых стандартах для программного обеспечения. Любой желающий может загрузить и изучать код Java платформы.

Итак, основываясь на вышеизложенном высказывании, можно сказать, что разработка проекта именно на Java имеет много перспектив в будущем.

Система управления базой данных

Построение современных информационных систем сегодня напрямую связано с реляционными и объектно-ориентированными СУБД. Данные системы в последнее время утвердились как основные средства для обработки данных в информационных системах различного масштаба от больших приложений обработки транзакций в банковских системах до персональных систем на PC. В настоящее время существует множество систем управления базами данных (СУБД) и других программ выполняющих сходные функции. Инструментальные средства Oracle Database 9i – одни из лучших и наиболее мощных имеющихся инструментов разработки профессионального класса.

В современном мире существует как минимум три направления для выбора СУБД для своих проектов. Первое направление – Microsoft со своим

MSSQL Server. Программисты этой СУБД неплохо выполнили свою задачу. Главный недостаток – данная СУБД работает только на платформе Windows. Второе направление – OpenSource СУБД, такие как MySQL и Firebird. Хотя первая и предназначена для работы в Web, но не имеет поддержку длительных транзакций. Вторая вовсе предназначена для малых систем управления. Третье направление – СУБД Oracle. Перенесена под все существующие платформы. Поддерживает сверхбольшие БД (свыше 1 ТБ). Мощная и удобная система управления транзакциями. Встроенный язык PL/SQL со своими модулями DBMS позволяет сделать почти все для надежной и эффективной работы. Поддержка хранимых процедур на языке Pro*C, Pro*C++, Java. Существуют множество драйверов под различные языки для взаимодействия с БД, таких как Perl, PHP, Java. Пакет Oracle Database 9i, наделенный самым развитым набором функций для работы с языком Java и доступа к данным через Интернет, является системой оптимизации одновременного доступа. Единственным недостатком данной СУБД является сложность администрирования, однако все затраты на ее внедрение и освоение в последствии окупятся эффективной и надежной работой. По уровню производительности при работе в Web-среде под Linux Oracle занимает почетное второе место после OpenSource СУБД MySQL, при этом значительно превосходя все другие СУБД (в том числе и СУБД MySQL) по надежности и безопасности.

Для интеграции Oracle Database 9i и различных Интернет-технологий (в первую очередь всемирной паутины) и предназначена данная курсовая работа. При использовании данной разработки взаимодействие пользователей с базой данных Oracle Database 9i можно организовать, совместив средства этой СУБД и стандартные возможности Web-интерфейса. Web-приложения на основе Oracle могут входить в структуру внутрикорпоративных процессов управления, так и представлять собой Интернет-магазины.

Данная разработка может использоваться в любом учреждении, предприятии или организации, которая использует систему Oracle Database 9i для автоматизации своей деятельности и испытывает необходимость в электронной коммерции.

Выбор Web-сервера

Так как условиями задачи предопределено использование Java-технологии для решения задачи, значит, нам необходимо использовать тот сервер, который поддерживал бы элементы J2EE (напомним, что таковыми элементами являются JSP и сервлеты). На выбор нам предоставляется множество различных Web-серверов. В том числе и от фирмы Oracle, но, на мой взгляд, для нашей задачи подошли бы два из всего множества. Это Web-сервер Blazix и Jakarta Tomcat Servlet/JSP Container. Первый представляет собой быстродействующий сервер приложений Java. Предельно прост в настройке и администрировании. Бесплатен для некоммерческого использования. Второй имеет смысл использовать в том случае, когда будет необходимость использовать наш проект как коммерческий. Итак, выберем Blazix версии 1.2.5.

Выбор инструментов для разработки

Для разработки будут применяться следующие системы программирования:

1. NetBeans IDE 5.0. Распространяется по лицензии Sun Public License. Эта среда предназначена для создания проектов Java. Шаблоны кода позволяют быстро начать программировать. Встроенный отладчик, удобный редактор кода с подсветкой синтаксиса, встроенная справочная система по API. Программа работает на платформах Windows и Linux. Среда является бесплатным продуктом для коммерческих проектов.

2. Htmllpad Fisherman 1.9. Бесплатная среда для создания JSP и HTML страниц. Удобный генератор HTML кода позволяет быстро создать интерфейс;

3. PL/SQL Developer 6.0.6.947. Первокласный инструмент для создания кода PLSQL. Первокласный отладчик кода, браузер всех системных объектов СУБД Oracle, генератор всех известных конструкций встроенного языка и т.п. Продукт распространяется на коммерческой основе.

1.6. Описание программы

Уровень СУБД

Для создания логики на уровне базы данных придется писать модули на встроенном процедурном языке под названием PLSQL.

Для функционирования системы были написаны две процедуры и пятнадцать функций. Рассмотрим каждую поподробнее.

Процедура GET_INFO_CLIENT_GOODS

Модуль создан специально для показа пользователю статуса магазина. Если клиентов будет зарегистрировано много, то приоритет будет высоким и клиент обязательно совершит покупки. Количество товаров так же будет внушать доверие.

Данный модуль возвращает количество наименований товаров и количество зарегистрированных клиентов.

Процедура UPDATE_USER_LAST

Модуль обновляет дату и время последнего входа рабочего для совершения тех или иных операций с базой данных.

Функция REGISTRATION

С помощью данного модуля можно зарегистрировать клиента в магазине. Если операция происходит успешно, то возвращается идентификатор зарегистрированного пользователя, иначе – ноль.

Функция AUTH_USER

С помощью данного модуля клиент авторизуется в магазине. Если ав-

торизация проходит успешно, возвращается идентификатор клиента. В данной функции происходит сравнение зашифрованной строки со строкой, хранящейся в базе данных.

Функция REGISTRATION_NEW_GOODS

С помощью данного модуля можно зарегистрировать новый товар в базе данных. Если регистрация товара проходит успешно, возвращается указатель на поле, хранящее изображение данного товара. Это необходимо для того, чтобы занести файл с изображением в другую таблицу, а в текущей поставить ссылку на данное поле.

Функция FIND_STRING

С помощью данного модуля можно произвести контекстный поиск по товарам. Возвращаются идентификаторы на найденные товары. То есть возвращается строка вида $\langle ID_1; \dots; ID_N \rangle$.

Функция GETLASTTIME

С помощью данного модуля можно получить время последнего захода рабочего в магазин.

Функция AUTH

С помощью данного модуля персонал магазина авторизуется в магазине. В данной функции происходит сравнение зашифрованной строки со строкой, хранящейся в базе данных.

Функция ADD_WAREHOUSES

С помощью данного модуля можно зарегистрировать новый склад и все товары, которые есть в магазине автоматически зарегистрировать на этом складе. Дело в том, что каждый товар должен быть зарегистрирован на всех складах, которые есть у предприятия. Количество каждого товара будет равно 0 на этом складе.

Функция ADD_DELIVERY

С помощью данного модуля можно добавить новый способ доставки заказов.

Функция ADD_PAYMENT

С помощью данного модуля можно добавить новый способ оплаты заказов.

Функция ADD_SUPPLIER

С помощью данного модуля можно добавить нового поставщика товаров.

Функция ADD_CATEGORY

С помощью данного модуля можно добавить новую категорию товаров.

Функция ISSUE_WAYBILL

С помощью данного модуля можно оформить накладную прихода на товар.

Функция CANCEL_REG_PERS_SHOP

С помощью данного модуля администратор может удалить учетную запись персонала магазина.

Функция REGISTRATION_PERSONNEL_SHOP

С помощью данного модуля администратор может регистрировать персонал магазина.

Функция REGISTRATION_ORDER

С помощью данного модуля клиент может оформить заказ. В случае успеха возвращается 0. Если какого-либо товара из заказа не хватает на складе, то возвращается идентификатор этого товара. Данный модуль выполняет довольно большую работу. В его задачи входит изменять данные о количестве товаров на складах после вычета количества, приходящегося на заказ, внесение характеристик заказа в базу данных. Также этот модуль автоматизирует комплектацию заказов, то есть показывает рабочим, сколько и с какого склада необходимо выбрать товаров, чтобы выполнить заказ.

Уровень Web

Для решения главной задачи по программированию логики на промежуточном уровне были созданы семнадцать классов, каждый из которых выполнял свою работу. Все модули упакованы в jar-архив Shop.jar.

Модуль ShowImage.java

Приложение представляет собой апплет. Логика его работы заключается в том, чтобы соединиться с Web-сервером, а именно получить от сервлета бинарные данные, представляющие собой изображение товара.

Апплет прорисовывает изображение в определенной области браузера. Ему на вход подаются три параметра: x и y координаты области, в которой будет происходить рисование и идентификатор изображения в БД. Апплет должен находиться в корне Web-директории, т.е. чтобы переменная CodeBase была равна <http://myhost/>.

Модуль ImageServlet.java

Сервлет для извлечения BLOB-рисунков из БД. Главный метод обрабатывает Get-запросы, поступившие от клиента.

Для соединения использует минимальный набор прав пользователя SHORMIN.

Модуль Admin.java

Модуль для администратора БД. Основное назначение: регистрировать рабочий персонал для работы в магазине. Дополнительное назначение: вызов методов преобразования файлов *.jsp в ISO-8859-1. Программа предоставляет пять операций:

- 1 – Добавить пользователя
- 2 – Удалить пользователя
- 3 – Создать симметричный секретный ключ
- 4 – Показать всех пользователей
- 5 – Преобразование файлов

Предоставляет консольный интерфейс.

Для работы администратор должен ввести логин и пароль, затем произойдет соединение с базой данных.

Модуль CodingISO.java

Модуль кодирования файлов из Win-1251 в ISO-8859-1. Работает как со строками, так и с файлами. В файлах находит буквы кириллицы и преобразует их в ISO кодировку. Один символ кириллицы занимает 7 байт в ISO-кодировке. Например, первая буква "А" имеет формат "А".

Модуль ShopResultSet.java

Модуль предназначен для преобразования набора данных ResultSet в тип List для доступа к строкам и столбцам как к элементам списка. Данные из набора ResultSet представляют собой таблицу $M[i,j]$. Конечный список будет иметь вид $L[1,1]...L[1,i]...L[j,1]...L[j,i]$. Таким образом, матрица вытягивается в прямую линию.

Модуль FormHTML.java

Модуль для генерации динамических JSP-страниц для раздела персонала магазина.

Модуль GoodsWarehousesClass.java

Модуль, методы которого позволяют выводить статистику на странице GoodsWarehouses.jsp. Статистические данные уже не раз упоминались в данной курсовой работе.

Модуль ShopCipher.java

Модуль предназначен для шифрации паролей пользователей.

Модуль GoodsEdit.java

Модуль, методы которого предоставляют интерфейс для работы с пунктом с пунктом "Редактирование товаров".

Модуль ShowGoods.java

Модуль, методы которого предоставляют интерфейс для работы с пунктом с пунктом "Просмотр товаров".

Модуль Orders.java

Модуль предназначен для обработки заказов персоналом магазина. Предоставляет функции для работы страниц OrderMain.jsp, OrderOne.jsp и OrderTwo.jsp.

Модуль StorekeeperBean.java

Модуль, представляющий собой главный JavaBean для персонала магазина. Именно его используют все JSP-страницы для выполнения логики.

Работает с правами пользователя SHOP. В тексте этого класса находится пароль в открытом виде. Если хакеру удастся скопировать этот файл, то конфиденциальность данных в базе данных будет обречена на нет.

Модуль HtmlPage.java

Модуль, методы которого генерируют html-код стандартных элементов браузера.

Модуль Basket.java

Модуль, реализующий концепцию виртуальной корзины при посещении магазина. Каждый элемент корзины абстрактно представляет собой запись, состоящую из 4х элементов: идентификатор товара, количество, цена и название товара. Над элементами корзины можно выполнять основные функции: добавлять товар, удалять товар, редактировать количество товара при оформлении заказа. Также возможно просматривать содержимое корзины.

Модуль PLSQLStored.java

Модуль, методы которого представляют собой оболочку для работы с хранимыми объектами БД персонала магазина. Работает через интерфейс JDBC.

Модуль PLSQLStoredUser.java

Модуль, методы которого вызывают хранимые объекты из БД. Процедуры и функции этого класса предназначены для клиентской части.

Модуль ClientBean.java

Модуль предназначен для работы клиента с магазином. Представляет собой "обертку" для других классов, через которые происходит взаимодействие с базой данных. Для отладки выполненных пользователем операций в консоли выводятся сообщения о возможных ошибках. Администратор Web-сервера может просмотреть данные ошибки и своевременно на них отреагировать. Максимальное количество товаров, которое может заказывать пользователь, ограничено ста единицами. Пользователь может заказывать товары только в том случае, если он авторизован.

Клиент имеет возможность работать в неавторизованном и авторизованном режиме. В первом случае он может регистрироваться, искать товары по контекстному поиску, просматривать товары. Во втором случае он должен авторизоваться или зарегистрироваться. После этого ему будут доступны: добавление, удаление товаров из корзины, редактирование количества товаров в корзине. Переход на личную страницу, где ему будет выведена история его заказов, статус заказов, а также его личные данные.

Web-интерфейс

Для клиента был создан Web-интерфейс. Его структура изображена на рисунке 1:

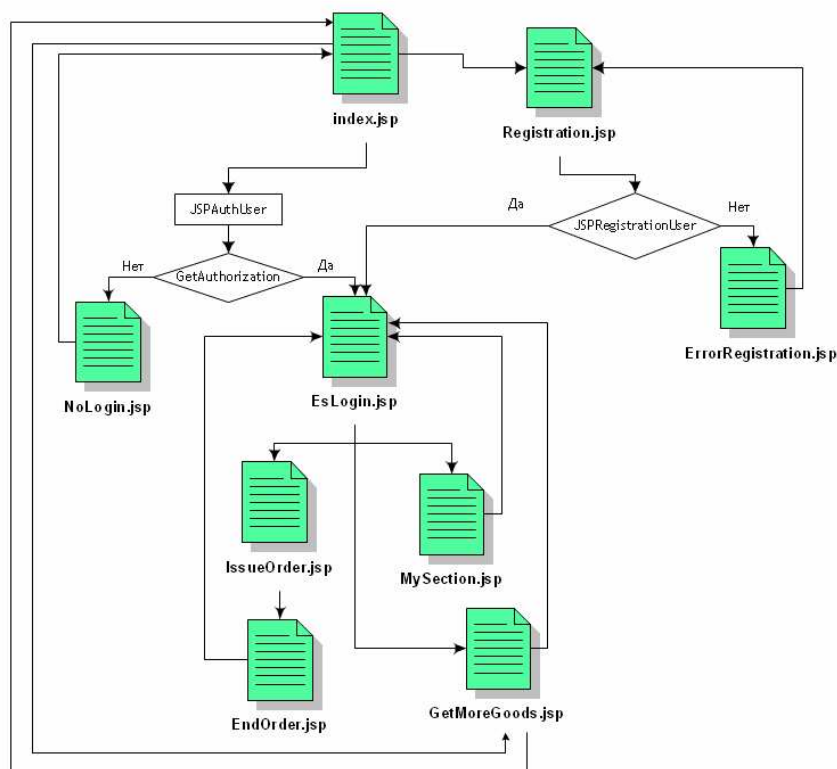


Рис. 1. Раздел сайта для клиента.

Рассмотрим каждую страницу в отдельности.

- index.jsp – главная страница, с которой пользователь начинает свою работу;

- Registration.jsp – страница, на которой пользователь может зарегистрироваться. Если регистрация проходит удачно, то клиент сразу же становится авторизованным пользователем и может совершать покупки, иначе ему повторно предоставляется возможность зарегистрироваться;
- EsLogin.jsp – страница, на которую переходит пользователь в случае успешной авторизации;
- NoLogin.jsp – страница, на которую переходит пользователь в случае неуспешной авторизации;
- ErrorRegistration.jsp – страница, на которую переходит пользователь в случае ошибки регистрации;
- IssueOrder.jsp – страница, на которой клиент может оформить заказ;
- EndOrder.jsp – страница, на которой клиент может просмотреть статус выполнения заказа (диагностическое сообщение);
- MySection.jsp – персональная страница пользователя;
- GetMoreGoods.jsp – страница, на которой пользователь сможет просмотреть более подробное описание товара.

В процессе решения поставленной задачи был реализован интерфейс для персонала магазина. По функциональности он гораздо шире и сложнее. Его структура изображена на рисунке 2:

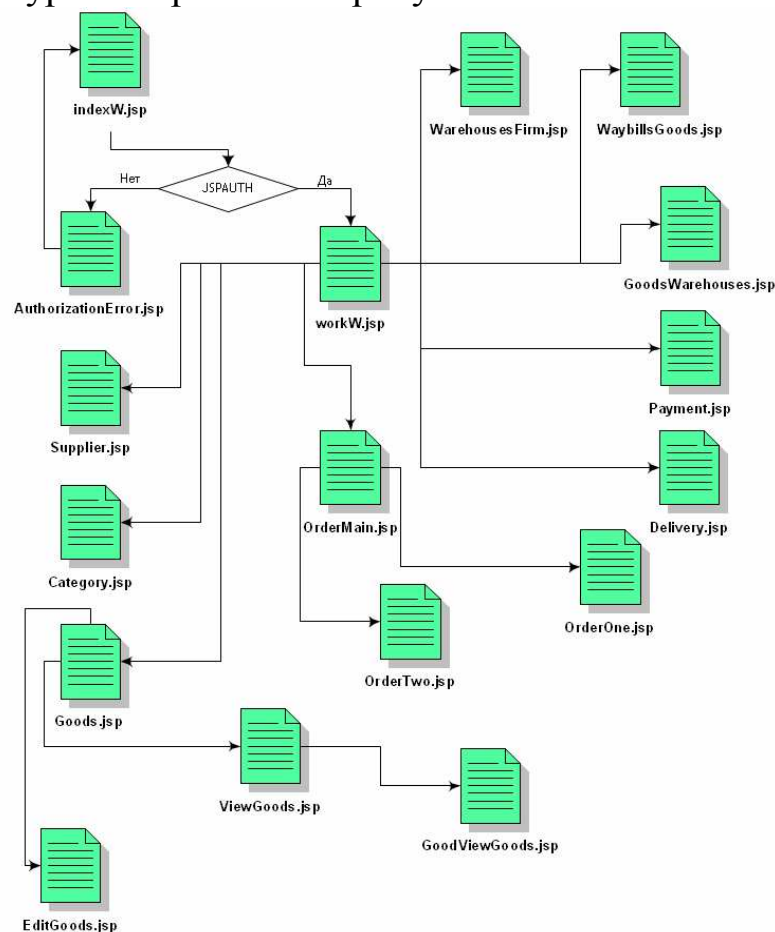


Рис. 2. Раздел сайта для персонала магазина.

Рассмотрим каждую страницу в отдельности.

- indexW.jsp – главная страница, с которой пользователь может авторизоваться;
- AuthorizationError.jsp – страница, на которую попадает пользователь в случае неуспешной авторизации;
- workW.jsp – страница, с которой пользователь начинает свою работу;
- WarehousesFirm.jsp – страница, на которой пользователь может работать со складами фирмы;
- WaybillsGoods.jsp – страница, на которой пользователь может работать с накладными прихода товаров;
- GoodsWarehouses.jsp – страница, на которой пользователь может просмотреть всю статистику по товарам и складам;
- Payment.jsp – страница, на которой пользователь может работать со способами оплаты заказов;
- Delivery.jsp – страница, на которой пользователь может работать со способами доставки заказов;
- Supplier.jsp – страница, на которой пользователь может работать с поставщиками товаров;
- Category.jsp – страница, на которой пользователь может работать с категориями заказов;
- Goods.jsp – страница, на которой пользователь может работать с товарами, то есть добавлять товары в базу данных;
- EditGoods.jsp – страница, на которой пользователь может редактировать свойства товаров;
- ViewGoods.jsp – страница, на которой пользователь может вести поиск товаров по определенным критериям;
- GoodViewGoods.jsp – страница, на которой пользователь просмотреть свойства отдельного товара;
- OrderMain.jsp – страница, на которой пользователь может выбирать накопившиеся заказы на обработку;
- OrderOne.jsp – страница, на которой пользователь может просмотреть свойства заказа и выполнить его, укомплектовав нужные товары со склада;
- OrderTwo.jsp – страница, на которой пользователь может завершить заказ, удостоверившись, что от клиента поступила оплата.

Интерфейс командной строки

Администратор магазина имеет в своем распоряжении консольную программу, написанную специально для выполнения его обязанностей.

1.7. Описание применяемых алгоритмов

Никто не сомневается, что хранить номера кредитных карт и пароли клиентов в базе данных в открытом виде нельзя. Доступ к БД могут иметь многие пользователи, в частности персонал магазина, работающие на предприятии. Не все так бескорыстны. Поэтому эта мера является вторым шагом по обеспечению приватности данных пользователей в системе.

Отсюда вытекает следствие – программа должна иметь возможность для шифрования данных.

Автор остановил свой выбор на блочно-симметричном способе шифрования информации под названием «DES-EDE». Поставщик алгоритма DES: National Institute of Standards and Technology (NIST). DES – это 64-битный алгоритм с эффективной длиной ключа в 56-bits (хотя часто говорят о 8 байтах, но старший бит в байте не используется). DES-EDE – тройной DES. Алгоритм шифрования состоит в следующем: исходный текст зашифровывается DES с ключом K_1 , результат расшифровывается DES с ключом K_2 , а этот результат опять зашифровывается DES с ключом K_3 . Итого, эффективная длина ключа получается 168 бит. Действительная длина ключа все равно будет 192 бита или 24 байта.

Использование данного способа шифрования избавляет нас от юридической ответственности, так как данный алгоритм является на данный момент открытым.

1.8. Структура базы данных

Для хранения информации используется база данных под управлением Database Oracle 9i. База данных имеет следующую структуру:

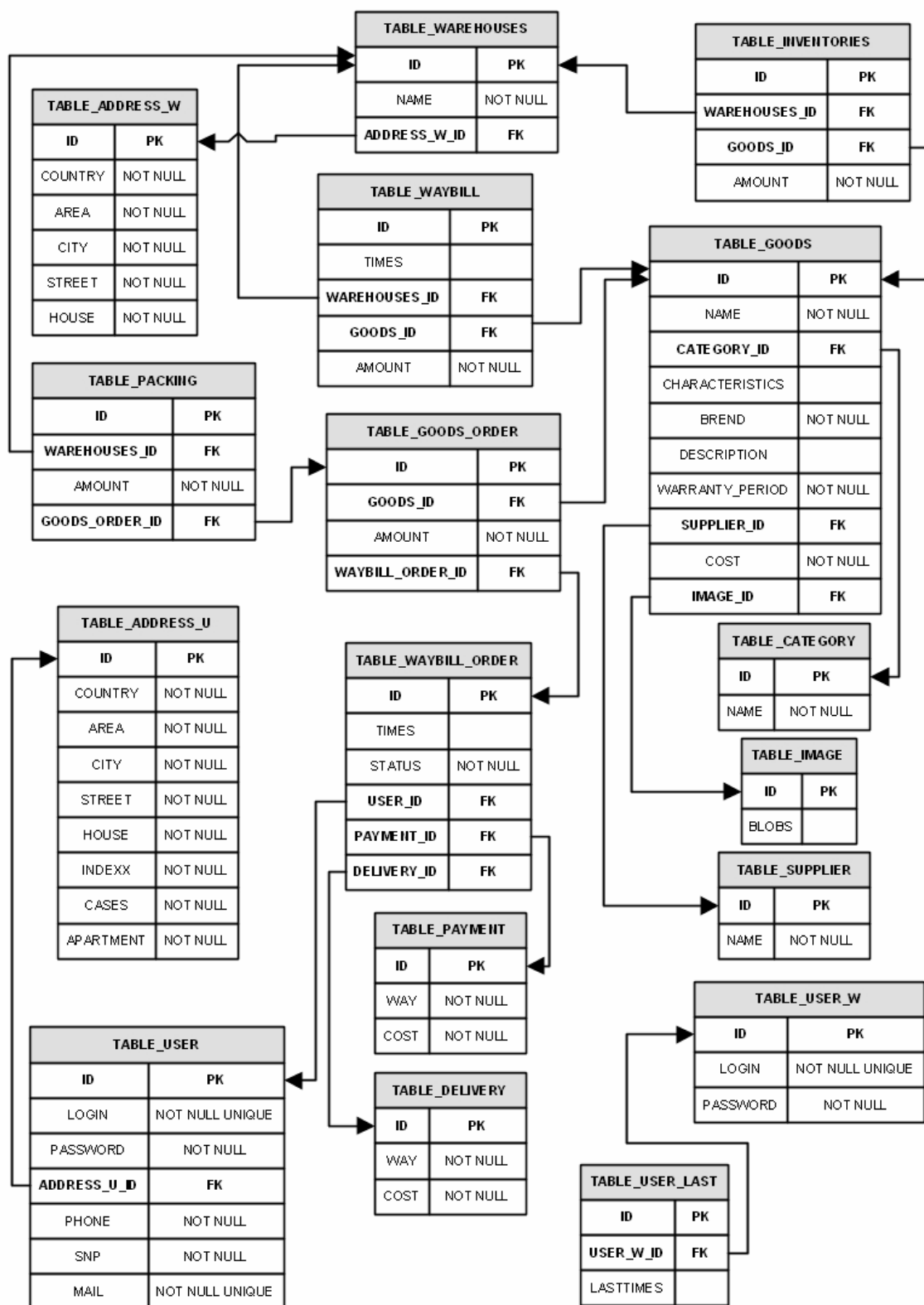


Рис. 3. Структура таблиц базы данных.

ID	LOGIN	PASSWORD
1	USER1	82FD8614A577C46F1A02040E1E45CD64

Таблица TABLE_USER_LAST предназначена для хранения времени о последнем входе рабочего. Данная таблица в первую очередь предназначена для того, чтобы администратор мог проверять рабочих магазина, выполняют ли они свои функции в отведенное для них время. Также возможно отследить несанкционированный доступ со стороны посторонних. Если один из рабочих магазина не работал в то время, когда указано в таблице, значит, кто-то вошел под его учетной записью через Web-интерфейс и администратор должен срочно сменить пароль для этого рабочего. Записи в таблице выглядят следующим образом:

ID	USER_W_ID	LASTTIMES
1	1	08.02.2007 12:35

Таблица TABLE_ADDRESS_U предназначена для хранения сведений об адресе клиентов для доставки товаров. Записи в таблице выглядят следующим образом:

ID	INDEXX	COUNTRY	AREA	CITY	STREET	HOUSE	CASES	APARTMENT
1	56465	Russia	Samarskaya	Samara	Promishler	190	0	36
2	56465	Russia	Ulyanovckaya	Ulyanovck	Sovetskaya	285	0	88
3	56465	Russia	Tomskaya	Tomsk	Kalguskaya	100	0	96
4	453454	Р о М о	М о	М о	М о	4545	0	18

Таблица TABLE_USER предназначена для хранения сведений о личных данных клиента. Записи в таблице выглядят следующим образом:

ID	LOGIN	PASSWORD	MAIL	ADDRESS U ID	PHONE	SNP
1	aslan	79FF3A8ADE43678	aslan@google.com	1	31525398	Anrey Genadevich Mohov
2	laden	F8D12C06B5A4BC	laden@google.com	2	51525394	Mihail Genadevich Giguin
3	liwer	39F89EF83A78C46	liwer@google.com	3	37825394	Sergey Genadevich Gukov
4	RASMUS	D5CCDE20F5EF2F	RASMUS@aslan.com	4	1 231 F+09	Г:а:л:и:

Таблица TABLE_WAYBILL_ORDER предназначена для хранения сведений о заказах клиентов. Записи в таблице выглядят следующим образом:

ID	TIMES	STATUS	USER_ID	PAYMENT_ID	DELIVERY_ID
1	08.02.2007 13:09	0	1	1	1

Таблица TABLE_PAYMENT предназначена для хранения сведений о способах оплаты заказов. Записи в таблице выглядят так:

ID	WAY	COST
1	Cash-to the courier	150,78
2	Cash on delivery	50,17
3	The advance payment	0
4	Credit card	0
5	Electronic money	0

КР-ДИТУД-230105-013-07

Таблица TABLE_DELIVERY предназначена для хранения сведений о способах доставки заказов. Записи в таблице выглядят следующим образом:

ID	WAY	COST
1	The courier	78,5
2	Packet by rail in territory of Russia	59,22
3	Packet by air mail in territory of Russia	65
4	The custom-made parcel post by air mail in territory of other states	89,5

Таблица 7.

Таблица TABLE_GOODS_ORDER предназначена для хранения сведений о том, сколько каждого товара в заказе. Записи в таблице выглядят следующим образом:

ID	GOODS_ID	AMOUNT	WAYBILL_ORDER_ID
1	1	30	1
2	2	15	1

Таблица 8.

Таблица TABLE_PACKING предназначена для хранения сведений о том, сколько каждого товара с какого склада необходимо взять, чтобы выполнить заказ. Записи в таблице выглядят следующим образом:

ID	WAREHOUSES_ID	AMOUNT	GOODS_ORDER_ID
1	4	0	1
2	3	7	1
3	1	22	1
4	2	1	1
5	1	7	2
6	2	3	2
7	4	0	2
8	3	5	2

Таблица 9.

Таблица TABLE_INVENTORIES предназначена для хранения сведений о том, сколько каждого товара находится на каждом складе в текущий момент. Записи в таблице выглядят следующим образом:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	8
4	3	2	40
5	1	2	0
6	2	2	0
7	4	1	0
8	4	2	0
9	3	3	0
10	4	3	0
11	1	3	0
12	2	3	0
13	3	4	0
14	4	4	0
15	1	4	0
16	2	4	0
17	3	5	0
18	4	5	0
19	1	5	0
20	2	5	0

Таблица 10.

Таблица TABLE_WAYBILL предназначена для хранения сведений о накладных прибытия товаров. Записи в таблице выглядят следующим образом:

ID	TIMES	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	08.02.2007 12:04	1	1	5
2	08.02.2007 12:04	1	2	7
3	08.02.2007 12:04	2	1	9
4	08.02.2007 12:04	2	2	3
5	08.02.2007 12:04	3	1	7
6	08.02.2007 12:04	1	1	17
7	08.02.2007 12:04	3	2	45

Таблица 11.

Таблица TABLE_WAREHOUSES предназначена для хранения сведений о накладных прибытия товаров. Записи в таблице выглядят следующим образом:

ID	NAME	ADDRESS_W_ID
1	Warehouse #1	1
2	Warehouse #2	2
3	Warehouse #3	3
4	Warehouse #4	4

Таблица 12.

Таблица TABLE_ADDRESS_W предназначена для хранения сведений об адресах складов предприятия. Записи в таблице выглядят следующим образом:

ID	COUNTRY	AREA	CITY	STREET	HOUSE
1	Russia	The Samara area	Samara	Kievskaya	1001
2	Russia	The Samara area	Samara	Kievskaya	1002
3	Russia	The Samara area	Samara	Kievskaya	1003
4	Russia	The Samara area	Samara	Kievskaya	1004

Таблица 13.

Таблица TABLE_GOODS предназначена для хранения сведений о товарах. Записи в таблице выглядят следующим образом:

ID	NAME	CATEGORY_ID	CHARACTERISTICS	BREND	DESCRIPTION	WARRANTY_PERIOD	SUPPLIER_ID	COST	IMAGE_ID
1	Digital IXUS 700	1	<CLOB>	Canon	<CLOB>	12	1	12446.64	1
2	EXILIM EX-Z750	1	<CLOB>	Casio	<CLOB>	12	2	13633.29	2
3	FinePix F10	1	<CLOB>	FUJIFILM	<CLOB>	12	3	10257.93	3
4	Pentium 4 550	6	<CLOB>	Intel	<CLOB>	12	2	7647.3	4
5	Athlon64 3000+ S939	6	<CLOB>	AMD	<CLOB>	12	4	3559.95	5

Таблица 14.

Таблица TABLE_CATEGORY предназначена для хранения сведений о категориях товаров. Записи в таблице выглядят следующим образом:

ID	NAME
1	Digital chambers
2	Pocket personal computers
3	Cellular telephones
4	Flash-memory
5	Videocards
6	Processors
7	Winchesters
8	DVD-drives

Таблица 15.

Таблица TABLE_SUPPLIER предназначена для хранения сведений о поставщиках товаров. Записи в таблице выглядят следующим образом:

ID	NAME
1	Eqaulins
2	Minisota
3	Drivston
4	Jindison
5	Aston

Таблица 16.

Таблица TABLE_IMAGE предназначена для хранения изображений товаров.

1.9. Тестирование

Тестирование ПС – это процесс выполнения его программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ.

Итак, опишем данные, которые были загружены в БД.

Способы оплаты

1. Наличными – курьеру (Cash-to the courier), при получении товара.
2. Наложный платеж (Cash on delivery). Оплата происходит на почтовом отделении при получении заказа.
3. Предварительная оплата (The advance payment). Банковским переводом в течение 7-ми дней с момента оформления заказа. Реквизиты для оплаты должны быть оставлены на сайте.
4. Кредитная карта (Credit card). Магазин принимает оплату через платежные системы. Оплата происходит через сервер платежной системы.
5. Электронные деньги (Electronic money). Оплаты через систему WebMoney. Номер кошелька должен быть оставлен на сайте. При осуществлении перевода необходимо указать в примечании номер заказа клиента. Этот номер клиент может узнать в разделе сайта "Мой раздел". После того как перевод был сделан, необходимо сообщить об этом в письме администратору магазина.

Способ доставки

1. Курьером (при оплате наличными курьеру).
2. Ценная бандероль ЖТ на территории России (ЖТ – железнодорожный транспорт).
3. Ценная бандероль АП на территории России (АП - авиапочта).
4. Заказная бандероль АП на территории других государств.

Категории товаров

1. Цифровые камеры (Digital chambers).
2. Карманные персональные компьютеры (Pocket personal computers).
3. Сотовые телефоны (Cellular telephones).
4. Флэш-память (Flash-memory).
5. Видеокарты (Videocards).
6. Процессоры (Processors).
7. Винчестеры (Winchesters).
8. DVD-приводы (DVD-drives).

Поставщики товаров

1. Фирма-поставщик "Eqaulins".
2. Фирма-поставщик "Minisota".
3. Фирма-поставщик "Drivston".
4. Фирма-поставщик "Jindison".

Клиенты

1. Логин - aslan.
Пароль - ?_dorman_? (79FF3A8ADE43676B973B9DF5EDC9EBAB)
Почта - aslan@google.com
Телефон - 31525398
ФИО - Андрей Генадьевич Мохов (Anrey Genadevich Mohov)
Индекс - 56465
Страна - Россия (Russia)
Область - Самарская (Samarskaya)
Город - Самара (Samara)
Улица - Промышленная (Promishlennaya)
Дом - 190
Корпус - 0
Квартира- 36
2. Логин - laden.
Пароль - vilnus+-\$ (F8D12C06B5A4BC29F2C5B22785D3EF9D)
Почта - laden@google.com
Телефон - 51525394
ФИО - Михаил Генадьевич Жигулин (Mihail Genadevich Gigulin)
Индекс - 56465
Страна - Россия (Russia)
Область - Ульяновская (Ulyanovskaya)
Город - Ульяновск (Ulyanovsk)
Улица - Советская (Sovetskaya)
Дом - 285
Корпус - 0
Квартира - 88
3. Логин - liwer.
Пароль - zulcul__\$ (39F89EF83A78C46BF2C5B22785D3EF9D)
Почта - liwer@google.com
Телефон - 37825394
ФИО - Сергей Генадьевич Жуков (Sergey Genadevich Gukov)
Индекс - 56465
Страна - Россия (Russia)
Область - Томская (Tomskaya)

Город - Томск (Tomsk)
Улица - Калужская (Kalugskaya)
Дом - 100
Корпус - 0
Квартира- 96

Товары

Курс доллара: 26,37 руб. Цена товара дается в рублях.

1. Название: Digital IXUS 700

Категория: Цифровые камеры

Характеристики: 7,1 Мр 3072x2304; 2,0" 118000 p; SD (32 Mb), MMC; JPEG, AVI; USB 2.0.

Бренд: Canon

Описание: Выглядит Canon Digital IXUS 700 стильно благодаря сочетанию серого матового корпуса и блестящих обода объектива и выступа на корпусе. Фотоаппарат довольно тяжелый.

Гарантийный период (мес.): 12

Поставщик: Eqaulins

Цена: 12446.64

2. Название: EXILIM EX-Z750

Категория: Цифровые камеры

Характеристики: 7,1 Мр, 3072x2304; SD/MMC, 8,3 Mb; 2,5., 115200 p.; JPEG, AVI (MPEG4 640x480), WAV; USB 2.0; TV OUT

Бренд: Casio

Описание: Когда берешь камеру в руки, то, несмотря на компактные размеры, сразу понимаешь, что речь идет не просто о дорогой игрушке. Корпус сделан очень хорошо.

Гарантийный период (мес.): 12

Поставщик: Minisota

Цена: 13633.29

3. Название: FinePix F10

Категория: Цифровые камеры

Характеристики: 6,1 Мр, 2848x2136; xD, 16 Mb; 2,5", 115000 p.; JPEG, AVI (640x480), WAV; USB 2.0, A/V; EXIF print

Бренд: FUJIFILM

Описание: Камера имеет цельнометаллический корпус, стильный дизайн и удобную форму. На большом 2,5. TFT-дисплее специальное покрытие, защищающее его от царапин.

Гарантийный период (мес.): 12

Поставщик: Drivston

Цена: 10257.93

4. Название: Pentium 4 550

Категория: Процессоры

Характеристики: 3400 MGc, Prescott, LGA775, 0.09, FSB 200 MGc, L2 1024 Kb, HyperThreading, MMX, SSE, SSE2, SSE3

Бренд: Intel

Описание: Процессор имеет великолепно сочетание цены и производительности. 3.4 ГГц явно стоят своих денег.

Гарантийный период (мес.): 12

Поставщик: Minisota

Цена: 7647.3

5. Название: Athlon64 3000+ S939

Категория: Процессоры

Характеристики: 1800 MGc, Winchester, Socket 939, 0.09, FSB 200, L2 512 Kb, К: 9, 64bt, HyperThreading, MMX, SSE, SSE2, 3DNow

Бренд: AMD

Описание: В сочетании с производительностью получается, в принципе, довольно заманчивый коэффициент для сборки домашней системы.

Гарантийный период (мес.): 12

Поставщик: Jindison

Цена: 3559.95

Тестирование функций SHOP.ADD_PAYMENT,
SHOP.ADD_DELIVERY, SHOP.ADD_CATEGORY,
SHOP.ADD_SUPPLIER

После выполнения кода:

```
DECLARE
  i NUMBER;
BEGIN
  -----#1-----
  i:=SHOP.ADD_PAYMENT('Cash-to the courier',150.78);SHOP.ShowStatusOperation(i,1,1);
  i:=SHOP.ADD_PAYMENT('Cash on delivery',50.17);SHOP.ShowStatusOperation(i,1,2);
  i:=SHOP.ADD_PAYMENT('The advance payment',0);SHOP.ShowStatusOperation(i,1,3);
  i:=SHOP.ADD_PAYMENT('Credit card',0);SHOP.ShowStatusOperation(i,1,4);
  i:=SHOP.ADD_PAYMENT('Electronic money',0);SHOP.ShowStatusOperation(i,1,5);
  -----#2-----
  i:=SHOP.ADD_DELIVERY('The courier',78.5);SHOP.ShowStatusOperation(i,2,1);
  i:=SHOP.ADD_DELIVERY('Packet by rail in territory of Russia',59.22);SHOP.ShowStatusOperation(i,2,2);
  i:=SHOP.ADD_DELIVERY('Packet by air mail in territory of Russia',65);SHOP.ShowStatusOperation(i,2,3);
  i:=SHOP.ADD_DELIVERY('The custom-made parcel post by air mail in territory of other states',89.5);
  SHOP.ShowStatusOperation(i,2,4);
  -----#3-----
  i:=SHOP.ADD_CATEGORY('Digital chambers');SHOP.ShowStatusOperation(i,3,1);
  i:=SHOP.ADD_CATEGORY('Pocket personal computers');SHOP.ShowStatusOperation(i,3,2);
  i:=SHOP.ADD_CATEGORY('Cellular telephones');SHOP.ShowStatusOperation(i,3,3);
  i:=SHOP.ADD_CATEGORY('Flash-memory');SHOP.ShowStatusOperation(i,3,4);
  i:=SHOP.ADD_CATEGORY('Videocards');SHOP.ShowStatusOperation(i,3,5);
  i:=SHOP.ADD_CATEGORY('Processors');SHOP.ShowStatusOperation(i,3,6);
  i:=SHOP.ADD_CATEGORY('Winchesters');SHOP.ShowStatusOperation(i,3,7);
  i:=SHOP.ADD_CATEGORY('DVD-drives');SHOP.ShowStatusOperation(i,3,8);
  -----#4-----
  i:=SHOP.ADD_SUPPLIER('Eqaulins');SHOP.ShowStatusOperation(i,4,1);
  i:=SHOP.ADD_SUPPLIER('Minisota');SHOP.ShowStatusOperation(i,4,2);
  i:=SHOP.ADD_SUPPLIER('Drivston');SHOP.ShowStatusOperation(i,4,3);
  i:=SHOP.ADD_SUPPLIER('Jindison');SHOP.ShowStatusOperation(i,4,4);
  i:=SHOP.ADD_SUPPLIER('Aston');SHOP.ShowStatusOperation(i,4,5);
END;
```

можно увидеть следующее содержание главных таблиц.

Содержимое таблицы TABLE_PAYMENT будет иметь вид:

ID	WAY	COST
1	Cash to the courier	150,78
2	Cash on delivery	50,17
3	The advance payment	0,00
4	Credit card	0,00
5	Electronic money	0,00

Содержимое таблицы TABLE_DELIVERY будет иметь вид:

ID	WAY	COST
1	The courier	78,50
2	Packet by rail in territory of Russia	59,22
3	Packet by air mail in territory of Russia	65,00
4	The custom-made parcel post by air mail in territory of other states	89,50

Содержимое таблицы TABLE_CATEGORY будет иметь вид:

ID	NAME
1	Digital chambers
2	Pocket personal computers
3	Cellular telephones
4	Flash-memory
5	Videocards
6	Processors
7	Winchesters
8	DVD-drives

Содержимое таблицы TABLE_SUPPLIER будет иметь вид:

ID	NAME
1	Eqaulins
2	Minisota
3	Drivston
4	Jindison
5	Aston

Тестирование функции SHOP.ADD_WAREHOUSES

Если не зарегистрировано ни одного товара, то данная функция выполняет лишь две таблицы.

После выполнения кода:

```
DECLARE
  i NUMBER;
BEGIN
  i:=SHOP.ADD_WAREHOUSES('Warehouse #1','Russia','The Samara area','Samara','Kievskaya',1001);
  SHOP.ShowStatusOperation(i,1,1);
  i:=SHOP.ADD_WAREHOUSES('Warehouse #2','Russia','The Samara area','Samara','Kievskaya',1002);
  SHOP.ShowStatusOperation(i,1,2);
  i:=SHOP.ADD_WAREHOUSES('Warehouse #3','Russia','The Samara area','Samara','Kievskaya',1003);
  SHOP.ShowStatusOperation(i,1,3);
END;
```

Содержимое таблицы TABLE_WAREHOUSES будет иметь вид:

ID	NAME	ADDRESS_W_ID
1	Warehouse #1	1
2	Warehouse #2	2
3	Warehouse #3	3

Содержимое таблицы TABLE_ADDRESS_W будет иметь вид:

ID	COUNTRY	AREA	CITY	STREET	HOUSE
1	Russia	The Samara area	Samara	Kievskaya	1001
2	Russia	The Samara area	Samara	Kievskaya	1002
3	Russia	The Samara area	Samara	Kievskaya	1003

Тестирование функции REGISTRATION_NEW_GOODS

После выполнения кода:

```
DECLARE
  i NUMBER;
BEGIN
  i:=SHOP.REGISTRATION_NEW_GOODS('Digital IXUS 700',1,
```

```

'7,1 Mp 3072x2304; 2,0" 118000 p; SD (32 Mb),MMC; JPEG,AVI; USB 2.0.',
'Canon',
'Looks Canon Digital IXUS 700 stylishly due to a combination of the grey matte case and brilliant rims of an objective and a ledge on the case. The camera rather heavy.',
12,1,12446.64);
IF (i>0) THEN SHOP.ShowStatusOperation(1,1,1);
ELSE SHOP.ShowStatusOperation(0,1,1); END IF;
i:=SHOP.REGISTRATION_NEW_GOODS('EXILIM EX-Z750',1,
'7,1 Mp, 3072x2304; SD/MMC, 8,3 Mb; 2,5., 115200 p.; JPEG, AVI (MPEG4 640x480), WAV; USB 2.0; TV
OUT',
'Casio',
'When you take the chamber in hands, that, despite of the compact sizes, at once you understand,
that the question is not simply a dear(an expensive) toy. The case is made very well.',
12,2,13633.29);
IF (i>0) THEN SHOP.ShowStatusOperation(1,1,2);
ELSE SHOP.ShowStatusOperation(0,1,2); END IF;
END;
/

```

Содержимое таблицы TABLE_GOODS будет иметь вид:

ID	NAME	CATEGORY_ID	CHARACTE	BREND	DESCRIP	WARRANTY_PERIOD	SUPPLIER_ID	COST	IMAGE_ID
1	Digital IXUS 700	1	7,1 Mp 3072x	Canon	Looks Canc	12	1	12446,64	1
2	EXILIM EX-Z750	1	7,1 Mp, 3072	Casio	When you t	12	2	13633,29	2

Содержимое таблицы TABLE_INVENTORIES будет иметь вид:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	0
4	3	2	0
5	1	2	0
6	2	2	0

Как видно, товары автоматически зарегистрировались на всех складах.

Тестирование №2 функции SHOP.ADD_WAREHOUSES

Если мы добавим новый склад, то все товары, которые уже зарегистрированы должны автоматически зарегистрироваться и на этом складе.

```

DECLARE
i NUMBER;
BEGIN
i:=SHOP.ADD_WAREHOUSES('Warehouse #4','Russia','The Samara area','Samara','Kievskaya',1004);
SHOP.ShowStatusOperation(i,1,1);
END;
/

```

Содержимое таблицы TABLE_WAREHOUSES будет иметь вид:

ID	NAME	ADDRESS_W_ID
1	Warehouse #1	1
2	Warehouse #2	2
3	Warehouse #3	3
4	Warehouse #4	4

Содержимое таблицы TABLE_ADDRESS_W будет иметь вид:

ID	COUNTRY	AREA	CITY	STREET	HOUSE
1	Russia	The Samara area	Samara	Kievskaya	1001
2	Russia	The Samara area	Samara	Kievskaya	1002
3	Russia	The Samara area	Samara	Kievskaya	1003
4	Russia	The Samara area	Samara	Kievskaya	1004

Содержимое таблицы TABLE_INVENTORIES будет иметь вид:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	0
4	3	2	0
5	1	2	0
6	2	2	0
7	4	1	0
8	4	2	0

Последние две строчки 7 и 8 говорят о том, что товары 1 и 2, которые были зарегистрированы на складах 1,2,3, также зарегистрировались на складе 4.

Тестирование №2 функции SHOP.REGISTRATION_NEW_GOODS

Если мы добавляем новый товар, то он должен автоматически регистрироваться на всех складах.

```
DECLARE
i NUMBER;
BEGIN
i:=SHOP.REGISTRATION_NEW_GOODS('FinePix F10',1,
'6,1 Mp, 2848x2136; xD, 16 Mb; 2,5", 115000 p.; JPEG, AVI (640x480), WAV; USB 2.0, A/V; EXIF print',
'FUJIFILM',
'The chamber has the all-metal case, stylish design and the convenient form. On the big 2,5. TFT-
display the special covering protecting it(him) from scratches.',
12,3,10257.93);
IF (i>0) THEN SHOP.ShowStatusOperation(1,1,1);
ELSE SHOP.ShowStatusOperation(0,1,1); END IF;
END;
/
```

Содержимое таблицы TABLE_GOODS будет иметь вид:

ID	NAME	CATEGORY_ID	CHARACTERISTICS	BREND	DESCRIPTION	WARRANTY_PERIOD	SUPPLIER_ID	COST	IMAGE_ID
1	Digital IXUS 700	1	7,1 Mp 3072x2304; 2,0"	Canon	Looks Canon Digita	12	1	12446,64	1
2	EXILIM EX-Z750	1	7,1 Mp, 3072x2304; SD	Casio	When you take the	12	2	13633,29	2
3	FinePix F10	1	6,1 Mp, 2848x2136; xD	FUJIFILM	The chamber has t	12	3	10257,93	3

Содержимое таблицы TABLE_INVENTORIES будет иметь вид:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	0
4	3	2	0
5	1	2	0
6	2	2	0
7	4	1	0
8	4	2	0
9	3	3	0
10	1	3	0
11	2	3	0
12	4	3	0

Как показывают строки 9,10,11,12, товар с идентификатором 3 успешно зарегистрировался на складах 1,2,3,4.

Тестирование функции SHOP.ISSUE_WAYBILL

В магазин поступили товары. Оформим накладные по приходу товара на склад.

```
DECLARE
i NUMBER(10);
BEGIN
i:=SHOP.ISSUE_WAYBILL(1,1,5);DBMS_OUTPUT.PUT_LINE(i);
i:=SHOP.ISSUE_WAYBILL(1,2,7);DBMS_OUTPUT.PUT_LINE(i);
i:=SHOP.ISSUE_WAYBILL(2,1,9);DBMS_OUTPUT.PUT_LINE(i);
i:=SHOP.ISSUE_WAYBILL(2,2,3);DBMS_OUTPUT.PUT_LINE(i);
i:=SHOP.ISSUE_WAYBILL(3,1,7);DBMS_OUTPUT.PUT_LINE(i);
END;
/
```

Содержимое таблицы TABLE_INVENTORIES будет иметь вид:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	7
2	1	1	5
3	2	1	9
4	3	2	0
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	1	3	0
11	2	3	0
12	4	3	0

Если товар поступил на склад, где он уже присутствует в определенном количестве, то количество суммируется.

```

DECLARE
  i NUMBER(10);
BEGIN
  i:=SHOP.ISSUE_WAYBILL(1,1,17);DBMS_OUTPUT.PUT_LINE(i);
  i:=SHOP.ISSUE_WAYBILL(3,2,45);DBMS_OUTPUT.PUT_LINE(i);
END;
/

```

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	7
2	1	1	22
3	2	1	9
4	3	2	45
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	1	3	0
11	2	3	0
12	4	3	0

Тестирование функции SHOP.REGISTRATION

При выполнении кода:

```

DECLARE
  i NUMBER(10);
BEGIN
  i:=SHOP.REGISTRATION('aslan','79FF3A8ADE43676B973B9DF5EDC9EBAB','aslan@google.com',
    31525398,'Anrey Genadevich Mohov',56465,'Russia','Samarskaya',
    'Samara','Promishlennaya',190,0,36);

  DBMS_OUTPUT.PUT_LINE(i);
  i:=SHOP.REGISTRATION('laden','F8D12C06B5A4BC29F2C5B22785D3EF9D','laden@google.com',
    51525394,'Mihail Genadevich Gigulin',56465,'Russia','Ulyanovckaya',
    'Ulyanovck','Sovetskaya',285,0,88);

  DBMS_OUTPUT.PUT_LINE(i);
  i:=SHOP.REGISTRATION('liwer','39F89EF83A78C46BF2C5B22785D3EF9D','liwer@google.com',
    37825394,'Sergey Genadevich Gukov',56465,'Russia','Tomskaya',
    'Tomsk','Kalugskaya',100,0,96);

  DBMS_OUTPUT.PUT_LINE(i);
END;
/

```

можно авторизоваться на сайте от имени тех клиентов, которые были упомянуты выше.

Тестирование функции SHOP.REGISTRATION_ORDER

1. В заказе всего один товар А.

Пусть клиент заказал товар А с количеством N. Всего на складах присутствует М единиц данного товара. Склады: $S_1..S_N$.

Тестирование данной функции заключается в следующем:

1.1. На первом складе S_1 есть N товаров для выполнения заказа.

1.2. На первом складе S_1 нет N товаров для выполнения заказа.

1.3. На всех складах суммарное количество $M < N$.

2. В заказе товары $A_1..A_N$.

Пусть клиент заказал товары $A_1..A_N$ с количеством $N_1..N_N$.

Данная процедура ничем не отличается от пункта №1, за исключением того, что процедура выборки товара A_i будет циклически повторяться для каждого товара $A_i\{i:1..N\}$.

Первоначальное состояние положений дел на складах отображается в таблице TABLE_INVENTORIES:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	7
2	1	1	22
3	2	1	9
4	3	2	45
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	1	3	0
11	2	3	0
12	4	3	0

- Тестирование пункта 1.1.

Пусть в заказе всего один товар с идентификатором 1. Клиент заказал 7 единиц товара.

```
DECLARE
  i NUMBER(10);
BEGIN
  i:=SHOP.REGISTRATION_ORDER('1:7;',1,1,1);
  DBMS_OUTPUT.PUT_LINE(i);
END;
/
```

Тогда после выполнения процедуры таблица TABLE_INVENTORIES будет выглядеть следующим образом:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	22
3	2	1	9
4	3	2	45
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	4	3	0
11	1	3	0
12	2	3	0

Содержимое таблицы TABLE_WAYBILL_ORDER будет иметь вид:

ID	TIMES	STATUS	USER_ID	PAYMENT_ID	DELIVERY_ID
1	28.01.2007 15:48:28	0	1	1	1

Содержимое таблицы TABLE_GOODS_ORDER будет иметь вид:

ID	GOODS_ID	AMOUNT	WAYBILL_ORDER_ID
1	1	7	1

Содержимое таблицы TABLE_PACKING будет иметь вид:

ID	WAREHOUSES_ID	AMOUNT	GOODS_ORDER_ID
1	3	7	1

Из таблицы TABLE_PACKING видно, что всего одного склада хватило, чтобы выполнить заказ.

- Тестирование пункта 1.2.

Пусть в заказе всего один товар с идентификатором 1. Клиент заказал 35 единиц товара.

```
DECLARE
  i NUMBER(10);
BEGIN
  i:=SHOP.REGISTRATION_ORDER('1:35;',1,1,1);
  DBMS_OUTPUT.PUT_LINE(i);
END;
/
```

Тогда после выполнения процедуры таблица TABLE_INVENTORIES будет выглядеть следующим образом:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	3
4	3	2	45
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	4	3	0
11	1	3	0
12	2	3	0

Содержимое таблицы TABLE_GOODS_ORDER будет иметь вид:

ID	GOODS_ID	AMOUNT	WAYBILL_ORDER_ID
1	1	35	1

Содержимое таблицы TABLE_PACKING будет иметь вид:

ID	WAREHOUSES_ID	AMOUNT	GOODS_ORDER_ID
1	3	7	1
2	1	22	1
3	2	6	1

- Тестирование пункта 1.3.

Пусть в заказе всего один товар с идентификатором 1. Клиент заказал 39 единиц товара. Хотя всего на складах 38 единиц данного товара.

```
SQL> DECLARE
2   i NUMBER(10);
3   BEGIN
4   i:=SHOP.REGISTRATION_ORDER('1:39;',1,1,1);
5   DBMS_OUTPUT.PUT_LINE(i);
6   END;
7   /
```

1
PL/SQL procedure successfully completed

Нам показано сообщение о том, что товара с ID=1 не хватает на складах.

- Тестирование пункта 2.

Пусть в заказе два товара с идентификатором 1 и 2. Клиент заказал 30 единиц товара 1 и 15 единиц товара 2.

```
DECLARE
i NUMBER(10);
BEGIN
i:=SHOP.REGISTRATION_ORDER('1:30;2:15;',1,1,1);
DBMS_OUTPUT.PUT_LINE(i);
END;
/
```

Тогда после выполнения процедуры таблица TABLE_INVENTORIES будет выглядеть следующим образом:

ID	WAREHOUSES_ID	GOODS_ID	AMOUNT
1	3	1	0
2	1	1	0
3	2	1	8
4	3	2	30
5	1	2	7
6	2	2	3
7	4	1	0
8	4	2	0
9	3	3	0
10	4	3	0
11	1	3	0
12	2	3	0

Содержимое таблицы TABLE_GOODS_ORDER будет иметь вид:

ID	GOODS_ID	AMOUNT	WAYBILL_ORDER_ID
1	1	30	1
2	2	15	1

Содержимое таблицы TABLE_PACKING будет иметь вид:

ID	WAREHOUSES_ID	AMOUNT	GOODS_ORDER_ID
1	3	7	1
2	1	22	1
3	2	1	1
4	3	15	2

Отсюда видно, что программа работает верно.

1.10. Вид рабочей программы

Внешний вид программы представлен на рисунке 4.

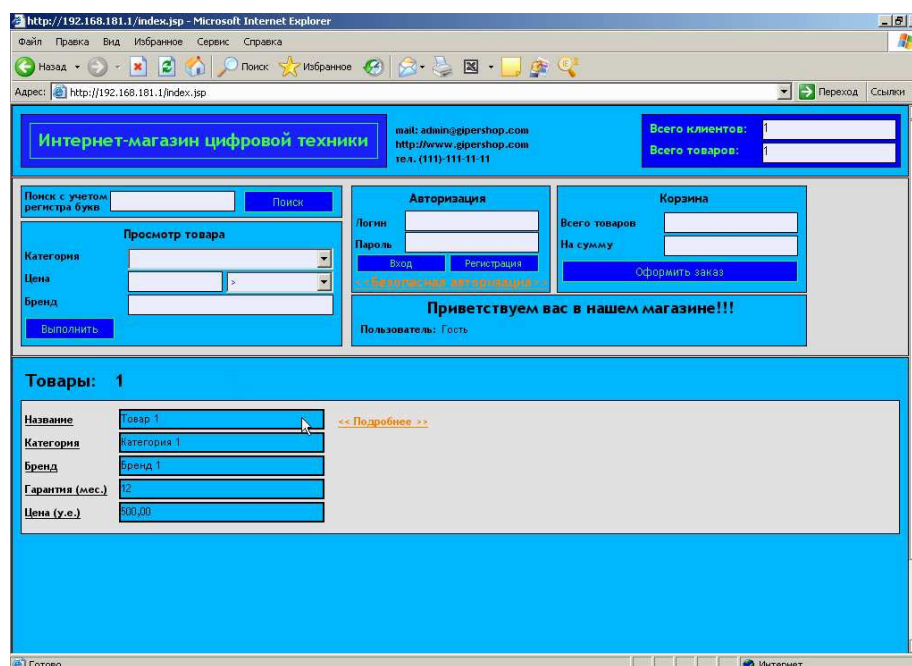


Рис. 4. Интерфейс клиента
Изображение товара можно увидеть на рисунке 5.

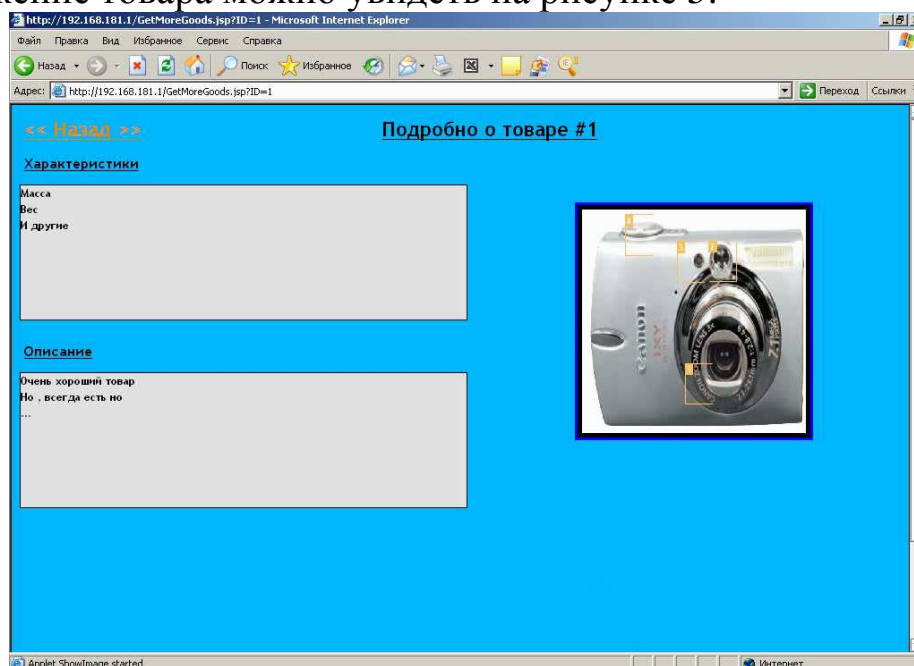


Рис. 5. Карточка товара

2. ЭКСПЛУАТАЦИОННАЯ ДОКУМЕНТАЦИЯ

2.1. Руководство программиста

Назначение и условия применения программы

Система предназначена для обработки заказов от пользователей в режиме реального времени.

В состав технических средств должны входить два IBM-совместимых персональных компьютера (ПЭВМ), первый – для выполнения функций Web-сервера, второй – под сервер баз данных.

Машина под СУБД должна быть очень надежной и иметь такие параметры:

- а) процессор с тактовой частотой, МГц – 2400, не менее;
- б) материнскую плату с FSB, МГц – 533, не менее;
- в) оперативную память объемом, Мб – 512, не менее;
- г) жесткий диск объемом, Гб – 200, не менее; все зависит от предполагаемого объема данных.

На данном компьютере должны быть установлены ОС Linux Red Hat Enterprise и СУБД Oracle 9i.

Данное ПО должно работать в режиме 24x7 (24 часа 7 дней в неделю).

Машина под Web-сервер также должна иметь конфигурацию:

- а) процессор с тактовой частотой, МГц – 1600, не менее;
- б) материнскую плату с FSB, МГц – 366, не менее;
- в) оперативную память объемом, Мб – 256, не менее;
- г) жесткий диск объемом, Гб – 2, не менее.

На данной машине должна быть установлена виртуальная Java-машина версии не ниже 1.2. (J2SE) и Web-сервер Blazix или Tomcat с дополнительными библиотеками (из J2EE).

Данное ПО должно работать в режиме 24x7 (24 часа 7 дней в неделю).

Характеристика программы

Ниже перечислены основные технические характеристики программы.

- Обработка заказов в режиме реального времени.
- Многопоточная обработка пользовательских запросов.
- Высокий уровень безопасности.
- Независимость клиентской части от операционной системы.
- Независимость пользовательской части от операционной системы.
- Независимость административной части от операционной системы.
- Управление данными с помощью СУБД Oracle 9i.
- Программное обеспечение Web-сервера – Blazix и Java-платформы J2EE и J2SE.
- Аппаратная платформа – Intel.

Инсталляция на Web-сервере

В стандартную поставку программного продукта входит инсталляционный пакет GiperShopPersonal. Пакет нужен для развертывания Интернет-магазина на сервере. Установка пакета проходит в несколько этапов:

1. Появляется окно с выбором языка интерфейса.
2. Появляется окно с общим ознакомлением.
3. Появляется окно с лицензионным соглашением.
4. Появляется окно с выбором папки установки.
5. Появляется окно с окончательными выбранными параметрами для установки и предлагается нажать кнопку “Установить”.
6. Далее появляется окно с завершением мастера установки.
7. Убедиться, что ярлыки к программе установлены.

Инсталляция на сервере СУБД

В стандартную поставку программного продукта входит инсталляционный пакет GiperShopSql. Установка пакета проходит в несколько этапов:

1. Появляется окно с выбором языка интерфейса.
2. Появляется окно с общим ознакомлением.
3. Появляется окно с лицензионным соглашением.
4. Появляется окно с выбором папки установки.
5. Появляется окно с окончательными выбранными параметрами для установки и предлагается нажать кнопку “Установить”.
6. Далее появляется окно с завершением мастера установки.
7. Убедиться, что ярлыки к программе установлены.
8. Запустить сценарии SQL, находящиеся в установленной директории из командной утилиты SQL Plus для создания хранимых объектов в базе данных Oracle и привилегий к ним.

Инсталляция на администраторской машине

В стандартную поставку программного продукта входит инсталляционный пакет GiperShopAdmin. Установив данный пакет, администратор сможет управлять персоналом магазина.

Установка пакета проходит в несколько этапов:

1. Появляется окно с выбором языка интерфейса.
2. Появляется окно с общим ознакомлением.
3. Появляется окно с лицензионным соглашением.
4. Появляется окно с выбором папки установки.
5. Появляется окно с окончательными выбранными параметрами для установки и предлагается нажать кнопку “Установить”.
6. Далее появляется окно с завершением мастера установки.

7. Убедиться, что ярлыки к программе установлены.

Обращение к программе

Для запуска исполняемого модуля GiperShopAdmin необходимо передать параметры:

<SID> <host> <port> <key>

где SID – идентификатор экземпляра базы данных Oracle; host – машина, на которой работает сервер Oracle; port – удаленный порт для соединения; key – секретный файл с ключом.

Если необходимо воспользоваться дополнительной функцией, то необходимо передать главный параметр -ISO и три для обрабатываемых файлов jsp: папка, где содержатся обрабатываемые файлы, папка, куда будут созданы новые файлы и собственно обрабатываемый файл.

Сеанс работы администратора магазина можно увидеть на рисунке 6:

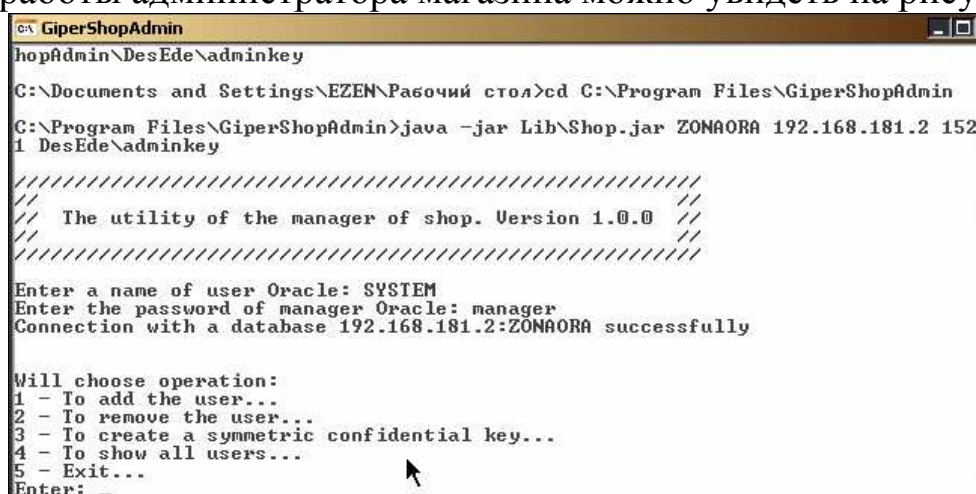


Рис. 6. “Рабочий кабинет” администратора

Для запуска Web-сервера администратору необходимо запустить главный bat-файл, находящийся в инсталлируемой директории GiperShopPersonal. При запуске можно увидеть примерно следующее:

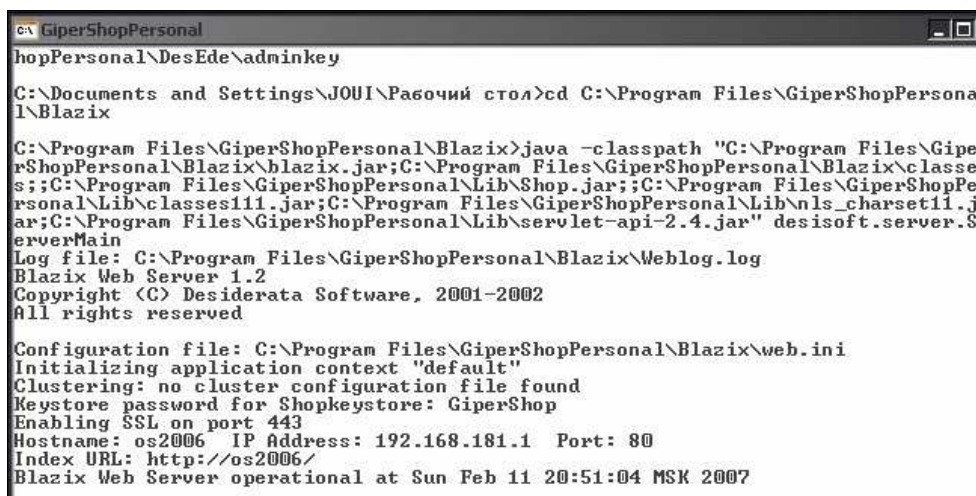


Рис. 7. Запуск Web-сервера

После того как мы установили и запустили “ядро” Интернет-магазина, можно переходить напрямую к работе через Web-интерфейс.

Все, что остается сделать персоналу – запустить браузер и ввести следующее:

`http://myhost/indexw.jsp`.

После этого необходимо авторизоваться, предварительно получив логин и пароль у администратора, и получить полноценную рабочую комнату персонала магазина.

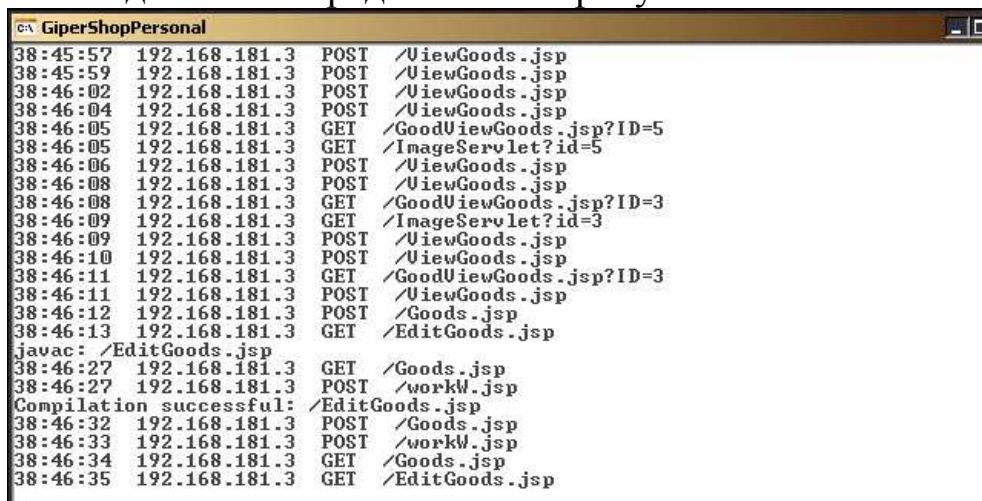
Деинсталляция

Для деинсталляции каждого из трех пакетов необходимо запустить программу `unins000.exe` в каждой из трех устанавливаемых директорий. Произойдет автоматическое удаление всех частей программной системы.

Сообщения

Программа не имеет встроенных средств для генерации файлов с сообщениями. Если в главном модуле программы или при выполнении исполняемого модуля Web-сервера возникли какие-либо исключительные ситуации или обычные ситуации, то в консоли Web-сервера их можно будет увидеть. Но программисту при этом ничего ненужно будет принимать. Исключительные ситуации все отрабатываются внутри системы. Программа сама позаботится об откате транзакций и т.п.

Типичный вид консоли представлен на рисунке ниже:



```
GiperShopPersonal
38:45:57 192.168.181.3 POST /ViewGoods.jsp
38:45:59 192.168.181.3 POST /ViewGoods.jsp
38:46:02 192.168.181.3 POST /ViewGoods.jsp
38:46:04 192.168.181.3 POST /ViewGoods.jsp
38:46:05 192.168.181.3 GET /GoodViewGoods.jsp?ID=5
38:46:05 192.168.181.3 GET /ImageServlet?id=5
38:46:06 192.168.181.3 POST /ViewGoods.jsp
38:46:08 192.168.181.3 POST /ViewGoods.jsp
38:46:08 192.168.181.3 GET /GoodViewGoods.jsp?ID=3
38:46:09 192.168.181.3 GET /ImageServlet?id=3
38:46:09 192.168.181.3 POST /ViewGoods.jsp
38:46:10 192.168.181.3 POST /ViewGoods.jsp
38:46:11 192.168.181.3 GET /GoodViewGoods.jsp?ID=3
38:46:11 192.168.181.3 POST /ViewGoods.jsp
38:46:12 192.168.181.3 POST /Goods.jsp
38:46:13 192.168.181.3 GET /EditGoods.jsp
javac: /EditGoods.jsp
38:46:27 192.168.181.3 GET /Goods.jsp
38:46:27 192.168.181.3 POST /workW.jsp
Compilation successful: /EditGoods.jsp
38:46:32 192.168.181.3 POST /Goods.jsp
38:46:33 192.168.181.3 POST /workW.jsp
38:46:34 192.168.181.3 GET /Goods.jsp
38:46:35 192.168.181.3 GET /EditGoods.jsp
```

Рис. 8. Сообщения Web-сервера

ЗАКЛЮЧЕНИЕ

В результате проделанной работы был разработан механизм Web-доступа к базе данных Oracle 9i. Данный механизм позволяет получить доступ к информации Oracle 9i из внешних приложений, например, из скриптовых языков. Это делает возможным интегрировать Oracle 9i с Интернет-технологиями, что в свою очередь позволяет решать широкий круг задач. Примерами таких задач могут являться: создание Интернет-магазина и организация электронной коммерции, Web -доступ к базе данных из удаленных подразделений фирмы и прочее. Следовательно, предприятие, учреждение или организация может существенно расширить сферу своей деятельности, автоматизировать ее, не прибегая к использованию другой учетной системы и работая в рамках одной платформы. Это намного эффективнее и продуктивнее, позволяет автоматизировать деятельность и, следовательно, снизить расходы на ее поддержание.

Работа механизма доступа основана на использовании возможностей технологии JDBC. Очень важно то, что разработка позволяет осуществлять многопоточную работу с данными, хранящимися в базе данных, и механизм транзакций, реализованный в Oracle и JDBC, позволяет это сделать с легкостью. Механизм синхронизации потоков, реализованный в Java, и используемый в данной работе, позволяет “общаться” с базой данных множеству потоков в одном процессе Web-сервера. Все это необходимо при реализации полноценных Web-приложений, таких как Интернет-магазин.

Автор достиг цели – создать полноценное Web-приложение, реализующее механизм работы Интернет-магазина. На его примере была показана работа механизма доступа к базе данных, а также основные функциональные элементы Интернет-магазина.

Данная разработка – актуальна, так как сервер баз данных Oracle является стандартом для построения больших кластерных и корпоративных систем, реализуем на всех платформах. В то время как все упомянутые технологии Java от Sun – стандарт при построении мощных, распределенных и безопасных Web-приложений.

Проект имеет все шансы стать коммерческим проектом при решении следующих задач:

- доработать защиту от SQL-инъекций;
- добавить поддержку оплаты с помощью Web Money и Credit Card;
- добавить функцию “Обратная связь с покупателем”;
- добавить форум по обсуждению товаров и работы магазина;
- получить сертификат от VeriSign;
- написать собственный FTP-сервер для загрузки изображений с машин персонала магазина;
- оптимизировать хранимые функции и процедуры для наивысшего выполнения поставленных задач;
- добавить процедуру для отправки писем клиентам.

ИСТОЧНИКИ ИНФОРМАЦИИ

1. Д. Флэнаган, Java. Справочник, 4-е издание – Пер. с англ. – СПб: Символ-Плюс, 2004. – 1040 с., ил.
2. К. Хорстманн, Г. Корнелл, Java 2 Библиотека профессионала, том 2. Тонкости программирования, 7-е изд.: Пер. с англ. – М.: Издательский дом “Вильямс”, 2006. – 1168 с.: ил. – Парал. тит. англ.
3. Н. Моррисо-Леруа, М. Соломон, Д. Басу, ORACLE 8i: Java-компонентное программирование при помощи EJB, CORBA и JSP, Издательство “ЛОРИ”, 2001.
4. С. Урман, Oracle8: Программирование на языке PL/SQL, Издательство “ЛОРИ”, 2001.
5. К. Луни, М. Терьо и эксперты TUSC, Oracle 9i: Настольная книга администратора, Издательство “ЛОРИ”, 2004.
6. В. Будилов, Интернет-программирование на Java. – СПб.: БХВ-Петербург, 2003. – 704 с.: ил.