

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЕВА
(национальный исследовательский университет)»

**Комплект учебно-методических указаний
к лабораторным работам по курсу
«Технологии промышленного распределённого программирования»**

Электронные учебно-методические указания к лабораторным работам

Самара
2010

Составитель: ГАВРИЛОВ Андрей Вадимович

В пособии приведены методические указания к лабораторным работам по дисциплине "Технологии промышленного распределенного программирования".

Учебное пособие предназначено для студентов, обучающихся по магистерской программе "Технологии параллельного программирования и суперкомпьютинг" по направлению 010400.68 "Прикладная математика и информатика" и изучающих дисциплину "Технологии промышленного распределенного программирования".

Разработано на кафедре технической кибернетики.

Лабораторная работа №1

Разработка базы данных

Темы для предварительного изучения

Реляционные базы данных

Схема базы данных

Язык SQL

Команды DDL

Команды DML

Задание на лабораторную работу

Разработать схему базы данных и реализовать её в выбранной СУБД.

Введение

Полученные в ходе выполнения этой работы модели и реализация базы данных будут использоваться на протяжении всего дальнейшего курса лабораторных работ.

В качестве СУБД может использоваться любая SQL-совместимая СУБД, для которой существует JDBC-драйвер (желательно четвёртого типа). Однако рекомендуется использование Oracle (допускается учебная версия OracleXE), поскольку в нём присутствуют корректная поддержка транзакций, язык хранимых процедур и ряд других расширений стандартного SQL, также для него существуют бесплатные средства разработки.

При работе с СУБД рекомендуется использование графических средств разработки, однако возможна работа и с консольным управлением СУБД. Для формирования логической и физической моделей БД настоятельно рекомендуется использование графических средств разработки.

Задание 1

Выбрать и согласовать с преподавателем предметную область, для которой будет создаваться база данных.

Задание 2

Разработать логическую модель базы данных.

В модели должны присутствовать минимум 4 вида сущностей, а также все виды отношений: один к одному, один ко многим, многие ко многим.

Задание 3

Разработать физическую модель базы данных.

Задание 4

Создать скрипт, формирующий таблицы в базе данных. Убедиться, что корректно указаны все ограничения из логической модели, а также что правильно реализованы все связи между таблицами.

Задание 5

Создать скрипт, заполняющий созданную базу данными. В каждой из таблиц должно быть не менее 5 записей, при этом должны быть реализованы все возможные виды отношений.

|

Лабораторная работа №2

Настольное приложение на Java

Темы для предварительного изучения

Синтаксис языка Java

Механизмы ввода-вывода

Сериализация

Средства создания графических приложений

JavaBeans

Задание на лабораторную работу

Разработать настольное (desktop) графическое приложение на Java, позволяющее просматривать и редактировать сущности выбранной в первой работе предметной области. Ознакомиться с принципом Model-View-Controller.

Задание 1

Разработать набор Java-классов, каждый из которых описывает сущность в рамках созданной ранее логической модели БД. Классы должны удовлетворять требованиям JavaBeans.

Обратить особое внимание на реализацию инкапсуляции, а также на реализацию отношений между сущностями в рамках возможностей Java.

Задание 2

Создать класс модели, агрегирующий в себе все данные модели в виде объектов классов из задания 1.

Подготовить все классы к сериализации.

Задание 3

Разработать класс контроллера, проверяющего корректность изменений данных в модели.

Задание 4

Разработать графическое приложение, являющееся представлением модели и использующее контроллер для модификации данных модели.

Приложение должно позволять сохранять и считывать модель в файлы (средствами сериализации), редактировать свойства сущностей, удалять сущности (с учётом зависимостей) и создавать новые сущности.

Лабораторная работа №3

Сетевое приложение на Java

Темы для предварительного изучения

Синтаксис языка Java

Механизмы ввода-вывода

Сериализация

Средства создания графических приложений

Сокеты

Многопоточные приложения

Задание на лабораторную работу

Разработать клиент-серверное приложение на Java на основе сокетов, позволяющее просматривать и редактировать сущности выбранной в первой работе предметной области. Ознакомиться с образцом проектирования Observer.

Введение

Отличие от лабораторной работы №2 должно заключаться в том, что все данные должны храниться на сервере. Клиент может получать, просматривать и изменять их, как и ранее.

Клиент не должен держать копию данных на своей стороне.

При одновременной работе нескольких клиентов следует не давать клиенту редактировать данные, уже редактируемые другим клиентом.

При изменении данных одним клиентом следует оповещать других клиентов, чтобы они обновили своё состояние.

Задание 1

Разработать протокол взаимодействия между клиентом и сервером, отражающий возможные ситуации:

- получение клиентом данных о сущности;
- создание новой сущности;
- удаление сущности;
- начало изменения данных сущности;
- завершение изменения данных сущности;
- оповещение о создании сущности;
- оповещение об удалении сущности;
- оповещение об изменении сущности;
- и т.д.

Задание 2

Разработать архитектуру (например, в виде диаграмм UML) клиентской и серверной части, с учётом использовавшегося в первой работе MVC.

Клиентская сторона должна быть готова принять информацию (оповещение) от сервера в любой момент, независимо от действий пользователя.

Серверная сторона должна отслеживать сущности, с которыми работают клиенты, и оповещать всех активных клиентов об изменении сущностей.

Задание 3

Разработать программы клиентской и серверной части.

Лабораторная работа №4

Тестирование с JUnit

Темы для предварительного изучения

Синтаксис языка Java

Регулярные выражения

Основы TDD

Технология JUnit

Задание на лабораторную работу

Добавить в разрабатываемое приложение контроль ввода данных пользователем. Для проверки корректности данных следует использовать регулярные выражения. Разработка регулярных выражений должна проводиться в соответствии с методологией TDD, для тестирования должна использоваться технология JUnit.

Задание 1

Определить виды данных, вводимых пользователем при редактировании параметров сущностей, ограничения на возможные значения.

Задание 2

Добавить в систему на клиентской стороне объект, осуществляющий контроль ввода:

- определить интерфейс этого объекта;
- реализовать методы в виде простых заглушек.

Задание 3

Разработать в рамках JUnit систему тестов для проверки корректности контроля.

Задание 4

В соответствии с TDD постепенно реализовать функциональное наполнение методов контроля вводимых значений.

Лабораторная работа №5

XML

Темы для предварительного изучения

Синтаксис языка Java

XML

DTD

XML Schema

Работа с XML на Java

DOM

JAXB

Задание на лабораторную работу

Заменить способ хранения и передачи данных между клиентом и сервером на XML-документы.

Задание 1

Разработать DTD для XML-документа, хранящего данные приложения на стороне сервера.

Задание 2

Разработать XML Schema для XML-документа, хранящего данные приложения на стороне сервера.

Задание 3

Изменить механизмы сохранения данных на стороне сервера таким образом, чтобы сохранение и восстановление данных происходило в/из XML-документа, описываемого разработанной схемой.

Задание 4

Изменить протокол взаимодействия клиента с сервером таким образом, чтобы данные о сущностях передавались по сети в виде XML-документов.

Задание 5

С применением XML изменить серверную часть программы, отвечающую за передачу данных клиентам и за получение данных от них.

Задание 6

С применением XML изменить клиентскую часть программы, отвечающую за передачу данных серверу и за получение данных от него.

Лабораторная работа №6

JDBC

Темы для предварительного изучения

Синтаксис языка Java

JavaBeans

Java Collections Framework

SQL

JDBC

Задание на лабораторную работу

Модифицировать серверную часть программы таким образом, чтобы хранение данных обеспечивалось СУБД. Клиентскую часть программы по сути должна стать клиентом для базы данных.

Задание 1

Развернуть в выбранной СУБД базу данных с помощью разработанных в ходе выполнения лабораторной работы 1 скриптов.

Задание 2

С применением технологии JDBC разработать класс, обеспечивающий доступ к базе данных и позволяющий получать данные из базы и вносить в неё изменённые данные.

Методы класса должны в качестве типов параметров и типов возвращаемых значений иметь ссылки на объекты классов JavaBeans, разработанных во второй лабораторной работе, либо их коллекции.

Для получения соединения с базой рекомендуется использование класса DriverManager.

Задание 3

В серверном приложении изменить модуль сохранения данных таким образом, чтобы данные получались и изменялись посредством разработанного в задании 2 класса доступа к данным.

Задание 4

Изменить в серверном приложении блок взаимодействия с клиентами таким образом, чтобы:

- данные извлекались из СУБД только в том случае, если их запросил клиент;
- отсутствовала репликация данных в виде копий объектов на стороне сервера (т.е. сервер должен отслеживать загруженные сущности по значениям первичных ключей);
- данные должны заноситься в базу данных после каждой успешной операции со стороны клиента.

Лабораторная работа №7

Servlets. JSP

Темы для предварительного изучения

Синтаксис языка Java

Web-приложения на Java

Servlets

JSP

XML

HTML

Задание на лабораторную работу

Разработать Web-приложение, реализующее тонкий клиент для базы данных.

Задание 1

Разработать сервлет (или набор сервлетов), обеспечивающий взаимодействие с СУБД.

Сервлет должен позволять извлекать данные из базы, а также вносить данные в базу.

Для работы с базой следует использовать разработанный в лабораторной работе 6 класс доступа к СУБД.

Результаты выполнения запросов должны сохраняться в строковые объекты, являющиеся атрибутами сессии, в которых данные представлены в виде XML-фрагментов.

При написании сервлета может быть использована значительная часть разрабатывавшегося ранее серверного приложения.

Задание 2

Разработать web-интерфейс приложения в виде набора JSP-страниц.

Должны присутствовать страницы просмотра списка сущностей каждого вида (с поиском хотя бы по одному критерию), а также страницы редактирования для каждого вида сущностей из базы данных.

Кроме того, должны быть реализованы страницы сообщений об ошибках.

Все запросы на обработку и получение данных должны переадресовываться сервлетам.

Все данные для вывода пользователю должны быть получены из строковых объектов с XML-фрагментами, сформированных сервлетами.

JSP-страницы должны содержать только форматированный вывод информации, проверку корректности ввода информации, и навигацию между страницами.

При написании JSP-страниц может быть использована значительная часть разрабатывавшегося ранее клиентского приложения.

Лабораторная работа №8

Session Beans

Темы для предварительного изучения

Синтаксис языка Java

Web-приложения на Java

EJB

Session Beans

JNDI

Servlets

HTML

Задание на лабораторную работу

Разработать Enterprise-приложение, реализующее работу с базой данных с применением сессионных компонентов

Задание 1

С помощью средств конфигурирования сервера приложений создать на нём пул соединений (Connection Pool) для вашей базы данных, а также источник данных (Data Source) для работы с ней (на основе пула соединений).

Задание 2

Разработать сессионный компонент (или набор компонентов) без состояния, имеющий удалённый интерфейс и предназначенный для извлечения данных из СУБД по запросу клиента.

Использующийся класс для доступа к СУБД следует изменить таким образом, чтобы соединение получалось из объекта типа DataSource, полученного у сервера приложений средствами JNDI.

Методы сессионного компонента должны иметь своими параметрами типы сущностей, разработанные ранее, или их коллекции.

Задание 3

Модифицировать сервлеты web-модуля таким образом, чтобы взаимодействие с СУБД они осуществляли через сессионный компонент (компоненты).

Получение ссылки на компонент можно выполнить единственный раз при инициализации сервлета.

Лабораторная работа №9

Entity Beans (BMP)

Темы для предварительного изучения

Синтаксис языка Java
Web-приложения на Java
EJB
Session Beans
Entity Beans (BMP)
JNDI

Задание на лабораторную работу

Разработать Enterprise-приложение, реализующее работу с базой данных с применением сущностных компонентов.

Задание 1

Разработать сущностные компоненты в режиме Bean Managed Persistence для всех сущностей вашей модели.

Компонентный интерфейс этих компонентов должен в целом напоминать методы классов JavaBeans, использовавшихся для этих же сущностей.

Методы получения ссылок на связанные сущности должны возвращать тип компонентного интерфейса соответствующего Entity Bean.

Интерфейсы доступа к компонентам должны быть локальными.

Задание 2

Модифицировать дескриптор развёртывания EJB-модуля таким образом, чтобы разработанный ранее сессионный компонент мог локальным образом получить доступ к разработанным в задании 1 сущностным компонентам.

Для этого потребуется использовать дополнительный тег `<ejb-local-ref>`.

Задание 3

Модифицировать сессионный компонент таким образом, чтобы всё взаимодействие с СУБД он производил через разработанные в задании 1 сущностные компоненты.

Доступ к локальным объектам сущностных компонентов может быть получен через JNDI.

Лабораторная работа №10

JMS. MDB

Темы для предварительного изучения

Синтаксис языка Java

EJB

Session Beans

Entity Beans (BMP)

JNDI

JDBC

JMS

MDB

Задание на лабораторную работу

Добавить в разработанное Enterprise-приложение функцию ведения журнала изменения состояния сущностей, реализовав её через механизмы JMS и MDB.

Задание 1

Добавить в схему используемой базы данных таблицу для хранения сообщений об изменениях сущностей системы. Каждая строка должна содержать первичный ключ, время изменения сущности, значение первичного ключа сущности, поле с именем сущности (или аналогичным свойством, позволяющим зрительно различить сущности).

Задание 2

Выбрать домен (точка-точка или публикация-подписка) передачи сообщений об изменениях состояния сущностей, обосновать свой выбор.

Задание 3

Определить тип сообщений JMS, более удобный для передачи данных об изменениях состояния компонента, выбрать внутреннюю структуру сообщения, обосновать свой выбор.

Задание 4

Средствами администраторской консоли сервера приложений создать целевой объект (очередь или тему) для передачи сообщений, а также настроить фабрику соединений с JMS.

Задание 5

Разработать MDB-компонент, получающий сообщения из созданного целевого объекта.

Обработка сообщений должна заключаться во внесении в таблицу из задания 1 данных о произошедшем изменении.

Для работы с базой можно использовать функции JDBC, однако соединение следует получать из объекта типа DataSource, который следует находить с помощью механизмов JNDI.

Задание 6

Модифицировать код сущностных компонентов таким образом, чтобы при изменении состояния сущности в созданный в задании 4 целевой объект отправлялось сообщение об изменении состояния компонента.

Лабораторная работа №11

EJB 3

Темы для предварительного изучения

Синтаксис языка Java

EJB 3

Session Beans

JNDI

JDBC

JMS

MDB

Dependency Injection

Servlets

Задание на лабораторную работу

Модифицировать разработанное Enterprise-приложение с учётом нововведений стандарта EJB 3.

Задание 1

Изменить разработанный ранее MDB-компонент с учётом требований EJB 3.

Для получения ссылки на Data Source следует использовать механизм внедрения зависимостей.

Задание 2

Изменить разработанный ранее сессионный компонент с учётом требований EJB 3.

Для получения ссылок на сущностные компоненты следует использовать механизм внедрения зависимостей.

Задание 3

Изменить разработанные ранее сервлеты с учётом требований EJB 3.

Для получения ссылок на сессионные компоненты следует использовать механизм внедрения зависимостей.

Лабораторная работа №12

Entity Persistence

Темы для предварительного изучения

Синтаксис языка Java

EJB 3

Session Beans

JNDI

JMS

MDB

Dependency Injection

Servlets

Entity Persistence

Задание на лабораторную работу

Модифицировать разработанное Enterprise-приложение с применением технологии Entity Persistence для доступа к данным.

Задание 1

Заменить классы JavaBeans, описывающие состояние сущностей, аналогичными классами сущностей в соответствии с технологией Entity Persistence. При изменении состояния сущности объекты основных сущностей должны посылать сообщения в целевой объект JMS.

Задание 2

Разработать класс сущностей для работы с таблицей, в которой отражаются изменения состояния сущностей.

Задание 3

Сконфигурировать модуль персистентности.

Следует добавить возможность развёртывания схемы в базе данных по модулю персистентности.

Сделать так, чтобы никакие другие классы, кроме перечисленных, не могли быть добавлены в схему.

Задание 4

В коде сессионного компонента заменить сущностные компоненты и обращения к ним на экземпляры классов сущностей и обращения к менеджеру персистентности.

Задание 5

В коде MDB-компонента заменить взаимодействие с базой данных через JDBC на работу с менеджером персистентности и классом соответствующей сущности.

Лабораторная работа №13

JSF

Темы для предварительного изучения

Синтаксис языка Java

EJB 3

Session Beans

JSF

Servlets

Entity Persistence

Задание на лабораторную работу

Модифицировать Web-модуль разработанного Enterprise-приложения с применением технологии JSF для формирования отклика для клиента.

Задание 1

Определить виды и типы подложенных компонентов (managed beans), необходимые для JSF-страниц приложения. Обосновать свой выбор.

Задание 2

Реализовать дополнительные необходимые классы подложенных компонентов.

Задание 3

Изменить слой сервлетов таким образом, чтобы данные для представления на web-слое передавались через состояние подложенных компонентов.

Задание 4

Определить JSF-страницы приложения и правила навигации между ними. Описать правила навигации в соответствующих файлах.

Задание 5

Реализовать код JSF-страниц с применением одной из технологий визуальных компонентов JSF.

Лабораторная работа №14

Web Services

Темы для предварительного изучения

Синтаксис языка Java

EJB 3

Session Beans

Web Services

Задание на лабораторную работу

Выделить функции взаимодействия с базой данных как общедоступные web-сервисы. Разработать клиентское приложение для работы с этими web-сервисами.

Задание 1

На основе разработанного сессионного компонента без состояния создать публичный web-сервис в стиле SOA.

Задание 2

Изучить WSDL-файл сервиса.

Задание 3

Изменить клиентское приложение (последний раз использовалось в лабораторной работе 6) таким образом, чтобы получение и изменение данных производилось с помощью удалённого вызова web-сервиса на сервере.