

Лабораторная работа

Тема: Передача данных через параметры URL. Передача данных через HTML- форму. Основные принципы организации веб — приложения, использующего базы данных.

Цель работы: Освоить методы передачи данных в веб -среде.
Разработать простейшее веб — приложение ,
использующее базы данных.

Краткие теоретические сведения

Web-приложение состоит из отдельных страниц. Передача данных между страницами занимает важное место в работе приложения. Существует два универсальных (не зависящих от технологии программирования) способа передачи данных между веб — страницами :

1. В параметрах URL
2. В переменных полей формы

Передача параметров в URL

Этот способ основан на использовании гиперссылок, содержащих специальные параметры URL. URL (*Universal Resource Locator*) – универсальный адрес ресурса. Спецификация HTTP позволяет завершать URL списком параметров, используя следующий формат:

http://хост:порт/путь_к_файлу_от_корневого_каталога_веб_сервера?параметр1=значение1&параметр2=значение2&...

При щелчке на гиперссылке, содержащей параметры URL, они автоматически передаются документу, указанному в ссылке, и становятся доступны в нем как переменные URL. Следующий пример демонстрирует передачу параметров между шаблонами ColdFusion:

```
<cfset      x      =1 >  
<cfset      flag  =TRUE >
```

```
<cfoutput>
```

```
<a href="http://localhost:8500/newrelib.cfm?x=#x#&flag=#flag#">Нажмите здесь,  
чтобы посмотреть URL параметры</a>
```

```
</cfoutput>
```

В шаблоне *newrelib.cfm*, куда передаются параметры, их значения можно просмотреть следующим способом:

```
<cfoutput>
```

```
x:      #url.x#<br>
```

```
color:  #url.flag#
```

```
</cfoutput>
```

Передача данных в формах HTML

Это другой способ передачи данных между страницами. Основан на отправке данных методом HTTP post или get из полей формы расширения веб-сервера для дальнейшей обработки.

Форму можно поместить в любом месте тела HTML-документа или шаблона(скрипта, сценария), заключив ее элементы между открывающим и закрывающим тегами `<form ></form>`.

Форма имеет следующие атрибуты:

1. *Action* –указывается URL приложения, которое должно получить и обработать данные формы. Параметр обязательный.
2. *Method*- методы POST или GET, с помощью которых форма отправляет данные на обработку. Параметр обязательный.
3. *EncType* – метод кодировки данных формы для предотвращения искажения или повреждения при передаче. *Application/x-www-form-urlencoded* – формат по умолчанию, а *multipart/form-data* – используется при передаче файлов.

Большинство элементов определяется на форме с помощью тега `<input >`.

В этом теге только два обязательных атрибута *type* и *name*. *Name* определяет имя элемента, а *type* – его тип. Существуют следующие типы элементов:

`<input type="text">` - поле для ввода текста

`<input type="password">` - поле пароля для ввода пароля

`<input type="checkbox" value="">` - флажки дают возможность выбирать опциональные значения на форме, делая множественный выбор. Опциональное значение задается в параметре *value*.

`<input type="file">` -позволяет пользователю выбрать файл из файловой системе своего компьютера и отправить его на сервер.

`<input type="radio" value="">` -радиокнопки, аналогичны флажкам, с той разницей, что из группы радиокнопок можно выбрать только один элемент. Опциональное значение задается в параметре *value*.

`<input type="hidden" value="">` - скрытые поля. Элементы не отображаются на форме. Служат для того, чтобы “пометить” форму и очень

полезны для выделения различных форм из совокупности переданных или сохраненных.

`<input type="image">` - пользовательская кнопка-изображение.

`<input type="reset">` - кнопка сброса значений формы.

`<input type="submit">` - кнопка отправки данных из формы на сервер.

На форме можно использовать такой элемент, как `<textarea>`, который создает область ввода текста. Обязательный атрибут `name`.

А так же такой элемент, как `<select> </select>` - список значений. Обязательный атрибут `name`. Поместив список пунктов при помощи тегов `<option></option>`, можно создать выпадающее меню, в котором можно выбирать необходимые пункты. Что бы разрешить множественный выбор в списке, необходимо указать атрибут `multiple`. Значение `size` указывает, сколько вариантов выбора в меню будет отображаться одновременно (`size = "1"` – выпадающее меню; `size` со значением, большим 1 - прокручиваемый список). Что бы изначально отметить какой – то элемент списка, в соответствующем теге `<option>` необходимо установить атрибут `selected`.

Создание веб — интерфейса, использующего базы данных

В данной лабораторной работе рассматривается разработка веб — интерфейса с помощью технологии Cold Fusion. Эта среда разработки веб — приложений состоит из следующих важных компонент:

1. **Сервер приложений ColdFusion** обрабатывает весь код CFML-шаблонов, передаваемых веб-сервером, а затем возвращает серверу динамически сгенерированные страницы для передачи клиентскому приложению. После настройки выполняется в фоновом режиме.
2. **ColdFusion Studio, Macromedia Dreamweaver, HomeSite**-интегрированные среды программирования для сервера приложений ColdFusion. Приложения можно создавать с помощью любого текстового редактора. Но выше названные приложения предоставляют удобные инструменты для разработки.
3. **Шаблоны CFML** – это страницы, содержащие наряду с HTML-тегами CFML-теги. HTML-теги определяют, как нужно отображать данные, а CFML-теги определяют, какие данные нужно отображать.
4. **Веб-сервер.** В ColdFusion встроен веб-сервер для локальной разработки приложений. Настроен на порт 8500. Каталог сервера является папка `...\CFusionMX\wwwroot`.
5. Через браузер к ресурсам сервера необходимо обращаться следующим образом:

<http://localhost:8500/>

<http://127.0.0.1:8500/>

6. **Администратор ColdFusion**- приложение, отвечающее за настройку и администрирование сервера приложений ColdFusion. Конфигурируется через браузер:

<http://127.0.0.1:8500/CFIDE/administrator/>

7. **Источники данных** — это объекты сервера приложений ColdFusion, позволяющие организовать взаимодействие с различными источниками данных, такими как базы данных, коллекции Verity, почтовые сервера POP3 и SMTP, серверы FTP и HTTP и т. д. Источник данных требует предварительной настройки с помощью приложения Администратор ColdFusion. Которая заключается в указании параметров подключения к базе данных:

Data Source Name – имя создаваемого источника данных.

Driver- выбирается драйвер к соответствующей СУБД

DataBase – база данных, к которой производится подключение.

Server – сервер баз данных, к которому будет производиться обращение

Username и Password – учетная запись и пароль на сервере, от имени которой будет производиться аутентификация в системе безопасности СУБД

На рис.1и рис.2 иллюстрируется регистрация источник данных на сервере приложений Cold Fusion.

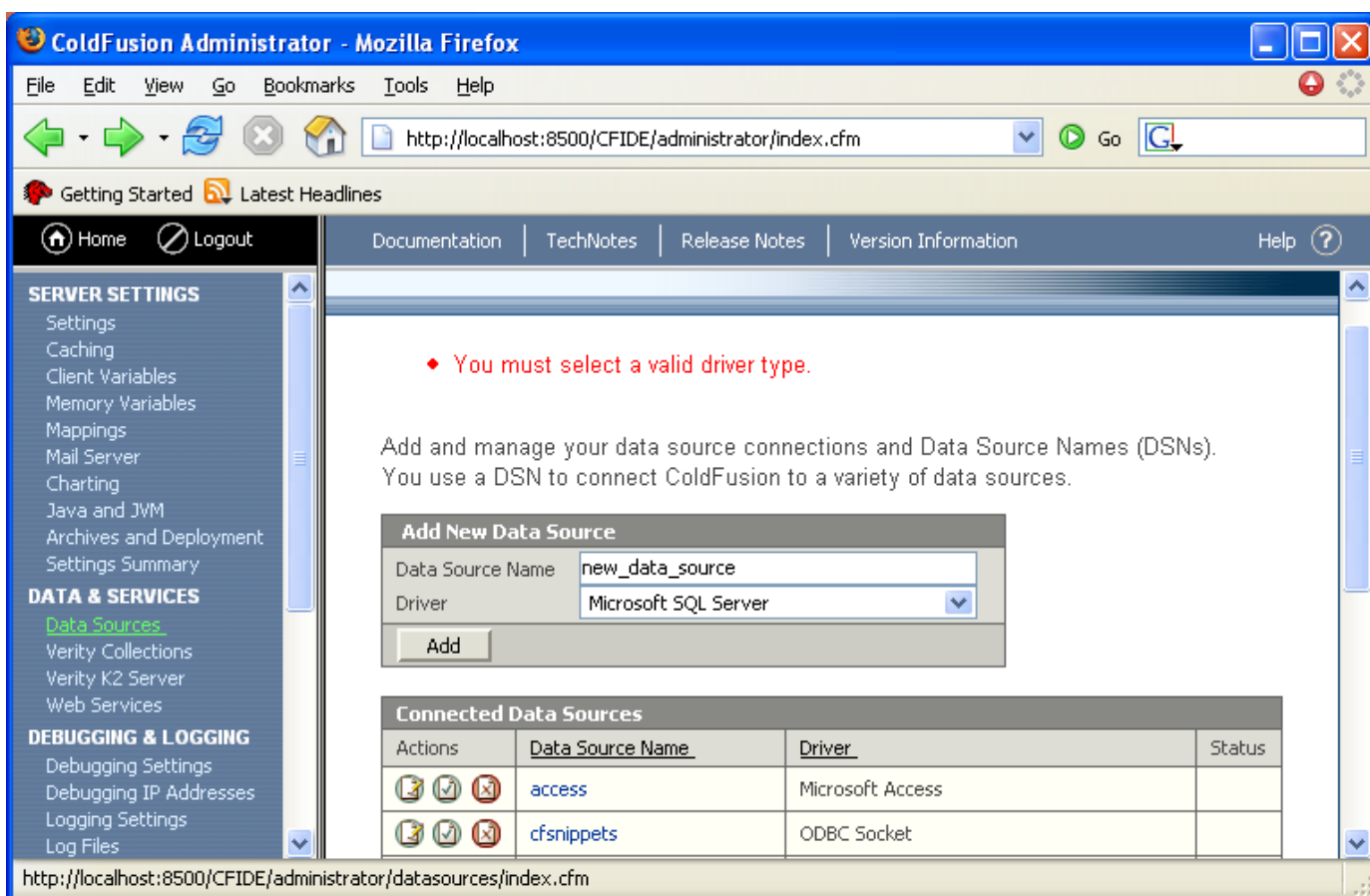


рис.1

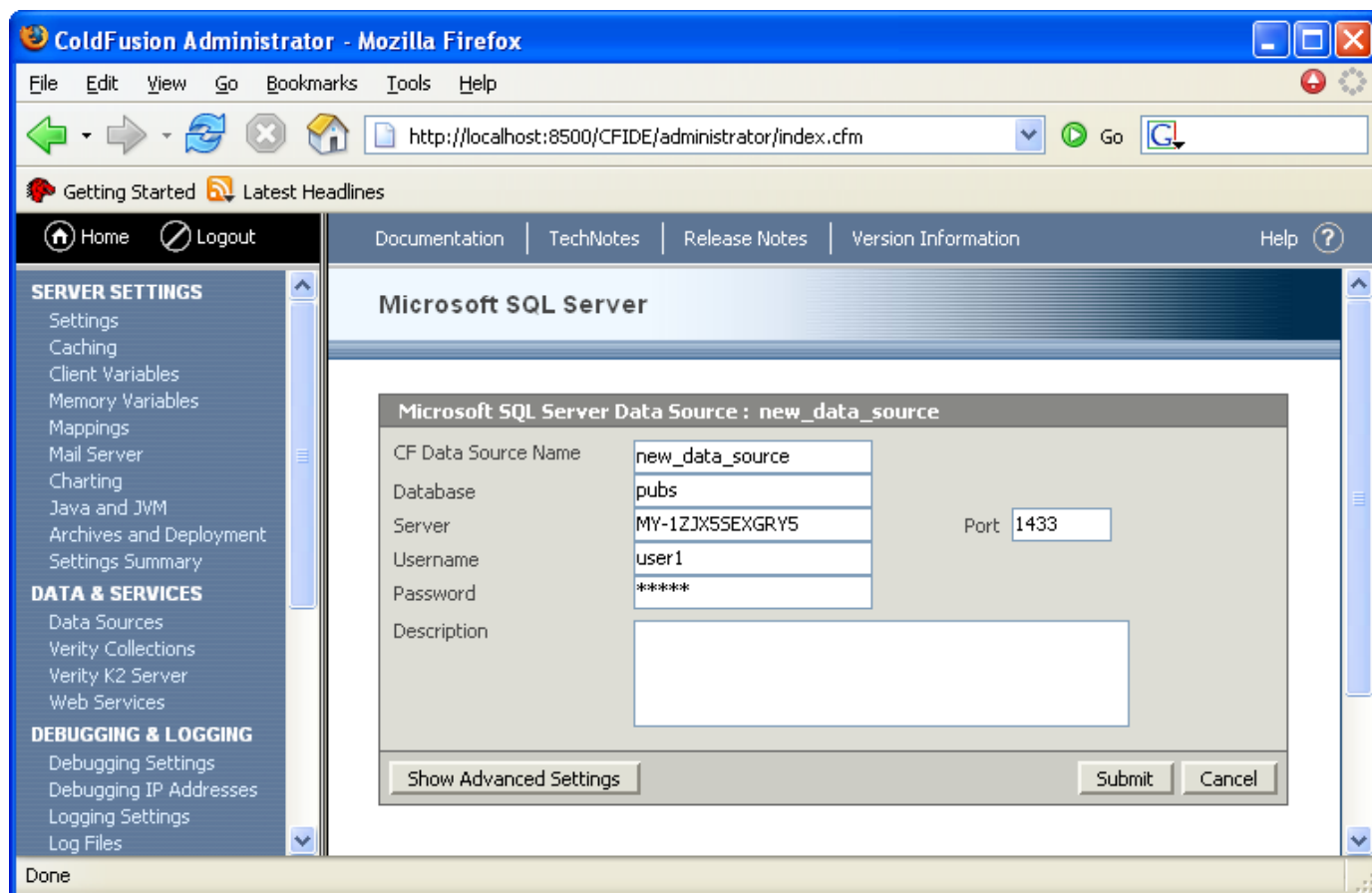


рис.2

Обмен данными с базами данных осуществляется через источник данных путем передачи запросов SQL и получения от СУБД результирующих данных с помощью тега CFQUERY. Упрощенный синтаксис тега представляет собой следующее:

```
<cfquery name="my_query" datasource="my_data_source">
    Команда SQL
</cfquery>,
```

где name- имя, присваиваемое запросу и впоследствии результирующему набору данных

datasource- источник данных, к которому идет обращение

Возвращаются переменные:

Имя_результатирующего_набора.ColumnList – список полей таблицы, разделенных запятыми

Имя_результатирующего_набора.RecordCount- общее количество записей, возвращаемых запросом

Имя_результатирующего_набора.CurrentRow- текущая запись, обрабатываемая тегом CFOUTPUT

Результаты запроса выводятся при помощи тега CFOUTPUT.

```
<cfoutput query="my_query">
    #field1#<br>
    #field2#<br>
```

```
#field3#<br>
</cfoutput>
```

Где field1,field2,field3 имена полей запроса SQL. Имена переменных, заключенных между знаками #, заменяются в выходных данных значениями этих переменных.

Уточняющие запросы

Уточняющим запросом называется многоэтапный запрос, который на первом этапе извлекает из базы относительно широкий набор записей в виде списка гиперссылок. Затем при выборе той или иной гиперссылки вызывается другой запрос, который проводит выборку на основании параметров URL, переданных по выбранной гиперссылке. И так далее, пока не будет достигнут желаемый уровень точности

Ход работы

1. Создать шаблон, предающий данные в другой шаблон через URL, в котором они будут сохранены и после этого считаны из БД так, что в браузере должны публиковаться примерно следующие данные:

Персональные данные

Имя: Kelly Smith
Должность: Vice President
Отдел: M&M
Телефон: 8(0297)11-23-45
Дата: 08/15/2000
Email: Kelly@example.com
Комментарий: I get up at 7 o'clock.
I wash my hands, my face and clean my teeth.
I air my room and make the bed.
Then I have my breakfast.

2. Создать шаблон personal.cfm, передающий через переменные полей формы данные в другой шаблон savepersonal.cfm, в котором они должны быть сохранены в БД, считаны из БД и выведены на экран:

personal.cfm

Имя:	<input type="text" value="Kelly Smith"/>
Должность:	<input type="text" value="Vice President"/>
Пол:	<input checked="" type="radio"/> Woman <input type="radio"/> Man
Отдел:	<input type="text" value="M&M"/>

E-mail:

Телефон:

Комментарий:

I get u
 I was
 I air m
 Then

savepersonal.cfm

Имя: Kelly Smith
 Должность: Vice President
 Отдел: M&M
 Телефон: 8(0297)11-23-45
 Пол: Woman
 Email: Kelly@example.com
 Комментарий: I get up at 7 o'clock. I wash my hands, my face and clean my teeth. I air my room and make the bed. Then I have my breakfast.

3. Создать базу данных, содержащую персональные данные о сотрудниках и отделах организации. Создать cfm – шаблон, выводящий актуальный список о сотрудниках организации. В интерфейсе предусмотреть кнопки ДОБАВИТЬ, ИЗМЕНИТЬ, УДАЛИТЬ. С помощью которых можно соответственно добавить данные о сотруднике в базу данных, изменить данные выбранного сотрудника и удалить данные выбранного сотрудника из базы данных. Интерфейс может быть аналогичен следующему:



4. С помощью уточняющих запросов отразить организационную структуру БГТУ: Список факультетов – Список Курсов – Список Групп – Список

Студентов – Информация о выбранном студенте. Все данные для уточняющих запросов должна храниться в БД.

Контрольные вопросы

1. Опишите основные компоненты технологии разработки веб - приложений ColdFusion.
2. Каким тегом осуществляется выполнение запросов к базам данных в ColdFusion?
3. Каким тегом осуществляется выполнение вывода данных в ColdFusion?

Литература

1. Пол Нильсен . SQL Server 2005. Библия пользователя -Вильямс ,2008
2. Брук-Билсон Р. Программирование на Cold Fusion. Для профессионалов. - СПб: Питер,2002