**Лабораторна робота 5\_1**

**ФОРМИ**

Завдання:

Створіть форму для даних, що входять до залікової книжки студента. Вона повинна містити усі поля, які знаходяться в книжці. Емблему університету визначити як малюнок фону.

Розв’язок:



**Контрольні завдання і запитання**

1. Дайте визначення форми, ярлика і контролю?

2. Визначте типи головного контролю.

3. Особливості і приклади використання елементу SELECT.

4. Назвіть можливі використання елементу OPTGROUP.

5. За що відповідають елементи LEGEND і FIELDSET

6. Дайте приклад використання елементу ЯРЛИК.

7. Чим є головні елементи XForm?

Відповіді:

1. Дайте визначення форми, ярлика і контролю?

Форма XHTML - частина документу, що містить нормальний зміст, розмітки, спеціальні елементи контролю (перемикачі, radio кнопки, меню, і т.п.) і ярлики. Користувачі загалом "завершують" форму, змінюючи контроль (текст, що вводиться, вибір пунктів меню, і т.п.), перед формою для обробки (наприклад, до веб-сервера, до поштового сервера, і т.п.)

Тег <label> встановлює зв'язок між певною міткою, в якості якої зазвичай виступає текст, і елементом форми (<input>, <select>, <textarea>). Такий зв'язок необхідна, щоб змінювати значення елементів форми при натисканні курсором миші на текст. Крім того, за допомогою <label> можна встановлювати гарячі клавіші на клавіатурі і переходити на активний елемент подібно посиланнях.

Існує два способи зв'язування об'єкта і мітки. Перший полягає у використанні ідентифікатора id всередині елемента форми і надання його імені в якості атрибута for тега <label>. При другому способі елемент форми поміщається всередину контейнера <label>.

Кожен контроль має,як початкове значення, так і поточне значення, кожне з яких - рядки символів. Проте, початкове значення елементу TEXTAREA дає його вміст і початкове значення елементу OBJECT у формі визначає об'єктне виконання (тобто, лежить за межами можливості цієї специфікації).

"Поточне значення" контролю уперше встановлюється до початкового значення. Після того, поточне значення , можливо, змінюється. Початкове значення контролю не змінюється. Отже, коли форма знову встановлена, поточне значення кожного контролю знову встановлюється до його початкового значення. Якщо контроль не має початкового значення, якщо елементи контролю не мають початковго значення, єфект такого контроль не визначений.

Коли форма передається для обробки, деякі елементи управління мають свої назви у поєднанні з їх поточними значеннями.Цей контроль, називаються успішним контролем.

2. Визначте типи головного контролю.

**buttons**

Автори можуть створювати три види кнопок :

* кнопки відправлення,що представляє форму. Форма, можливо, містить більш ніж одну кнопку.
* кнопки повторення: коли її активізують, кнопка встановлює початкові значення управління.
* кнопки натискування: кожна кнопка може зв'язати скрипти з елементами атрибутів подій. Коли подія відбувається (наприклад, користувач натиснув на кнопку.) запускається відповідний сценарій.

Автори створюють кнопки з елементом BUTTON або елементом INPUT.

**Примітка.** Треба відзначити, що елемент BUTTON пропонує більші можливості візуалізації, ніж елемент INPUT.

**checkboxes**

Перемикачі (і радіокнопки) -є on/off вимикачами, які можуть перемикатися користувачем. Вимикач - оn, коли контрольний елемент змінив встановлений атрибут. Коли форма представлена, тільки on перемикачами, контроль може стати успішним.

Декілька перемикачів у формі можуть розділятись тим самим контрольним ім'ям. Отже перемикачі дозволяють користувачам вибирати декілька значень для тієї ж властивості. Елемент INPUT використовується для створення прапорця контролю.

**radio buttons**

Radio кнопки подібні до перемикачів за винятком того, що, коли деяка кількість розділяє те ж саме ім'я, вони взаємовиключні. Коли "на" одному перемкнуто, усе інші з тим же ім'ям вимкнені. Елемент INPUT використаний, щоб створити кнопку радіоконтролю.

**menus**

Меню пропонують користувачам опції вибору. Елемент SELECT створює меню, в комбінації з OPTGROUP і елементами вибору.

**text input**

Автори можуть створити два види контролю вхідного тексту. Елемент INPUT створює single-line вхідний контроль , а елемент TEXTAREA створює multi-line вхідний контроль. У обох випадках вхідний текст стає поточним значенням контролю.

**file select**

Цей контрольний тип дозволяє користувачеві вибирати файли таким чином, що їх вміст, можливо, представляється з формою. Елемент INPUT використаний, щоб створити файл вибору контролю.

**hidden controls**

Автори можуть створюювати контроль, які не наданий, але його значення представляються з формою. Автори загалом використовують цей контрольний тип, щоб зберігати інформацію між сервером обміну. Елемент INPUT використовується, щоб створити прихований контроль.

**object controls**

Автори, можливо, вставляють об'єкти у форми таке, що і представляють цінності разом з іншим контролем. Автори створюють контроль з елементом OBJECT.

3. Особливості і приклади використання елементу SELECT.

SELECT елемент створює меню. Кожен вибір пропонованих меню представлено опцією елемента. SELECT елемент повинен містити щонайменше одну опцію елемента.

Елемент OPTGROUP дозволяє авторам логічно згрупувати вибір. Це особливо корисно, коли користувач повинен вибрати з довгого списку вибору; групи пов'язаних альтернатив легше пам'ятати, ніж єдиний довгий список вибору.

Користувач може заздалегідь вибирати більше альтернатив. Треба визначити які альтернативи заздалегідь вибираються, як зазначено нижче:

* Якщо ніякий елемент SELECT не має відібраного набору властивості, поведінка для вибору невизначена. Відколи поведінка відрізняється, автори повинні гарантувати, що кожне меню включає невиконання заздалегідь відібраної опції.
* Якщо один елемент SELECT має відібраний набір властивості, його треба заздалегідь вибрати.
* Якщо елемент SELECT має багаторазовий набір властивостей і більш ніж один елемент, SELECT має відібраний набір властивостей, їх треба заздалегідь вибрати.
* Вважаєтья помилкою, якщо більш ніж один елемент SELECT має відібраний набір властивостей і елемент SELECT не має багаторазового набору властивостей.

Вибираючи меню, користувачі повинні використовувати значення властивості ярлика елементу OPTION. Якщо ця властивість не вказана, користувачі повинні використовувати вміст елементу OPTION.

<FORM action="http://somesite.com/prog/component-select" method="post">

<P>

<SELECT multiple size="4" name="component-select">

<OPTION selected value="Component\_1\_a">Component\_1</OPTION>

<OPTION selected value="Component\_1\_b">Component\_2</OPTION>

<OPTION>Component\_3</OPTION>

<OPTION>Component\_4</OPTION>

<OPTION>Component\_5</OPTION>

<OPTION>Component\_6</OPTION>

<OPTION>Component\_7</OPTION>

</SELECT>

<INPUT type="submit" value="Send"><INPUT type="reset">

</P>

</FORM>

4. Назвіть можливі використання елементу OPTGROUP.

Тег <optgroup> представляє собою контейнер, усередині якого розташовуються теги <option> об'єднані в одну групу. Особливістю тега <optgroup> є те, що він не виділяється як звичайний елемент списку, акцентується за допомогою жирного накреслення, а всі елементи, що входять до цей контейнер, зміщуються вправо від свого початкового положення.

**<select>**

**<optgroup** label="..."**>**

**<option>**...**</option>**

**</optgroup>**

**</select>**

5. Елемент FIELDSET дозволяє авторам групувати тематично пов'язані контролі і ярлики..

Елемент LEGEND дозволяє авторам призначати заголовок FIELDSET. Це покращує доступність, коли FIELDSET виявляється не візуально.

6. Дайте приклад використання елементу ЯРЛИК.

<LABEL for="email">email: </LABEL>

<INPUT type="text" id="email">

7. Чим є головні елементи XForm?

Фактично, XForms поділив форми на три частини :

* Мета
* Збір даних
* Представлення аранжування форми контролю на екрані
* Реєстраційна інформація даних

XForms складаються з окремих секцій, які визначають, що форма робить і яке це має бути зображено. Це означає, що ви можете використовувати ту ж форму на браузерах і переносних пристроях.