**ОРГАНИЗАЦИЯ ИНТЕРФЕЙСА**

**ДИАЛОГОВОЙ СИСТЕМЫ**

РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

*Диалоговый пользовательский интерфейс* с Базой знаний на Web-страницах информационной системы представляет собой использование интерактивных всплывающих диалоговых окон по запросам пользователя на Web-страницах. Для создания интерактивных сайтов чаще всего используется библиотека JavaScript **jQuery**.

Библиотека **jQuery** реализует архитектуру на основе концепций, положенных в основу языка разметки **HTML** и языка каскадных таблиц стилей **CSS,** со следующими базовыми возможностями:

* может обращаться к любому элементу объектной модели Web-документа **DOM** и предлагает механизм селекторов;
* может работать с событиями;
* осуществляет различные визуальные эффекты с изменением внешнего вида страницы с использованием **CSS**;
* имеет различные JavaScript**-плагины**, предназначенные для создания элементов пользовательских интерфейсов, например, диалоговых окон.

**СТРУКТУРА ДИАЛОГОВОЙ WEB-СТРАНИЦЫ**

Для реализации на Web-странице диалога с Базой знаний на основе интерфейса с диалоговыми окнами необходимо:

1. В **папку сайта** добавить файлы:

* библиотеку jQuery – файл **jquery.js,**
* файл разработанного в ЛР по Теме 10 программного кодана языке JavaScript для обработки диалога, например, с именем **dialog.js;**
* файлвнешней таблицы разработанных для диалогаCSS-стилей, сохраненного, например, с именем **style.css**.

1. В **HTML**-разметку диалоговой Web-страницы добавить следующие элементы:

* строку с тегом <script> для подключения файла библиотеки jQuery **jquery.js;**
* строку с тегом <script> для подключения внешнего файла с программным кодом JavaScript **dialog.js;**
* строку с тегом <link> для подключения внешней таблицы CSS-стилей из файла **style.css**
* в теге <body> загрузить оператором **onLoad()** созданную дополнительно функцию организации интерфейса для диалогового окна для Web-страницы, например, с именем **dialog\_window().**

Общая структура Web-страницы с диалогом может выглядеть следующим образом:

**<html lang="ru" xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<title>Диалог с базой знаний</title>**

**<meta charset="utf-8" />**

**<script src="jquery.js"></script>**

**<script src="dialog.js"></script>**

**<link href="style.css" type="text/css" rel="stylesheet"/>**

**</head>**

**<body onLoad="dialog\_window()">**

**…**

**</body>**

**</html>**

**СОЗДАНИЕ ИНТЕРФЕЙСА ДИАЛОГОВОГО ОКНА**

В программный код JS-файла семантического генератора ответов из базы знаний **dialog.js** необходимо добавить код для создания диалогового окна и программных модулей для функционирования его интерфейса с использованием файла библиотеки **jquery.js**.

***ШАБЛОН***

*ДЛЯ СОЗДАНИЯ КОДА ДИАЛОГОВОГО ОКНА И ФУНКЦИОНАЛЬНЫХ МОДУЛЕЙ ЕГО ИНТЕРФЕЙСА, А ТАКЖЕ ИСПОЛЬЗУЕМЫХ CSS-СТИЛЕЙ ДЛЯ ЕГО ОФОРМЛЕНИЯ.*

Вводим переменную, используемую для активизации диалога

**var dialogOn = false;**

ОПРЕДЕЛЯЕМ ФУНКЦИЮ ФОРМИРОВАНИЯ ДИАЛОГОВОГО ОКНА:

**function dialog\_window(){**

Добавляем на страницу диалоговое окно как новый раздел *div* с идентификатором

*dialog*, задаем ему CSS-стиль *dialog* и надвигаем его левое поле на страницу на 25px:

**document.body.innerHTML+=**

**"<div id='dialog' class='dialog' style='margin-left:-25px;'>"+**

Здесь используется внешний CSS-стиль:

***.dialog{position:fixed;top:50%;left:100%;width:300px;height:430px;margin-left:-25px; margin-top:-215px;padding:10px;border:#000 2px solid;border-radius:10px;background:#fff;color:#000;z-index:1500;font-size:12pt;}***

Добавляем в раздел диалогового окна новый раздел со строкой *«Нажми, чтобы спросить!»*, задаем ему CSS-стиль *label* и команду на выполнение функции выдвижения диалогового окна *toggleDialog()* щелчком мыши:

**"<div class='label' onclick='toggleDialog()'>Нажми, чтобы спросить!</div>"+**

Здесь используется внешний CSS-стиль:

***.dialog .label{transform:rotate(270deg);width:430px;height:20px;text-align:center;overflow:hidden;display:inline-block;margin-left:-71%;margin-top:68%;cursor:pointer;}***

Добавляем также в раздел диалогового окна новый раздел со строкой *«История»* изадаем ему CSS-стиль *header*

**"<div class='header'>История:</div>"+**

Здесь используется внешний CSS-стиль:

***.dialog .header{ text-align:center;margin-top:-200px;}***

Добавляем в раздел диалогового окна еще один новый раздел,задаем емуему CSS-стиль *history* и задаем ему идентификатор *history***:**

**"<div class='history' id='history'></div>"+**

Здесь используется внешний CSS-стиль:

***dialog .history{height: 300px;overflow-x:hidden;overflow-y:scroll;}***

Добавляем в диалоговое окно последний новый раздел со CSS-стилем оформления*question* в видеполя формы с начальным текстом,исчезающим при щелчке мышью и идентификатором *Qdialog,* а также кнопкой *«Спросить»* с командой для запуска функции *ask()*

**"<div class='question'><input id='Qdialog' placeholder='Введите вопрос'/><br>"+**

**"<button onclick='ask(\"Qdialog\")'>Спросить</button></div>"+**

Здесь используется внешний CSS-стиль:

***.dialog .history .question{margin-left:40px;background:#99f;}***

Наконец, закрываем общий блок для диалогового окна *div*

**"</div>";**

СОЗДАНИЕ ОКНА ДЛЯ УВЕЛИЧЕНИЯ ИЗОБРАЖЕНИЯ В ОТВЕТЕ С РИСУНКОМ:

Создаем в теле *body* диалоговой страницы новый внутренний элемент с добавлением нового блока *div* с внешним CSS-стилем *imgalert*и внутренним CSS-стилем *display:none* (вначале этот раздел скрыт):

**document.body.innerHTML+=**

**"<div id='imgalert' style='display:none'>"+**

Здесь используется внешний CSS-стиль:

**#imgalert{display: block;width: 100%;height: 100%;position: fixed;margin: 0;**

**padding:0;z-index:2000;}**

Добавляем в этот раздел новый раздел *div* с CSS-стилем *bg* и скрытием всего предыдущего раздела после щелчка мышью по фонуфункцией *hide()***:**

**"<div class='bg' onclick='hide(\"imgalert\")'>&nbsp;</div>"+**

Здесь используется внешний CSS-стиль:

**.bg{position:fixed;top:0;left:0;width:100%;height:100%;background:#000; opacity:0.5;z-index:100;cursor: pointer;}**

Добавляем на страницу рисунок с CSS-стилем *img\_in\_alert*

**"<img id='img\_in\_alert' src='' />"+**

Здесь используется внешний CSS-стиль:

**#img\_in\_alert{display: block;max-width: 100%;max-height: 100%;position:relative; margin: auto;z-index:2000;background: white;border: 2px solid black;}**

Закрываем общий раздел для крупного плана изображений и тело функции *dialog\_window()*

**"</div>";**

**}**

ФУНКЦИЯ ПЛАВНОГО ПОКАЗА-СКРЫТИЯ ДИАЛОГОВОГО ОКНА

**function toggleDialog(){**

**if(dialogOn){** //анимация закрытия диалогового окна

**$("#dialog").animate({"margin-left":"-25px"}, 1000, function() {});**

**dialogOn=false;**

**}**

**else{** //анимация открытия диалогового окна

**$("#dialog").animate({"margin-left":"-300px"}, 1000, function() {});**

**dialogOn=true;**

**clearInterval(timer);**

**}**

**}**

СТРУКТУРА ФУНКЦИИ ОБРАБОТКИ ВОПРОСА В ДИАЛОГОВОМ ОКНЕ

**function ask(questionInput){**

Переменная для считывания содержания окна ввода вопроса**:**

**var question=document.getElementById(questionInput).value;**

Активизация диалога**:**

**dialogOn=true;**

Создание поля для ввода вопроса в новом блоке <div>:

**var newDiv=document.createElement("div");**

Задание класса стиля оформления созданного блока для вопроса:

**newDiv.className='question';**

Заполнение созданного блока текстом вопроса:

**newDiv.innerHTML=question;**

Добавление созданного блока в блок диалога**:**

**document.getElementById("history").appendChild(newDiv);**

Создание поля для вывода ответа в новом блоке <div>:

**newDiv=document.createElement("div");**

Задание класса стиля оформления созданного блока для ответа:

**newDiv.className='answer';**

Получение ответа на вопрос и заполнение им созданного блока ответа:

**newDiv.innerHTML=getAnswer(question);**

Добавление созданного блока в блок диалога:

**document.getElementById("history").appendChild(newDiv);**

Прокрутка истории в самый низ:

**document.getElementById("history").scrollTop =   
 document.getElementById("history").scrollHeight;**

Очистка текстового поля для ввода нового вопроса и конец тела функции:

**document.getElementById(questionInput).value="";**

**}**

В этой функции используются следующие внешние CSS-стили:

**.dialog .history .question, .dialog .history .answer{ margin:5px;width:78%;**

**border-radius: 5px;padding:5px;}**

**.dialog .history .answer{margin-right:40px;background:#9ff;}**

**.dialog input, .dialog button, .dialog .history{ margin:2px 25px; width:85%;**

**border:1px #666 solid;border-radius: 5px;font-size:14px;}**

.**dialog .history .answer img {cursor:pointer; display:block; max-width:180px;}**

СОЗДАНИЕ ЭФФЕКТА УВЕЛИЧЕНИЯ РАЗМЕРА КАРТИНКИ В ОТВЕТЕ

Для обработки эффекта увеличения размера картинки, выведенной при ответе в диалоговое окно, необходимо добавить в конец функции получения ответа **getAnswer(question)** в переменную с полученным ответом **answer**замену картинки, выведенной в диалоговое окно, на картинку, увеличенную в размере с выполнением функции **zoom()** при щелчке мышью по картинке в окне с ответом:

**function getAnswer(question){**

**...**

**if(!result)answer = "Ответ не найден. <br/>";**

//если ответ есть - добавляем увеличение картинок

***else***

***answer = answer.replace("<img ","<img onclick='zoom(this.src)'");***

***return answer;***

**}**

Кроме того, в общий программный код в файле *dialog.js* необходимо добавить соответствующие функции, выполнения увеличения картинки *zoom()* и скрытия картинки при щелчке мышью по фону *hide().*

ФУНКЦИЯ УВЕЛИЧЕНИЯ РАЗМЕРА КАРТИНКИ В ОТВЕТЕ:

***function zoom(src){***

***document.getElementById("img\_in\_alert").src=src;***

***$("#imgalert").css({"display":"block"});***

***}***

ФУНКЦИЯ СКРЫТИЯ КАРТИНКИ ПРИ ЩЕЛЧКЕ НА ФОН:

***function hide(imgalert){***

***$("#imgalert").css({"display":"none"});***

***}***