# Лабораторная работа: Управление событиями и окнами на jQuery

## Цель работы:

Научиться управлять событиями и взаимодействовать с окнами на веб-странице с использованием библиотеки jQuery.

## Описание задания:

Создать веб-страницу с элементами, которые реагируют на события, такие как клик, наведение мыши, отправка формы, а также работа с диалоговыми окнами. В ходе выполнения работы вы освоите использование библиотеки jQuery для обработки событий и управления окнами.

## Инструменты:

- Текстовый редактор (например, Visual Studio Code)
- Браузер (Google Chrome, Firefox и т.д.)
- Подключенная библиотека jQuery

## Подключение jQuery:

Для работы с jQuery необходимо подключить библиотеку. В HTML-файл добавьте ссылку на jQuery, например:

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

## Ход работы:

### 1. Создание HTML-файла:

Создайте новый файл с расширением `.html`, например, `index.html`.
Добавьте структуру HTML и подключите jQuery:
<!DOCTYPE html>
<html lang="ru">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Управление событиями и окнами на jQuery</title>
 <link rel="stylesheet" href="styles.css">
 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
 <script src="script.js"></script>
</head>
<body>
 <!-- Контент страницы будет здесь -->
</body>
</html>

### 2. Добавление элементов для работы с событиями и окнами:

Создайте несколько элементов для практики управления событиями и окнами:
<button id="clickButton">Нажми меня</button>
<div id="hoverDiv">Наведи на меня</div>
<input type="text" id="inputField" placeholder="Введите текст">
<form id="myForm">
 <input type="text" placeholder="Введите что-нибудь">
 <button type="submit">Отправить</button>
</form>
<button id="openDialog">Открыть окно</button>
<div id="dialog" style="display: none;">Это диалоговое окно!</div>
<p id="output"></p>

### 3. Создание JavaScript-файла:

Создайте файл `script.js` в той же папке, что и `index.html`.

### 4. Добавление обработчика события клика с помощью jQuery:

В `script.js` добавьте код, который изменит текст кнопки после клика:
$(document).ready(function() {
 $("#clickButton").click(function() {
 $(this).text("Спасибо за клик!");
 });
});

### 5. Добавление обработчика событий для наведения мыши с помощью jQuery:

Используйте `mouseenter` и `mouseleave` для изменения стиля элемента при наведении и уходе мыши:
$("#hoverDiv").mouseenter(function() {
 $(this).css("background-color", "lightblue");
});
$("#hoverDiv").mouseleave(function() {
 $(this).css("background-color", "");
});

### 6. Обработка ввода текста в поле:

Используйте событие `keyup` для отображения текста в реальном времени:
$("#inputField").keyup(function() {
 $("#output").text("Вы ввели: " + $(this).val());
});

### 7. Добавление обработчика события для отправки формы:

Используйте событие `submit` для предотвращения стандартного действия и вывода сообщения:
$("#myForm").submit(function(event) {
 event.preventDefault();
 alert("Форма отправлена!");
});

### 8. Создание диалогового окна:

Добавьте функциональность для открытия и закрытия диалогового окна:
$("#openDialog").click(function() {
 $("#dialog").show();
});
$("#dialog").click(function() {
 $(this).hide();
});

### 9. Проверка работы:

Откройте `index.html` в браузере и проверьте работу всех событий и диалоговых окон:
- Клик по кнопке должен изменить её текст.
- Наведение на `div` должно изменить его цвет.
- Ввод текста в поле должен отображаться в `p` элементе.
- Отправка формы должна показывать сообщение.
- Диалоговое окно должно появляться по клику на кнопку и закрываться по клику на самом окне.

## Контрольные вопросы:

- Каковы преимущества использования jQuery для обработки событий по сравнению с чистым JavaScript?

- Как добавить обработчик события клика на элемент с помощью jQuery?

- Чем отличаются события `mouseenter` и `mouseleave` от `mouseover` и `mouseout`?

- Как управлять отображением элементов, таких как диалоговые окна, с помощью jQuery?

## Дополнительное задание:

Добавьте обработчики событий для двойного клика (dblclick) и для нажатия клавиши (keypress), чтобы больше узнать о взаимодействии с элементами с помощью jQuery.