**Project Criteria for Midterm Assessment**

**Project Requirements and Criteria**

You are required to create a comprehensive project that demonstrates their understanding of the key concepts covered in the course. The project should integrate various JavaScript elements, showcase coding proficiency, and demonstrate creativity in applying learned topics.

**Core Project Requirements**

1. **Project Theme and Concept**:
   * The project should be related to one or more of the following topics:
     + Introduction to JavaScript
     + Notification function and console log usage
     + Variables and data types
     + Conditional statements (if-else)
     + Loops (for, while, do-while)
     + Functions (declaration, expressions, arrow functions)
     + Objects and arrays
     + Browser events (mouse and keyboard interactions)
     + Editing documents (DOM manipulation)
     + Setting up and working with a local server
2. **Functionality and Features**:
   * Include interactive elements that utilize JavaScript notification functions and console logs.
   * Employ variables and data types effectively in different parts of the project.
   * Demonstrate conditional logic and looping structures to control the project’s flow.
   * Integrate objects and arrays to store and manage data.
   * Include functions for reusable code segments.
   * Incorporate browser event handling to respond to user interactions (e.g., mouse clicks, keyboard presses).
3. **Project Structure and Code Quality**:
   * Code should be well-organized and include appropriate comments explaining major sections.
   * Follow best practices for variable naming, indentation, and overall code readability.
   * Ensure that functions are modular and can be reused effectively.
4. **User Experience and Interface**:
   * The project should have a clear and intuitive user interface.
   * Demonstrate DOM manipulation to create dynamic content.
   * Include elements that showcase interactivity, such as buttons, forms, or animations.
5. **Documentation**:
   * Submit a document detailing the project’s objectives, main features, and how different JavaScript elements were utilized.
   * Provide sample use cases or a short user guide.

**Suggested Project Topics**

1. **Interactive To-Do List Application**:
   * Implement adding, editing, and deleting tasks.
   * Use objects and arrays to manage tasks.
   * Integrate browser notifications for task reminders.
2. **Simple Calculator or Conversion Tool**:
   * Create a basic calculator with arithmetic operations.
   * Implement a unit converter (e.g., temperature, distance).
   * Use functions and DOM events for interactive inputs.
3. **Interactive Quiz Game**:
   * Build a quiz with questions stored in an array of objects.
   * Use conditions and loops for question flow.
   * Display results and provide user feedback.
4. **Personal Budget Tracker**:
   * Track income and expenses using objects and arrays.
   * Include visual indicators for budget status.
   * Integrate user notifications for expense limits.
5. **Browser-Based Drawing Tool**:
   * Use mouse events to draw shapes or lines.
   * Implement color and brush size selection.
   * Manage drawings using arrays for undo and redo actions.
6. **Weather Dashboard**:
   * Create a dashboard that shows weather data.
   * Include notifications for severe weather conditions.
   * Demonstrate functions for data fetching and display.
7. **Dynamic Form with Validation**:
   * Build a form that validates user input (e.g., email, phone number).
   * Use DOM manipulation to show validation messages.
   * Include keyboard event handling for real-time feedback.
8. **Simple Game (e.g., Tic-Tac-Toe or Memory Game)**:
   * Implement game logic using conditions and loops.
   * Use arrays to track the game board state.
   * Provide a scoring system and reset option.

**Assessment Criteria**

1. **Completeness**: The project meets all outlined requirements.
2. **Functionality**: The project works as intended without significant errors.
3. **Code Quality**: Code is well-structured, commented, and follows best practices.
4. **Creativity and Innovation**: The project shows original ideas or unique implementation.
5. **User Experience**: The project has a user-friendly design and interface.
6. **Documentation**: Comprehensive and clear documentation accompanies the project.

**Additional Notes**

* Projects will be presented to the class, where students will explain the functionality and answer questions.

# Критерии к проекту для рубежного контроля

## Требования и критерии к проекту

Студенты должны создать комплексный проект, демонстрирующий их понимание ключевых концепций, изученных в курсе. Проект должен интегрировать различные элементы JavaScript, демонстрировать навыки кодирования и проявлять творческий подход к применению изученного материала.

### Основные требования к проекту

1. **Тема и концепция проекта**:
   * Проект должен быть связан с одной или несколькими из следующих тем:
     + Введение в JavaScript
     + Использование функций уведомления и console.log
     + Переменные и типы данных
     + Условные операторы (if-else)
     + Циклы (for, while, do-while)
     + Функции (объявления, выражения, стрелочные функции)
     + Объекты и массивы
     + События браузера (взаимодействия с мышью и клавиатурой)
     + Редактирование документа (манипуляции с DOM)
     + Настройка и работа с локальным сервером
2. **Функциональность и особенности**:
   * Включить интерактивные элементы, использующие функции уведомления и console.log.
   * Эффективно использовать переменные и типы данных в разных частях проекта.
   * Продемонстрировать использование условных операторов и циклов для управления логикой проекта.
   * Интегрировать объекты и массивы для хранения и управления данными.
   * Включить функции для создания повторно используемых блоков кода.
   * Реализовать обработку событий браузера для отклика на действия пользователя (например, клики мыши, нажатия клавиш).
3. **Структура проекта и качество кода**:
   * Код должен быть хорошо организован и содержать комментарии, поясняющие основные разделы.
   * Соблюдать лучшие практики именования переменных, отступов и читаемости кода.
   * Убедиться, что функции модульны и могут эффективно использоваться повторно.
4. **Пользовательский опыт и интерфейс**:
   * Проект должен иметь понятный и интуитивно понятный пользовательский интерфейс.
   * Продемонстрировать манипуляции с DOM для создания динамического контента.
   * Включить элементы, демонстрирующие интерактивность, такие как кнопки, формы или анимации.
5. **Документация**:
   * Подготовить документ, описывающий цели проекта, основные функции и использование различных элементов JavaScript.
   * Привести примеры использования или краткое руководство для пользователей.

### Предложенные темы для проектов

1. **Интерактивный список задач (To-Do List)**:
   * Реализовать добавление, редактирование и удаление задач.
   * Использовать объекты и массивы для управления задачами.
   * Интегрировать уведомления для напоминания о задачах.
2. **Простой калькулятор или инструмент для конверсий**:
   * Создать базовый калькулятор с арифметическими операциями.
   * Реализовать конвертер единиц (например, температуры, расстояния).
   * Использовать функции и события DOM для интерактивных вводов.
3. **Интерактивная викторина**:
   * Создать викторину с вопросами, хранящимися в массиве объектов.
   * Использовать условные операторы и циклы для управления ходом викторины.
   * Отображать результаты и предоставлять обратную связь пользователю.
4. **Персональный трекер бюджета**:
   * Отслеживать доходы и расходы с использованием объектов и массивов.
   * Включить визуальные индикаторы статуса бюджета.
   * Интегрировать уведомления при достижении лимитов расходов.
5. **Инструмент для рисования в браузере**:
   * Использовать события мыши для рисования форм или линий.
   * Реализовать выбор цвета и размера кисти.
   * Управлять рисунками с помощью массивов для функций отмены и повтора.
6. **Панель погоды**:
   * Создать панель, отображающую данные о погоде.
   * Включить уведомления о неблагоприятных погодных условиях.
   * Продемонстрировать функции для получения и отображения данных.
7. **Динамическая форма с валидацией**:
   * Создать форму, проверяющую вводимые данные (например, email, номер телефона).
   * Использовать манипуляции с DOM для отображения сообщений валидации.
   * Включить обработку событий клавиатуры для мгновенной обратной связи.
8. **Простая игра (например, крестики-нолики или игра на запоминание)**:
   * Реализовать логику игры с использованием условных операторов и циклов.
   * Использовать массивы для отслеживания состояния игрового поля.
   * Обеспечить систему подсчета очков и возможность перезапуска игры.

### Критерии оценки

1. **Полнота**: Проект соответствует всем указанным требованиям.
2. **Функциональность**: Проект работает как задумано, без существенных ошибок.
3. **Качество кода**: Код хорошо структурирован, содержит комментарии и следует лучшим практикам.
4. **Креативность и инновации**: Проект демонстрирует оригинальные идеи или уникальную реализацию.
5. **Пользовательский опыт**: Проект имеет удобный и понятный интерфейс.
6. **Документация**: Проект сопровождается подробной и понятной документацией.

### Дополнительные заметки

* Проекты будут представлены в классе, где студенты объяснят функциональность и ответят на вопросы.