

LP-II: Web Application Development

Assignment on *Kontra.js* Package

Q1) Illustrate the need of “*Kontra.js*” package.

Ans: In game development, managing sprite movement, collision detection, and rendering can be complex. **Kontra.js** simplifies this by providing built-in functions for creating 2D games, handling game loops, and managing sprites, allowing developers to focus on core game mechanics rather than low-level details.

1. The Problem with Pure Canvas API

In HTML5 game development, using the `<canvas>` element directly gives only basic drawing features like shapes and images. However:

- There is no built-in system for sprite management or game loops.
- Collision detection, movement, and input handling must be manually coded.
- Projects can quickly become cluttered, repetitive, and hard to scale.

This makes even small games time-consuming and difficult to maintain.

2. How *kontra.js* Solves This

The **kontra.js** package is designed to simplify and speed up 2D game development. It helps by:

- ✓ Providing built-in support for sprites, animations, and object movement.
- ✓ Managing game loops and user input efficiently.
- ✓ Allowing modular use — developers can pick only the components they need.

By handling these common tasks, *kontra.js* allows developers to focus more on gameplay rather than infrastructure.

3. Why It's Needed

Kontra.js is important because it addresses many common challenges in game development. It helps by:

- Eliminating the need to manually set up game loops, sprite handling, and input controls.
- Keeping the codebase cleaner, more modular, and easier to maintain.
- Saving development time, allowing quicker prototyping and faster iteration of ideas.
- Making it easier for both beginners and experienced developers to create high-quality 2D games without unnecessary complexity.

By handling the repetitive and technical parts behind the scenes, *kontra.js* lets developers concentrate more on designing gameplay and creative features.

4. Why It's Needed

In real-world projects, `kontra.js` makes a big difference by:

- Offering reliable asset loading for images, audio, and sprites.
- Managing multiple scenes and states easily.
- Delivering a smoother gaming experience through optimized loops and rendering.

Games built with `kontra.js` tend to be more stable, easier to debug, and faster to complete compared to games made with raw canvas coding.

5. Conclusion

Instead of manually recreating all the basics for every new game, **`kontra.js`** provides a lightweight, tested, and professional toolkit. It greatly improves the speed, quality, and organization of 2D game development in JavaScript. For anyone working with HTML5 canvas games, `kontra.js` is a highly valuable and efficient choice.

Q2) Implement small application which will make use of mentioned package (I/p, O/p)

Ans: This is a simple **Kontra.js-based game** where a red box moves automatically inside a canvas. The box bounces off the walls, reversing direction each time it hits an edge. The movement is continuous, and the box changes color randomly every time it bounces off a wall, creating a dynamic and engaging visual effect.

The game uses the **Kontra.js** library to handle canvas setup, sprite movement, and collision detection. It utilizes a **game loop** to continuously update and render the box's position and appearance

index.html

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Kontra.js Simple Game</title>

  <style>

    canvas {

      background: #ddd;

      display: block;

      margin: 30px auto;

      border: 2px solid black;

    }

  </style>

</head>

<body>

  <h2 style="text-align: center; font-family: 'Trebuchet MS', 'Lucida Sans Unicode', 'Lucida Grande', 'Lucida Sans', Arial, sans-serif;">Kontra.js Box Game</h2>

  <canvas id="gameCanvas" width="500" height="500"></canvas>
```

```
<!--kontra.js link -->
<script src="https://unpkg.com/kontra/kontra.min.js"></script>

<script src="game.js"></script>

</body>
</html>
```

game.js:

```
const { init, Sprite, GameLoop } = kontra;
let { canvas } = init('gameCanvas');

// Function to generate random color
function randomColor() {
  const colors = ['red', 'blue', 'green', 'orange', 'purple', 'pink', 'black', 'yellow'];
  return colors[Math.floor(Math.random() * colors.length)];
}

let player = Sprite({
  x: 250,
  y: 250,
  width: 30,
  height: 30,
  color: 'red',
  dx: 2,    // move 2 pixels right
  dy: 2,    // move 2 pixels down
  update: function () {
    this.advance();

    // Bounce on left/right walls
    if (this.x < 0 || this.x + this.width > canvas.width) {
      this.dx = -this.dx; // reverse direction
      this.color = randomColor(); // change color
    }

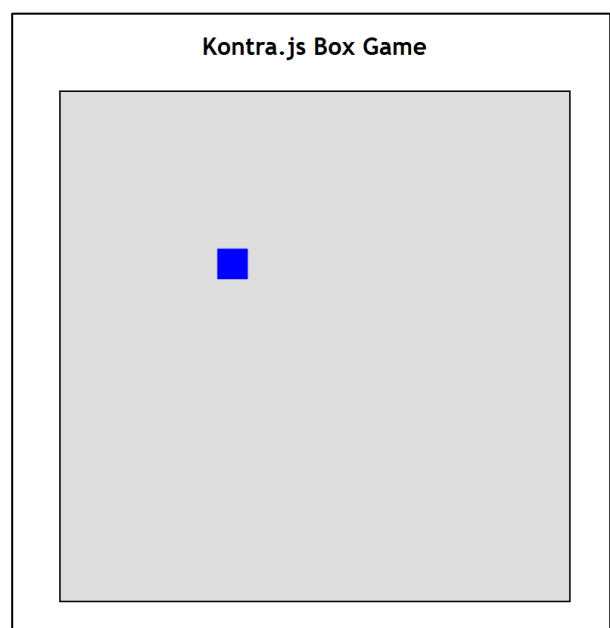
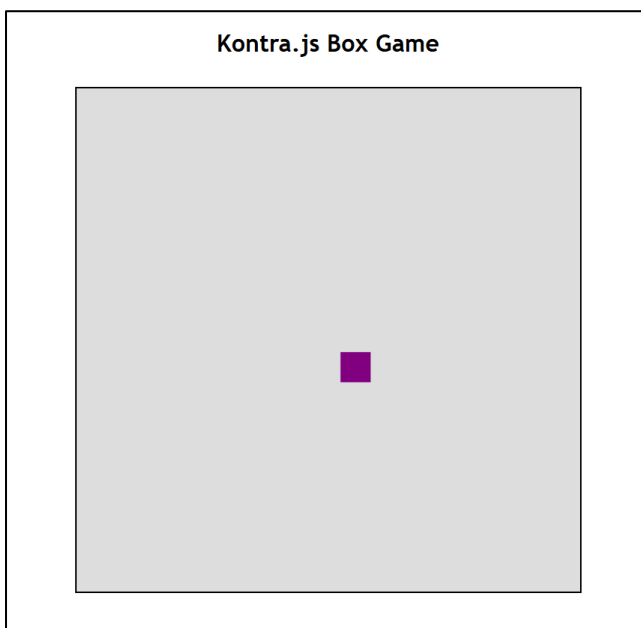
    // Bounce on top/bottom walls
    if (this.y < 0 || this.y + this.height > canvas.height) {
      this.dy = -this.dy; // reverse direction
      this.color = randomColor(); // change color
    }
  }
})
```

```
});

// Create the game loop
let loop = GameLoop({
  update: function () {
    player.update();
  },
  render: function () {
    player.render();
  }
});

// Start the game
loop.start();
```

OUTPUT:



Q3) Illustrate the need for code of ethics:

Ans: **The Need for a Code of Ethics in Web Application Development**

A **Code of Ethics** is crucial in web application development to ensure that the technology we build is safe, accessible, and respects user privacy. In an era where web applications are central to business, education, and personal life, the ethical guidelines governing their development help safeguard users and maintain trust.

1. Guiding Developer Behavior

A Code of Ethics provides web developers with clear standards on what constitutes responsible and ethical behavior. It helps developers navigate complex situations, such as balancing innovation with user security or making decisions that prioritize usability over convenience or profit.

2. Protecting User Privacy and Security

One of the most critical aspects of web application development is ensuring the safety and privacy of user data. Developers must adhere to ethical practices to safeguard sensitive information, preventing misuse, data breaches, or unauthorized access. A Code of Ethics ensures that these practices are not overlooked in the pursuit of efficiency or profitability.

3. Promoting Accessibility

Web developers are responsible for ensuring their applications are accessible to everyone, including people with disabilities. Ethical guidelines in web development emphasize the importance of creating applications that comply with accessibility standards (like WCAG) to ensure all users can interact with the web product.

4. Building User Trust

Web applications handle sensitive tasks like online banking, shopping, and communication. Following an ethical code helps establish transparency, build trust, and prevent unethical practices such as exploiting user data or misleading users. Trust is a key factor in user retention, especially for businesses that rely on their web platforms.

5. Encouraging Accountability in Development

A Code of Ethics encourages developers to be accountable for their work, from testing to deployment. This accountability ensures that developers take responsibility for bugs, security flaws, or design issues, which is vital for maintaining a high standard in web development and ensuring the application performs as expected.