

Clicker game in Vue



Hey Teks or future Teks, welcome to this new session of introduction to code!
Today we are dwelling into Web development, more specifically Vue.js framework one.

About Vue.js:

Vue.js is a **progressive JavaScript framework** used to build user interfaces and single-page applications (SPAs). It is designed to be **lightweight, easy to learn**, and **highly flexible**. Vue is particularly known for its **reactivity system** and its ability to scale from small projects to complex web applications.

Why use Vue.js:

✅ **Simple & Easy to Learn** – Vue has a gentle learning curve, making it beginner-friendly.

- ✓ **Reactive Data Binding** – The UI updates automatically when data changes.
- ✓ **Component-Based Architecture** – Applications can be structured using reusable components.
- ✓ **Lightweight & Fast** – Vue is smaller and faster than many other frameworks.
- ✓ **Great Ecosystem** – Includes Vue Router for navigation and Vuex (or Pinia) for state management.

Vue.js in the World of Work

Vue.js is widely used in the **tech industry**, especially for building modern web applications. It is favored by startups, large companies, and freelancers due to its simplicity, flexibility, and performance.

Getting Started with Vue.js

Introduction

In this exercise, you will complete and enhance a simple **Clicker Game** built with **Vue.js**. This game allows users to earn points by clicking a button and purchase upgrades to increase their point gain per click.

Objectives

By working on this project, you will:

- ✓ Understand Vue.js reactivity and data binding.
- ✓ Use Vue directives (`@click` , `:disabled`) to handle interactions.
- ✓ Implement a simple state persistence mechanism using `localStorage` .
- ✓ Modify and extend a Vue.js application to add new features.

Task Instructions

You are provided with a base code that implements a simple **Clicker Game**. Your tasks are:

1. **Fix the missing parts of the code** to ensure upgrades work correctly.
2. **Enhance the game** by adding new features, such as:
 - A **reset button** to restart the game.
 - Additional upgrade levels that increase the click value even further.
 - A visual effect or animation when clicking the button.
 - A progress bar showing the next upgrade milestone.
3. **Ensure data persistence** so progress is saved even after refreshing the page.

Bonus Challenge 🎯

If you're feeling confident, try to:

- Add **auto-clickers** that generate points over time.
- Introduce **different themes or styles** for a better user experience.

Good luck, and happy coding! 🚀