

2D Physics Engine

Language: C++

Focus Area: Simulation, Physics

[View On Github](#)

Overview

The 2D Physics Engine is a lightweight simulation project developed in C++ that models the motion of objects under basic physical forces. It serves as an introduction to physics simulation and focuses on implementing realistic linear motion through a custom-built math library.

The project was created to explore the mathematical foundations of physics simulation and strengthen understanding of vector math, dynamics, and numerical methods in C++.

Features

- **Gravity Simulation:** Objects are affected by a gravitational force, producing natural falling motion.
 - **Drag:** Models air resistance to create more stable and realistic movement.
 - **Custom Vector2 Library:** Implements a full 2D vector math module (addition, subtraction, normalization, dot product, etc.) used throughout the simulation.
 - **Fixed-Time Step:** Ensures consistent simulation results by updating physics at 60 frames per second.
 - **Simple, Extensible Design:** Structured to easily integrate future features such as rotational motion.
-
- Written entirely in modern C++, emphasizing clarity, precision, and efficient updates per frame.
 - Demonstrates a clear understanding of forces, and velocity integration.
 - Implements manual time control rather than relying on rendering loops, ensuring stable and reproducible simulation results.
 - Serves as a foundational framework for future extensions into 2D or 3D real-time simulation.

Learning Outcomes

Through this project, I gained hands-on experience with:

- Implementing physical motion through numerical updates.

- Designing a reusable math library for vector operations.
- Structuring simulation loops for consistent behavior across frames.

Screenshots

[2D Physics Engine Showcase](#)