🌍 Enhanced Globe Visualization

An interactive 3D globe visualization built with Three.js featuring country borders, markers, connections, and real-time data visualization.

✨ Features

Interactive Features

- 🖱️ **Click & Drag**: Rotate the globe by clicking and dragging
- 🔍 **Country Search**: Search for any country and highlight it
- 📍 **Custom Markers**: Add markers at any location on the globe
- 🔗 **Connections**: Visual arcs connecting different locations
- ℹ️ **Tooltips**: Hover over countries and markers for information
- 📊 **Country Info Panel**: Click countries to see detailed information

Visual Enhancements

- 🌟 **Glow Effect**: Inner glow effect around the globe
- 🌌 **Atmosphere**: Outer atmospheric glow effect
- 🎨 **Color-Coded Borders**: Each country has a unique color
- 👁️ **Depth-Aware Opacity**: Borders fade based on their position
- ✨ **Animated Markers**: Pulsing marker effects

Controls

- **Auto-Rotate Toggle**: Enable/disable automatic rotation
- **Rotation Speed**: Adjust rotation speed (0-5x)
- **Zoom Level**: Zoom in/out (0.5x-3x)
- **Reset View**: Reset camera and rotation to default
- **Feature Toggles**: Show/hide markers, connections, effects

Data & Stats

- **Real-time FPS**: Performance monitoring
- **Country Count**: Total countries loaded

- **Marker Count**: Active markers on globe
- **Interactive Legend**: Visual guide to elements

🚀 Quick Start

Prerequisites

- Python 3.x installed on your system
- Modern web browser (Chrome, Firefox, Safari, Edge)

Installation

1. **Download all files** to the same directory:
   - `index.html`
   - `globe.js`
   - `server.py`
2. **Make server.py executable** (Linux/Mac):

```bash
chmod +x server.py
```

3. **Start the server**:

```bash
python3 server.py
```

Or on Windows:

```bash
python server.py
```

4. **Open your browser** and navigate to:

```
http://localhost:8000
```

🎮 How to Use

Basic Interaction

1. **Rotate Globe**: Click and drag on the globe to rotate it
2. **Search Country**: Type in the search box to find countries
3. **View Information**: Click on country borders to see details
4. **Hover Tooltips**: Move mouse over countries or markers

Adding Markers

1. Click "Add Random Marker" to add a marker at a random location
2. Markers will pulse and glow
3. Toggle "Markers" button to show/hide all markers

Visual Effects

- **Glow Effect**: Click "Glow Effect" for inner glow
- **Atmosphere**: Click "Atmosphere" for outer atmospheric effect
- Both effects use shader materials for smooth rendering

Connections

- Click "Connections" to show/hide connection arcs
- Connections are drawn between consecutive markers
- Arcs follow a curved path over the globe surface

Controls

- **Rotation Speed**: Adjust from 0 (stopped) to 5x speed
- **Zoom**: Zoom from 0.5x (far) to 3x (close)
- **Auto-Rotate**: Toggle automatic rotation on/off
- **Reset View**: Return to default camera position

📁 File Structure

```
enhanced-globe/
├── index.html         # Main HTML with UI and styles
├── globe.js          # Three.js globe logic and features
├── server.py         # Python development server
└── README.md         # This file
```

🛠️ Technical Details

Technologies Used

- **Three.js r160**: 3D graphics library
- **ES6 Modules**: Modern JavaScript modules
- **WebGL**: Hardware-accelerated 3D rendering
- **Shader Materials**: Custom GLSL shaders for effects

Data Source

- Country borders from: D3 Graph Gallery GeoJSON
- Auto-loads on page load with loading screen

Performance

- Optimized border rendering with deduplication
- Depth-aware opacity calculations
- FPS counter for performance monitoring
- Efficient raycasting for interaction

🎨 Customization

Adding Custom Markers

Edit `globe.js` and modify the `cities` array in `addSampleData()`:

```javascript
const cities = [
    { lat: 40.7128, lon: -74.0060, label: 'New York' },
    { lat: 51.5074, lon: -0.1278, label: 'London' },
    // Add your cities here
];
```

Changing Colors

Modify the `borderColors` array in `globe.js`:

javascript

```
const borderColors = [
  0xff6b6b, // Red
  0x4ecdc4, // Teal
  // Add more colors in hex format
];
```

Adjusting Globe Size

Change the `radius` variable in `globe.js`:

javascript

```
const radius = 2; // Increase for larger globe
```

🐛 Troubleshooting

Globe not loading

- Check browser console for errors (F12)
- Ensure internet connection (loads external data)
- Try refreshing the page (Ctrl+R or Cmd+R)

Performance issues

- Reduce rotation speed
- Disable glow and atmosphere effects
- Close other browser tabs
- Check FPS counter in stats panel

Server won't start

- Ensure Python 3 is installed: `python3 --version`
- Check if port 8000 is already in use
- Try a different port by editing `server.py`

📝 Browser Compatibility

- ✅ Chrome/Edge 90+
- ✅ Firefox 88+
- ✅ Safari 14+
- ✅ Opera 76+

Requires WebGL support and ES6 modules.

🔧 Development

Adding New Features

1. Edit `globe.js` for 3D logic
2. Edit `index.html` for UI elements
3. Test in browser with server running
4. Server auto-disables caching for development

Debug Mode

Open browser console (F12) to see:

- Loading status
- Error messages
- Performance data

📊 Sample Data

The globe includes:

- **~200 countries** with borders
- **Mock data** for population and GDP
- **Sample markers** for major cities
- **Sample connections** between cities

🌟 Future Enhancements

Possible additions:

- Real-time data integration (weather, flights)
- Heat maps and data visualization
- Time-series animations
- Multiple marker types
- Save/load configurations
- Export screenshots
- Mobile touch controls

📄 License

This project is open source and available for educational and personal use.

🤝 Contributing

Feel free to modify and enhance the code for your needs!

Enjoy exploring the world! 🌍✨