Invoice Generator App

A modular and testable invoice generator built with React (or Next.js) and styled using Tailwind CSS. This app features live invoice preview, tax and total calculations, and PDF export capabilities. The UI components are documented with Storybook, and unit tests are written using Jest and React Testing Library.

Features

- Dynamic invoice form with client details and invoice metadata
- Line items with quantity, unit price, and tax rate selection
- Real-time subtotal, tax, and grand total calculations
- Currency selector supporting multiple currencies with formatting
- Live invoice preview with print-friendly Tailwind CSS styling
- Export invoice preview as PDF
- Fully documented UI components in Storybook
- Comprehensive unit tests for validation and calculations

Core Components

InvoiceForm

- Client Name and Email input fields
- Auto-generated unique Invoice Number
- Invoice Date and Due Date with validation
- Currency selector (e.g., USD, EUR)
- Optional notes or payment terms

LineItem

- Reusable component for invoice items
- Fields for Description, Quantity, Unit Price
- Tax rate dropdown (0%, 5%, 10%, etc.)
- Automatic line total calculation including tax
- Currency formatting based on selected currency

InvoiceTable

- Displays a list of LineItem components
- Supports adding and removing line items dynamically
- Shows Subtotal, aggregated tax, and Grand Total
- All amounts formatted according to selected currency

InvoicePreview

- Live, styled representation of the current invoice data
- Reflects all input in real time including metadata and notes
- Print-friendly layout using Tailwind CSS
- Export the invoice as a PDF document

Testing

Unit tests are implemented using Jest and React Testing Library, covering:

- Form validation for required fields and date logic
- Currency selector behavior and formatting
- LineItem calculations for totals and taxes
- Dynamic addition/removal of line items and totals update
- InvoicePreview accuracy and real-time updates
- PDF export function triggered correctly (mock tested)

Storybook

UI components are documented with Storybook featuring controls and documentation for interactive testing:

- **LineItem.stories.tsx**: empty, pre-filled, validation error states with controls
- **InvoiceTable.stories.tsx**: empty, single, multiple items with varied tax rates and currencies
- **InvoiceForm.stories.tsx**: blank, pre-filled, invalid date range, and control knobs
- **InvoicePreview.stories.tsx**: empty, complete invoices, long descriptions, multiple currencies
- **InvoiceGenerator.stories.tsx**: full integration workflow with mock data

---

Getting Started

Installation

```
npm install
```

Run Development Server

```
npm run dev
```

Run Storybook

```
npm run storybook
```

Run Tests

```
npm test
```

Folder Structure Overview

- `/components` - React components such as InvoiceForm, LineItem, InvoiceTable, InvoicePreview
- `/stories` - Storybook stories for each component
- `/tests` - Unit tests for components
- `/utils` - Helper functions for calculations and formatting