Advanced Angular Banking Dashboard Internship Task (5-Day)

Task Overview:

Develop a **"Bank Customer Portal"** featuring advanced UI/UX, comprehensive data handling, and enhanced security. Use Angular, TypeScript, PrimeNG, and in-memory JSON data (via `HttpClientInMemoryWebApiModule`) until real APIs are integrated.

🚀 Daily Task Breakdown:

📅 Day 1: Completed - Project Setup & Core Utilities

✔ Project setup, theming, layout, and core utilities have been implemented and verified.

📅 Day 2: Multi-Account Management, Statements & Exports

- **Accounts Module**:
  - Display multi-currency support with live dummy rates.
  - Inline editing: account nicknames, alerts threshold.
  - Statement preview: last 12 months, paginated.
- **Exports & Printing**:
  - Export statements to PDF, CSV.
  - Print-friendly statement view.
- **Enhanced Data Grid**:
  - Virtual scroll, column reordering, column resizing.
  - Context menu for quick actions (View statement, Export, Set alert).

📅 Day 3: Transactions, Analytics, Budgets & AI Insights

- **Transactions Module**:
  - Date-range picker, category filter, tagging (user-defined tags).
  - Bulk operations: mark as reviewed, dispute, export selection.
- **Analytics Dashboard**:
  - Charts: spending by category, balance trend, cash-flow waterfall.
  - Budget Planner: set monthly budgets per category, track usage (bar chart).
  - AI Insights (dummy): monthly savings suggestions, expense anomalies.
- **Reporting**:
  - Schedule generation of monthly PDF reports.

o   Email preview (template only).

📅 Day 4: Funds Transfer, Bill Payments, QR & Beneficiaries

- **Transfers & Bill Pay**:
    o   Domestic & international transfer simulation with exchange rates.
    o   QR code payment: generate QR for recipient to scan.
- **OTP & Security**:
    o   Multi-step OTP flow (generate code, validate via dialog).
    o   Rate-limit simulation (max 3 attempts).
- **Beneficiaries**:
    o   CRUD operations, grouping (family, business), favorite flag.
    o   Verify beneficiary details via modal (dummy data match).
- **Recurring & Marketplace**:
    o   Schedule recurring payments (frequency: weekly, monthly, yearly).
    o   In-app marketplace: pay utility bills (electricity, water) with sample list.

📅 Day 5: Loans, Credit Cards, Wealth Module & Security Settings

- **Loan Application & Tracking**:
    o   Multi-step wizard with file upload (ID docs), real-time form validation.
    o   Track status timeline (Pending → Reviewed → Approved/Rejected).
- **Credit Card Management**:
    o   Virtual card issuance: toggle on/off for online, ATM.
    o   Statement PDF download, transaction dispute.
    o   Request limit increase with simulated approval.
- **Wealth Module (Bonus)**:
    o   Simple portfolio overview: mock stocks/funds, performance chart.
    o   Buy/sell simulation (no real trades).
- **Profile & Security**:
    o   Profile edit, change password with strength meter.
    o   Two-Factor Authentication setup simulation (SMS/email toggles).
    o   View audit log of sensitive actions (login, password change, transfers).

⚙️ Technical & Quality Requirements:

- **Frameworks**: Angular v12+, TypeScript, PrimeNG & PrimeIcons.
- **Data Layer**: `HttpClientInMemoryWebApiModule` with modular services.
- **Performance**: Lazy-loaded feature modules; OnPush change detection.

- **PWA & Offline**: Caching strategies for JSON data.
- **Theming & Localization**: Dark/light themes; RTL support for Arabic.
- **Accessibility**: WCAG AA compliance; semantic HTML.
- **Testing & CI**: Unit tests for components/services; GitHub Actions for build & lint.
- **Version Control**: GitFlow with feature branches, PR reviews.

☑ Final Deliverables:

1. **JSON Data Files**: `accounts.json`, `transactions.json`, `analytics.json`, `beneficiaries.json`, `loans.json`, `profile.json`, `transfer-template.json`, `portfolio.json`, `marketplace.json`.

This comprehensive 5-day task simulates end-to-end banking portal development, covering advanced UI flows, data analytics, security processes, and real-world integrations.