⭐ Vue.js – Complete Course Syllabus

📘 Module 1: Introduction to Vue.js

- What is Vue.js?
- History & evolution (Vue 2 → Vue 3)
- Features of Vue.js
- Vue.js vs React vs Angular
- Understanding SPA (Single Page Applications)
- Installing Node.js, npm, yarn
- Setting up a Vue project using:
  - Vue CLI
  - Vite (recommended modern setup)
- Project structure overview
- Running the first Vue application

📘 Module 2: Vue.js Fundamentals

- Understanding Vue instance
- Template syntax
- Data binding (text, HTML, attributes)
- Event handling
- Two-way data binding with **v-model**
- Computed properties
- Watchers (watch & watchEffect)
- Methods vs Computed vs Watch

📘 Module 3: Directives in Vue

- Built-in directives:
  - v-if, v-else, v-show
  - v-for
  - v-bind
  - v-on
  - v-model
  - v-html
- Custom directives
- Directive modifiers

📘 Module 4: Vue Components

- Creating components
- Local & global components
- Props (passing data to children)
- Emitting events to parent (emit)
- Slots & Named Slots
- Dynamic components
- Component communication patterns

📘 Module 5: Vue 3 Composition API (Modern Standard)

- Options API vs Composition API
- setup() function
- Reactive variables using ref() and reactive()
- Computed properties in Composition API
- Watch & watchEffect
- Lifecycle hooks (onMounted, onUpdated, onUnmounted)
- Using reusable functions & logic with **Composables**
- Dependency injection with Provide/Inject

📘 Module 6: Styling & UI in Vue

- Inline styles / Scoped CSS / Global CSS
- CSS Modules
- Working with UI libraries:
  - Bootstrap
  - Tailwind CSS
  - Vuetify (Material Design UI)

📘 Module 7: Vue Router (Client-Side Routing)

- Installing & configuring Vue Router 4
- Creating routes
- Nested routes
- Dynamic routes & route params
- Navigation using router-link
- Route guards (beforeEach, beforeEnter)
- Programmatic navigation
- Redirects & 404 pages
- Lazy loading routes (code splitting)

📘 Module 8: State Management in Vue

Pinia (Official Vue State Manager – Vuex Replacement)

- What is Pinia?
- Setting up Pinia
- Stores: state, getters, actions
- Using Pinia with Composition API
- Async actions (API calls inside stores)
- Modular store structure

Vuex (Optional – Legacy Support)

- State, Actions, Mutations
- Getters
- Modules

📘 Module 9: Working with APIs

- Fetching data using:
    - Fetch API
    - Axios
- Async/await usage
- Handling loading, error & success states
- Working with RESTful APIs
- Interceptors in Axios
- Sending forms & file uploads
- CORS and API headers

📘 Module 10: Vue.js Performance Optimization

- Vue's reactivity system
- Using **computed** & **watch** effectively
- Lazy loading components
- Virtual DOM concepts
- Memoization
- Code splitting & route-level optimization
- Using <keep-alive>
- Optimizing large lists

📘 Module 11: Vue Animations & Transitions

- Built-in transition component
- Page transitions / List transitions
- Integrating GSAP / Animate.css for advanced animations

📘 Module 12: Authentication in Vue.js

- JWT authentication (Login, Signup)
- Storing tokens (LocalStorage, Cookies)
- Role-based routing
- Refresh token flow
- Protecting routes with navigation guards
- Logout functionality

📘 Module 13: Form Handling & Validation

- Working with forms
- Validation using:
    - VeeValidate
    - Yup validation schema
- Error handling / File uploads

📘 Module 14: Vue.js Advanced Concepts

- Render functions
- Teleport (portals)
- Custom events
- Mixins (legacy) vs Composition API
- Reusability patterns
- Error boundaries

📘 Module 15: Testing in Vue (Optional)

- Unit testing Vue components
- Jest setup / Vue Test Utils
- Mocking API calls
- Snapshot testing
- End-to-end testing with Cypress

📘 Module 16: Server-Side Rendering (Nuxt.js)

- Introduction to Nuxt.js
- File-based routing
- Server-side rendering (SSR)
- Static site generation (SSG)
- State management in Nuxt
- API routes in Nuxt
- Image optimization
- Deploying Nuxt apps

📘 Module 17: Deployment & DevOps (Optional)

- Building Vue apps for production
- Deploying on:
    - Netlify, Vercel, Firebase Hosting
    - AWS S3 + CloudFront
- Environment variables & config
- Dockerizing a Vue app
- Setting up CI/CD pipelines (GitHub Actions)

📘 Module 18: Real-World Capstone Project

Build a full **production-ready Vue.js application**, including:

- Authentication, Role-based access
- Full CRUD, API integration
- Pagination & filtering
- State management using Pinia
- Lazy loading, Uploading images/files
- Responsive UI with Tailwind/Vuetify
- Deployment to cloud

Examples:

- E-commerce Frontend
- Social Media Dashboard
- Real-time Chat App
- Admin Panel