

Path for Vue.js Beginners:

Pro Tip: Do code, all assignment questions to understand the logic behind the code. Vs code and console will be your best buddy now.

Use linux for development

Week 1: Basics of JavaScript

- Objective: Establish foundational knowledge of JavaScript syntax and concepts.
- Topics: Variables, Data Types, Operators, Control Structures (if, else, loops), Functions, Arrays, Objects.
- Resources: [JavaScript Tutorial](#) or Online tutorials, interactive coding platforms, JavaScript documentation.

Week 2: Advanced JavaScript

- Objective: Deepen understanding of JavaScript by exploring more complex concepts.
- Topics: Scope, Closures, Prototypes, ES6 Features (Arrow Functions, Classes, Template Literals), Promises, Async/Await.
- Resources: Advanced JavaScript tutorials, books, practical coding exercises.

Week 3: Introduction to Web Frontend

- Objective: Understand the basics of frontend development.
- Topics: HTML, CSS, DOM Manipulation, Responsive Design.
- Resources: Frontend development courses, tutorials, projects.

Week 4: Introduction to VueJS

- Objective: Learn the fundamentals of Vue.js framework.
- Topics: Vue Instance, Vue Directives, Vue Components, Vue Router.
- Resources: Vue.js documentation, beginner tutorials, Vue.js official guide,
 - ▶ GETTING STARTED with VueJS | VueJS | Learning the Basics ,
 - ▶ Vue.js Tutorial From Scratch - e01 - Introduction, Installation & Outputting Data (Lectures 1-6)
 - ▶ Vue JS 2 Tutorial #1 - Introduction (Lectures 1-16)

Quiz1 time: solve all previous year questions and GA PA. The theory part will come from week 1 & 2 mostly and long coding questions form week 3 & 4. Hope instructors don't change the pattern.

Real hustle starts Gear Up mate!!!

Week 5: Vue with APIs

- Objective: Integrate Vue.js frontend with backend APIs.
- Topics: Fetching data from APIs, Axios (HTTP client), RESTful API principles.
- Resources: Vue.js + API integration tutorials, hands-on projects.

ScreenCast 5.1 and [Vue JS 3 Tutorial - 47 - Lifecycle Hooks](#) ,

HTTP Request [What are HTTP requests?](#)

HTTP Response [HTTP Response](#) , [Learn how to build app with API in Vue js tutorial](#)

JavaScript Async/Await [JavaScript Async](#) , Fetch API in Vue

[Fetch api in Javascript | JavaScript Tutorial In Hindi #42](#) **Important**

Week 6: Advanced Vuejs

- Objective: Explore more advanced features of Vue.js.
- Topics: Vuex (State Management), Vue Lifecycle Hooks, Vue Mixins, Form Validation, Single File Components (Theory), Vue Testing.
- Resources: Advanced Vue.js tutorials, Vuex documentation, practical exercises, JavaScript Web Storage [Web Storage API](#) or YouTube tutorial
[JavaScript Cookies vs Local Storage vs Session Storage](#)

Week 7: Advanced State Management

- Objective: Master state management in Vue.js applications.
- Topics: Vuex Modules, Computed Properties, Watchers, State management lecture, Vuex overview, Single Page Application vs Multi Page Application
- Resources: Advanced Vuex tutorials, state management best practices, real world examples, [Vue Routing](#) , [Vuex](#)

Now get ready to start working on project tokens are the starting point

Week 8: Authentication and Designing APIs

Objective: Understand authentication mechanisms and API design principles.

Topics: JWT Authentication, OAuth, API Design Best Practices, GraphQL overview

Resources: Authentication tutorials, API design guidelines, security practices

[Flask API - Token Based Authentication](#) , [What Is JWT and Why Should You Use JWT](#)

Quiz2 time: solve all previous year questions and GA PA. Focus more on router vuex for coding questions cuz that's the new concept theory might come. The API http requests part pyqs will help.

Jan'24 revision session:

<https://github.com/Aditya-IITM-BS/MAD-II-Quiz-II-Revision/blob/main/Questions.js>

Week 9: Asynchronous Jobs

Objective: Learn to handle asynchronous backend jobs in Vue.js applications.

Topics: Background Jobs, Queues, Task Schedulers Web server handling requests, Celery, Redis database.

Resources: Asynchronous programming tutorials, job scheduling libraries [Implement caching](screencast 11.1), [Webhook implementation](screencast 10.3)

Tip: Watch the live session of past terms easy to implement redis and celery use wsl2 or linux environment for redis.

Week 10: InterService Messaging and Webhooks

Objective: Explore communication between services and handling webhooks.

Topics: Message Brokers (e.g., RabbitMQ, Kafka), Webhooks Implementation.

Resources: Messaging systems tutorials, webhook setup guides, hands on projects.

Week 11: Performance

Objective: Optimize Vue.js applications for better performance.

Topics: Code Splitting, Lazy Loading, Performance Monitoring.

Resources: Performance optimization guides, Vue.js performance tools, case studies. **Watch lectures.** [Implement caching in Flask application](screencast 11.1)

Week 12: Project

Objective: Apply all learned concepts to build a realworld Vue.js project.

Topics: Project planning, implementation, testing.

End sem: Week 9-12 are now important so code all the questions and theory might come from week 12. Check out PYQs and GA,PA. CODE'em ALLLL!!!

By following this structured path, beginners will gradually develop the skills and knowledge necessary to build highly interactive frontends using Vue.js, integrate them with backend APIs, handle asynchronous jobs, design APIs, and optimize performance, ultimately culminating in the completion of a comprehensive project.

Extra with extra:

 [useful links](#)