

# Tech Stack Learning Roadmap

## Overview

This roadmap outlines a potential learning path for a full-stack web developer using the specified tech stack. Remember, practical application is key to mastery.

## Backend

(Node.js, Express.js, Sequelize, MySQL)

### 1. Node.js

---

- Core concepts: asynchronous programming, event loop, modules, NPM
  - Practice building small command-line tools and scripts
- 

[Learn Node from the Official Site](#)  
[Official Node Documentation](#)  
[Nest JS](#)

### 2. Express.js:

---

- Routing, middleware, request/response cycle
  - Create RESTful APIs with Express
  - Explore popular middleware (e.g., body-parser, cors)
- 

[tutorials point](#)  
[Learn Express JS In 35 Minutes - Web Dev Simplified](#)  
[Express/Node introduction](#)

### 3. Sequelize:

---

- ORM basics, models, associations, migrations
  - Database interactions (CRUD operations)
- 

[How To Use Sequelize with Node.js and MySQL](#)

## 4. *MySQL*

---

- Database fundamentals, SQL queries (CRUD, joins, aggregations)
  - Database design, normalization
- 

[PostgreSQL vs MySQL - IBM Technology](#)  
[Relational vs. Non-Relational Databases](#)

# Frontend

(React.js, Axios, React Router)

## 1. *React.js*

---

- Components, JSX, state, props, lifecycle methods
  - Build small React applications
  - Explore functional components and hooks
- 

## 2. *Axios*

---

- Making HTTP requests (GET, POST, PUT, DELETE)
  - Handling responses and errors
- 

[RESTful APIs in 100 Seconds // Build an API from Scratch with Node.js Express](#)

[Axios JS Crash Course | GET, POST, PUT, DELETE Requests](#)

[Official Axios Documentation](#)

[Mastering JS IO](#)

### 3. *React Router*

---

- Routing basics, nested routes, parameters
  - Create multi-page applications
- 

[A Complete Guide to React Router: Everything You Need to Know](#)  
[W3Schools - React Router](#)  
[Official React Router Documentation](#)

## Deployment (AWS EC2, AWS RDS)

### AWS EC2:

- Understand EC2 instances, key pairs, security groups
- Deploy Node.js applications on EC2

### AWS RDS:

- Create and manage MySQL databases on RDS
- Connect your application to the RDS instance
- Database backups and security

## Version Control (Git, GitHub)

- Git:
  - Basic commands (init, add, commit, push, pull)
  - Branching, merging, resolving conflicts
  - Git workflow (e.g., Gitflow, GitHub Flow)
- GitHub:
  - Create repositories, collaborate with others
  - Pull requests, code reviews

## Learning Resources

- Official documentation for each technology
- Online tutorials and courses (free and paid)
- Practice projects
- Open-source contributions
- Build personal projects

## Additional Tips

- **Start small:** Build simple applications to solidify your understanding.
- **Practice regularly:** Consistent coding is crucial for improvement.
- **Collaborate:** Work on projects with others to learn different approaches.
- **Leverage community:** Participate in forums and online communities.
- **Stay updated:** Technology evolves, so keep learning new things.

Remember: This is a general roadmap. Adjust it based on your learning style and goals.