**🔹 Basic REST API Concepts**

1. **What is a REST API and how does it differ from a SOAP web service?**
2. **What are the common HTTP methods used in REST APIs?**
3. **Can you explain the difference between PUT and PATCH?**
4. **What is the significance of status codes in REST APIs? Give examples.**
5. **What are idempotent methods in REST? Which HTTP methods are idempotent?**

**🔹 PHP and REST Implementation**

1. **How do you create a REST API using plain PHP (without frameworks)?**
2. **What is the role of .htaccess in building RESTful services in PHP?**
3. **How do you handle JSON request and response in PHP?**
4. **How would you secure your REST API built in PHP?**
5. **Can you demonstrate a simple CRUD example using PHP and MySQL as a REST API backend?**

**🔹 Authentication & Security**

1. **What authentication mechanisms have you used in REST APIs? (e.g., JWT, OAuth2)**
2. **How would you prevent SQL injection in an API endpoint?**
3. **What is CORS, and how do you handle it in PHP APIs?**
4. **How do you validate API input data in PHP?**

**🔹 Error Handling & Best Practices**

1. **How do you handle errors and exceptions in a REST API built with PHP?**
2. **What is the standard way of sending error messages in a REST response?**
3. **How do you version a REST API?**
4. **Explain REST API rate limiting and how you might implement it in PHP.**

**🔹 Tools & Debugging**

1. **How do you test your REST APIs? What tools do you use (e.g., Postman, CURL)?**
2. **Have you used any libraries or frameworks like Guzzle, Lumen, or Slim for API development?**
3. **What’s the difference between Guzzle and CURL in PHP?**

**🔹 Advanced & Scenario-Based**

1. **Your API is slow — how would you troubleshoot and improve performance?**
2. **How would you log API activity and errors in a production environment?**
3. **How do you design a REST API for pagination and filtering large datasets?**
4. **Have you worked with any third-party APIs (e.g., Stripe, Google Maps)? How did you integrate them in PHP?**

**SECURITY**

**🔐 General Security Awareness**

1. **What are the most common security threats in web applications?**
2. **What is the OWASP Top 10? Can you name a few of the top vulnerabilities?**
3. **What is the difference between authentication and authorization?**
4. **How do you store user passwords securely in PHP?**
5. **What is the purpose of password\_hash() and password\_verify() in PHP?**

**🛡️ Input Validation & Sanitization**

1. **What’s the difference between validation and sanitization?**
2. **How do you prevent SQL Injection in PHP?**
3. **Why is using prepared statements (e.g., with PDO or MySQLi) important?**
4. **How do you sanitize user input in PHP? Provide an example.**
5. **How do you prevent XSS (Cross-Site Scripting) attacks in PHP?**

**🔒 Session Management & Authentication**

1. **How do you securely manage sessions in PHP?**
2. **What are session hijacking and fixation? How do you prevent them?**
3. **How do you implement secure login functionality in PHP?**
4. **What is CSRF (Cross-Site Request Forgery)? How can you prevent it?**
5. **What are best practices for handling remember-me functionality?**

**🔐 File Upload & Access Control**

1. **How do you securely handle file uploads in PHP?**
2. **What checks should be performed on uploaded files?**
3. **How do you prevent directory traversal attacks?**
4. **What is access control? How would you implement it in a PHP application?**

**🌐 Web Security Headers & HTTPS**

1. **What are HTTP security headers? Name a few commonly used ones.**
2. **How do you force HTTPS in a PHP application?**
3. **What is Content Security Policy (CSP)?**
4. **What is the purpose of the HttpOnly and Secure flags on cookies?**

**⚙️ Advanced & Scenario-Based**

1. **You find out a PHP app is vulnerable to XSS — how do you fix it?**
2. **How would you protect an API endpoint from abuse (e.g., rate limiting)?**
3. **What steps would you take to secure a legacy PHP application?**
4. **How would you implement JWT authentication securely in a PHP API?**
5. **How do you keep dependencies secure when using Composer?**

**⚙️ General Performance Concepts**

1. **What are some common performance bottlenecks in PHP applications?**
2. **How do you measure the performance of a PHP application?**
3. **What tools do you use to profile or debug performance issues in PHP?**
4. **What is opcode caching and how does it improve performance?**
5. **What is the difference between latency and throughput?**

**🧠 Code-Level Optimization**

1. **How can using isset() be faster than array\_key\_exists()?**
2. **What are the benefits of using single quotes vs double quotes in PHP?**
3. **How can you reduce the memory usage of a PHP script?**
4. **What are generators in PHP and how do they help with performance?**
5. **How do you avoid N+1 query problems in PHP applications?**

**🗃️ Database Optimization**

1. **How do you optimize database queries in PHP?**
2. **What are indexes, and how do they affect performance?**
3. **What is query caching and does PHP/MySQL benefit from it?**
4. **How can prepared statements improve performance as well as security?**
5. **How would you handle large data exports efficiently in PHP?**

**📂 File & Resource Management**

1. **How do you cache data in PHP? What are the options available (APCu, Memcached, Redis)?**
2. **What are the differences between file-based and memory-based caching?**
3. **What’s the impact of reading large files in PHP and how can you optimize it?**
4. **How do you handle image optimization in PHP?**
5. **How would you implement lazy loading in a PHP-based application?**

**🌐 Server-Side & Network Optimization**

1. **How do you reduce server response time for a PHP web app?**
2. **What are the benefits of using a reverse proxy (e.g., Nginx) with PHP?**
3. **What’s PHP-FPM and how does it improve performance over mod\_php?**
4. **How does using a CDN benefit PHP-based websites?**
5. **How do you use GZIP compression in PHP responses?**

**🧪 Testing & Monitoring**

1. **How do you monitor PHP performance in production?**
2. **What are some performance monitoring tools you’ve used (e.g., New Relic, Blackfire, Xdebug)?**
3. **How do you perform a load test for your PHP app? What tools do you recommend?**
4. **How do you ensure code changes don't degrade performance over time?**
5. **What is a performance budget, and how do you apply it in your work?**

**🎯 Scenario-Based Questions**

1. **You notice a PHP script is taking too long to load — how do you debug it?**
2. **How would you optimize a PHP e-commerce site with high traffic and product listings?**
3. **A page loads slowly only on mobile devices — how would you investigate?**
4. **Your app is slow during peak hours but fine otherwise — what steps do you take?**
5. **How do you choose between optimizing code vs scaling hardware?**