# Frequently Asked Questions


- General Questions
  - What is WPGraphQL?
  - Do I need to know GraphQL to use WPGraphQL?
- Common Issues
  - Why can't I see my Custom Post Type in the GraphQL Schema?
  - Why aren't my Menus showing up in queries?
  - How do I handle authentication?
  - Why am I getting "Authentication failed" errors?
  - How do I debug WPGraphQL?
  - How do I optimize performance?
  - Why can't I mutate (create/update/delete) data?
- Development Questions
  - How do I test my WPGraphQL implementation?
  - How do I extend the WPGraphQL schema?


# General Questions

## What is WPGraphQL?

WPGraphQL is a free, open-source WordPress plugin that provides an extendable GraphQL schema and API for any WordPress site.

## Do I need to know GraphQL to use WPGraphQL?

While GraphQL knowledge is helpful, you can get started with WPGraphQL using the built-in GraphiQL IDE which provides documentation, auto-completion, and query building tools. See our Intro to GraphQL guide to learn the basics.

# Common Issues

## Why can't I see my Custom Post Type in the GraphQL Schema?

Custom Post Types must be explicitly configured to show in GraphQL. Make sure you've set the following when registering your post type:

```
register_post_type('book', [
  'show_in_graphql' => true,
  'graphql_single_name' => 'book',
  'graphql_plural_name' => 'books',
  // ... other settings
]);
```

For detailed instructions and best practices, see our Custom Post Types guide.

## Why aren't my Menus showing up in queries?

> Menus and Menu Items in WordPress are not viewable by public requests until they are assigned to a Menu Location. WPGraphQL respects this access control.

Learn more about working with menus in our Menus guide.

## How do I handle authentication?

WPGraphQL itself doesn't include built-in authentication features. You'll need to use one of these common solutions:

1. WPGraphQL JWT Authentication
2. WPGraphQL Headless Login
3. WordPress Application Passwords
4. Cookie-based Authentication (for admin-context applications)

For detailed setup instructions and security considerations, see our Authentication and Authorization guide.

# Why am I getting "Authentication failed" errors?

Common causes include:

- No authentication method configured
- Missing or malformed authentication headers
- Invalid credentials
- Missing proper capabilities for the requested operation

> Never send authentication credentials in the query parameters. Always use HTTP headers for authentication.

See our Common Issues guide for troubleshooting steps.

# How do I debug WPGraphQL?

You can enable debugging through several methods:

1. Using the WPGraphQL Debug settings in wp-admin
2. Adding `define( 'GRAPHQL_DEBUG', true );` to wp-config.php
3. Using Query Monitor plugin with GraphQL Query Logging enabled

> Use the Query Monitor plugin during development. It provides detailed insights into query execution, performance metrics, and potential issues.

For comprehensive debugging strategies, see our Debugging guide.

# How do I optimize performance?

Key optimization strategies:

- Request only the fields you need
- Use fragments for reusable field selections
- Consider implementing caching
- Monitor query complexity and depth

> Large queries can impact performance. Consider implementing pagination for large datasets using the `first` and `after` arguments.

Learn more about optimizing queries in our Performance guide.

# Why can't I mutate (create/update/delete) data?

If you're unable to create, update, or delete data, check:

- User authentication status
- User capabilities and roles
- Required input fields
- Field validation rules

See our GraphQL Mutations guide for detailed examples and troubleshooting.

# Development Questions

## How do I test my WPGraphQL implementation?

You can test using:

1. The built-in GraphiQL IDE
2. Codeception tests (for plugin development)
3. Docker (via wp-env) testing environment
4. Local development environment with debugging enabled

For detailed testing instructions, see our Testing guide.

# How do I extend the WPGraphQL schema?

You can extend the schema by:

1. Registering custom types
2. Adding fields to existing types
3. Creating custom mutations
4. Using the various WPGraphQL filter hooks

Learn more in our Extending WPGraphQL guide.

For more detailed information on any of these topics, please refer to the specific documentation sections.

---

□□□ #2
□ □□□ □□□ 2026-08-29 23:36:26 UTC
□ □□□ □□□ 2026-08-29 23:36:26 UTC