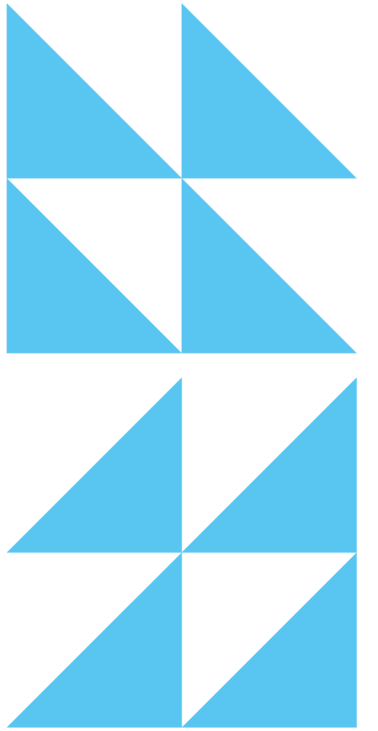


SESSION 3: PLUGIN DEVELOPMENT ESSENTIALS

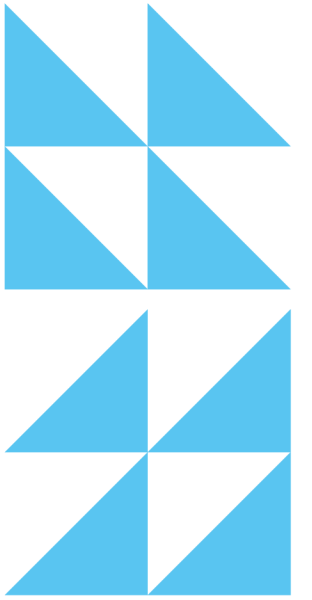
Learn how to create, customize, and
optimize WordPress plugins.



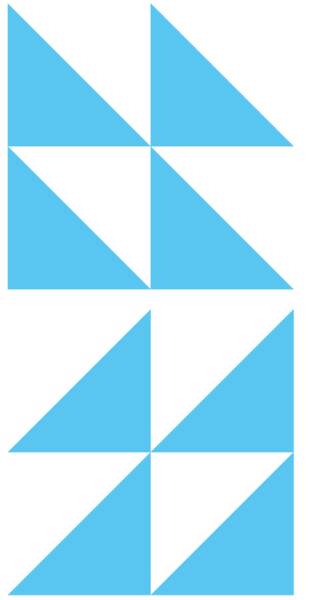
LEARNING OUTCOMES

By the end of this session, you will be able to:

- ✓ Build and activate a custom WordPress plugin.
- ✓ Use hooks and filters to extend WordPress functionality.
- ✓ Follow best practices for security and performance.



INTRODUCTION TO WORDPRESS PLUGINS



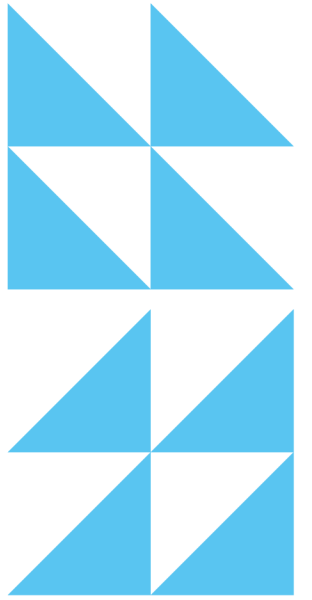
What are WordPress Plugins?

- Plugins add features and functionality to WordPress without modifying core files.
- Examples:
 - WooCommerce (e-commerce), Yoast SEO (SEO optimization), Contact Form 7 (forms).
- Plugin directory location: /wp-content/plugins/



CREATING A SIMPLE PLUGIN

– STEP 1



Setting Up the Plugin:

- 1** Navigate to the /wp-content/plugins/ directory.
- 2** Create a new folder called my-first-plugin.
- 3** Inside the folder, create a PHP file called my-first-plugin.php.

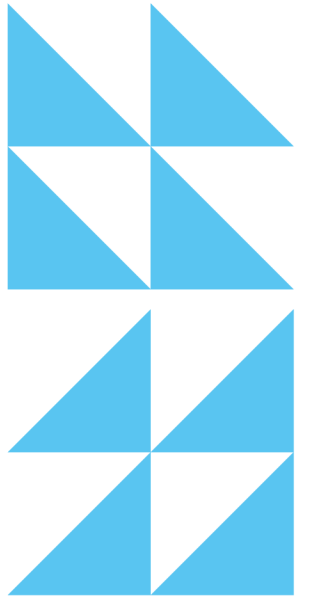


CREATING A SIMPLE PLUGIN

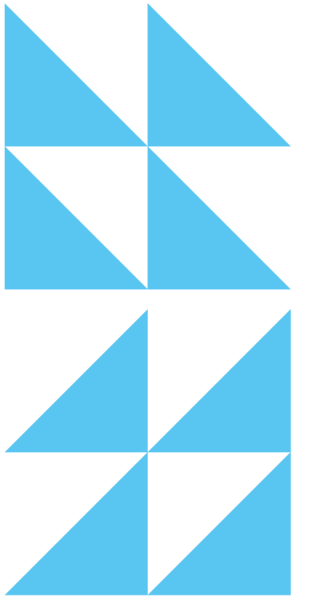
– STEP 1

Example Code (Plugin Header):

```
<?php
/**
 * Plugin Name: My First Plugin
 * Plugin URI: https://example.com
 * Description: A simple plugin that adds a footer message.
 * Version: 1.0
 * Author: Your Name
 * License: GPL2
 */
```



CREATING A SIMPLE PLUGIN – STEP 2



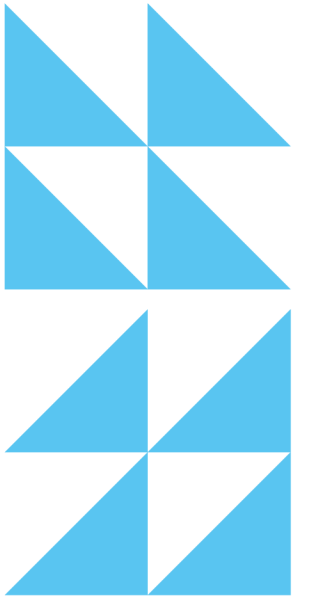
Activating the Plugin:

- 1** Go to the WordPress Admin Dashboard Plugins.
- 2** Locate "My First Plugin" in the list.
- 3** Click Activate.

Activity: Create and activate your own plugin.



ADDING FUNCTIONALITY TO THE PLUGIN



Displaying a Custom Message on Every Page

- Modify the my-first-plugin.php file and add:

```
function custom_footer_message() {  
    echo '<p style="text-align:center; color:blue;">This is a custom footer message.</p>';  
}  
add_action('wp_footer', 'custom_footer_message');
```

- This function will display a custom message in the footer of every page.



UNDERSTANDING HOOKS AND FILTERS

Hooks: Let us add new functionality at predefined points.

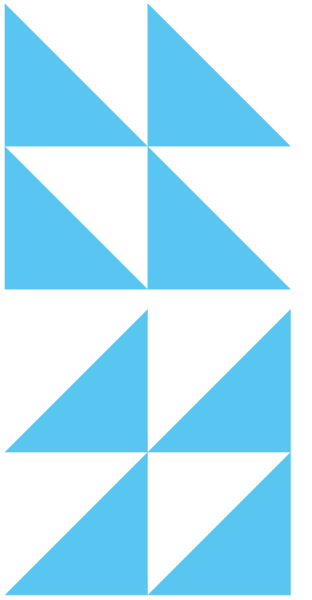
Action Hooks

Execute custom code.

Example:

Action Hook (Adding content before a post)

```
function add_text_before_post($content) {  
    return '<p><strong>Welcome to my blog!</strong></p>' . $content;  
}  
add_filter('the_content', 'add_text_before_post');
```



UNDERSTANDING HOOKS AND FILTERS

Hooks: Let us add new functionality at predefined points.

Filter Hooks

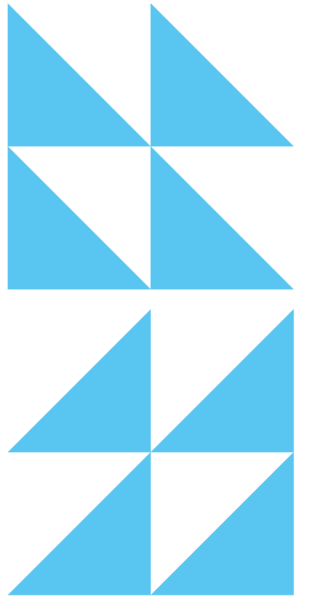
Modify existing content.

Example:

Changing the Excerpt Length Using a Filter Hook

```
function custom_excerpt_length($length) {  
    return 20; // Limit excerpt to 20 words  
}  
  
add_filter('excerpt_length', 'custom_excerpt_length');
```

Activity: Modify the example to display 10 words instead.

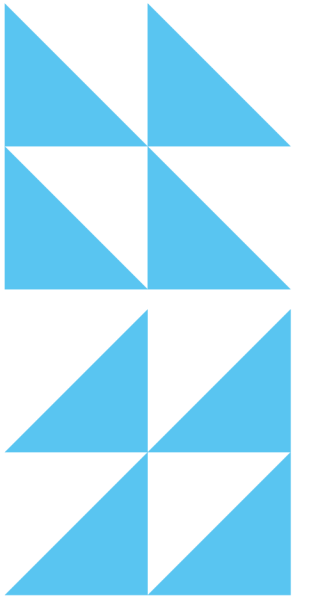


PLUGIN SECURITY BEST PRACTICES

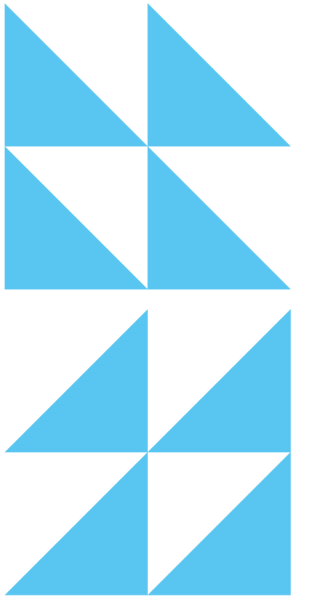
- ✓ Validate & Sanitize User Input (Prevent malicious input)
- ✓ Use Nonces (Prevent CSRF attacks)
- ✓ Follow WordPress Coding Standards
- ✓ Escape Output Properly (Avoid XSS attacks)

Example: Securely saving a plugin setting

```
if(isset($_POST['custom_input'])){  
    update_option('my_plugin_setting', sanitize_text_field($_POST['custom_input']));  
}
```



OPTIMIZING PLUGIN PERFORMANCE



- ◆ Load Scripts & Styles Correctly
(`wp_enqueue_script()`, `wp_enqueue_style()`).
- ◆ Limit Database Queries – Avoid excessive queries inside loops.
- ◆ Use Caching Mechanisms for performance improvement.
- ◆ Keep Code Lightweight – Avoid unnecessary functions.

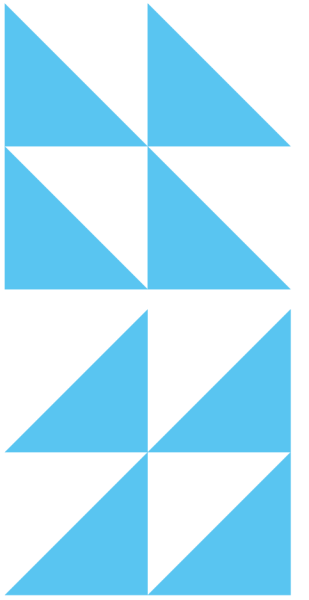
Examples

```
function enqueue_custom_styles() {  
    wp_enqueue_style('custom-plugin-style', plugin_dir_url(__FILE__) . 'style.css');  
}  
add_action('wp_enqueue_scripts', 'enqueue_custom_styles');
```

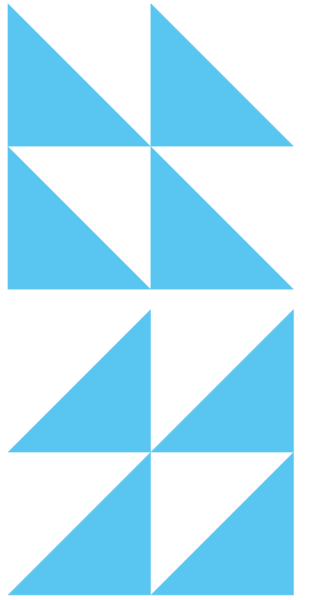


HANDS-ON ACTIVITY

- 1** Modify your plugin to include a filter hook that changes the post title.
- 2** Add a secure settings page where users can enter a custom footer message.
- 3** Test and debug your plugin.



HANDS-ON ACTIVITY



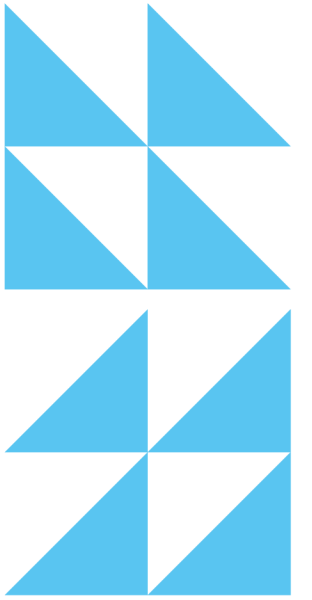
 Modify your plugin to:

- ✓ Add another action hook (e.g., show a message at the top of each page).
- ✓ Experiment with a new filter hook.
- ✓ Research how to add plugin settings to the WordPress dashboard.



SUMMARY & Q&A

- ◆ Plugins allow you to extend WordPress functionality.
- ◆ Hooks & Filters help integrate custom functionality.
- ◆ Security & Optimization ensure the best performance.
- ◆ Best Practice: Keep your plugins lightweight and secure.



THANK
YOU

