



A horizontal row of 20 empty square boxes, likely used for grading or marking student responses.



WordPress

 WordPress 

100% of the time

“manage options”

Wordpress

A horizontal row of 20 empty square boxes, intended for students to write their answers in a grid format.



A horizontal bar composed of 20 small, equal-sized squares arranged side-by-side. The bar is positioned horizontally across the page.



A horizontal bar composed of 24 small, equal-sized squares arranged side-by-side.

```

/*
 * Generate a Delete link based on the homepage url.
 *
 * @param string $content Existing content.
 *
 * @return string|null
 */
function wporg_generate_delete_link( $content ) {
    // Run only for single post page.
    if ( is_single() && in_the_loop() && is_main_query() ) {
        // Add query arguments: action, post.
        $url = add_query_arg(
            [
                'action' => 'wporg_frontend_delete',
                'post'   => get_the_ID(),
                'url', home_url()
            ],
            esc_url( $url )
        );
        return $content . ' <a href="' . esc_url( $url ) . '">' . esc_html__( 'Delete Post', 'wporg' ) . '</a>';
    }
}

return null;
}

```

```

/*
 * Request handler
 */
function wporg_delete_post() {
    if ( isset( $_GET['action'] ) && 'wporg_frontend_delete' === $_GET['action'] ) {
        // Verify we have a post id.
        $post_id = ( isset( $_GET['post'] ) ) ? ( $_GET['post'] ) : ( null );
        // Verify there is a post with such a number.
        $post = get_post( (int) $post_id );
        if ( empty( $post ) ) {
            return;
        }

```

```

    // Delete the post.
    wp_trash_post( $post_id );

    // Redirect to admin page.
    $redirect = admin_url( 'edit.php' );
    wp_safe_redirect( $redirect );

    // We are done.
    die;
}

/*
 * Add the delete link to the end of the post content.
 */
add_filter( 'the_content', 'wporg_generate_delete_link' );

/*
 * Register our request handler with the init hook.
 */
add_action( 'init', 'wporg_delete_post' );

```

██████████

█████████████████████“”█████████████████████“”████████

`edit_others_posts` ██████████████████████

```

/*
 * Generate a Delete link based on the homepage url.
 *
 * @param string $content Existing content.
 *
 * @return string|null
 */
function wporg_generate_delete_link( $content ) {
    // Run only for single post page.
    if ( is_single() && in_the_loop() && is_main_query() ) {
        // Add query arguments: action, post.

```

```

$url = add_query_arg(
    [
        'action' => 'wporg_frontend_delete',
        'post'   => get_the_ID(),
    ],
    home_url()
);

return $content . ' <a href="' . esc_url( $url ) . '">' . esc_html__( 'Delete Post', 'wporg' ) . '</a>';
}

return null;
}

/**
 * Request handler
 */
function wporg_delete_post() {
    if ( isset( $_GET['action'] ) && 'wporg_frontend_delete' === $_GET['action'] ) {

        // Verify we have a post id.
        $post_id = ( isset( $_GET['post'] ) ) ? ( $_GET['post'] ) : ( null );

        // Verify there is a post with such a number.
        $post = get_post( (int) $post_id );
        if ( empty( $post ) ) {
            return;
        }

        // Delete the post.
        wp_trash_post( $post_id );

        // Redirect to admin page.
        $redirect = admin_url( 'edit.php' );
        wp_safe_redirect( $redirect );

        // We are done.
        die;
    }
}

```

```
/**  
 * Add delete post ability  
 */  
add_action('plugins_loaded', 'wporg_add_delete_post_ability');  
  
function wporg_add_delete_post_ability() {  
    if ( current_user_can( 'edit_others_posts' ) ) {  
        /**  
         * Add the delete link to the end of the post content.  
         */  
        add_filter( 'the_content', 'wporg_generate_delete_link' );  
  
        /**  
         * Register our request handler with the init hook.  
         */  
        add_action( 'init', 'wporg_delete_post' );  
    }  
}
```

---

□□ #2  
□ Vcanson □ 20 □ 2023 10:48:40  
□ Vcanson □ 26 □ 2023 15:34:10