

CONSULTAS MARIADB SOBRE WORDPRESS

Contenido

CONSULTAS MARIADB SOBRE WORDPRESS.....	1
Mostrar la estructura de tablas de la base de datos con sus relaciones.	2
Muestra los usuarios de WordPress y sus roles.	3
Consulta los posts publicados.....	3
Cuenta el número de entradas que hay por cada usuario de WordPress	4
Muestra cuántos usuarios hay por cada rol.....	4
Consulta las páginas publicadas por cada usuario (título y fecha)	5
Mostrar y modificar si procede los siguientes parámetros relativos a las conexiones en el archivo de configuración.....	5
Puerto de escucha.....	6
Número máximo de conexiones	6
Tiempo de espera antes de cerrar una conexión inactiva	6
Asignar nuevo rol y permisos	7

Mostrar la estructura de tablas de la base de datos con sus relaciones.

```
Database changed
MariaDB [wordpress]> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_actionscheduler_actions |
| wp_actionscheduler_claims |
| wp_actionscheduler_groups |
| wp_actionscheduler_logs |
| wp_commentmeta |
| wp_comments |
| wp_e_events |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_surecart_incoming_webhooks |
| wp_surecart_integrations |
| wp_surecart_variant_option_values |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----+
20 rows in set (0.001 sec)
```

```
MariaDB [wordpress]> describe wp_posts;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| ID | bigint(20) unsigned | NO | PRI | NULL | auto_increment |
| post_author | bigint(20) unsigned | NO | MUL | 0 | |
| post_date | datetime | NO | | 0000-00-00 00:00:00 | |
| post_date_gmt | datetime | NO | | 0000-00-00 00:00:00 | |
| post_content | longtext | NO | | NULL | |
| post_title | text | NO | | NULL | |
| post_excerpt | text | NO | | NULL | |
| post_status | varchar(20) | NO | | publish | |
| comment_status | varchar(20) | NO | | open | |
| ping_status | varchar(20) | NO | | open | |
| post_password | varchar(255) | NO | | | |
| post_name | varchar(200) | NO | MUL | | |
| to_ping | text | NO | | NULL | |
| pinged | text | NO | | NULL | |
| post_modified | datetime | NO | | 0000-00-00 00:00:00 | |
| post_modified_gmt | datetime | NO | | 0000-00-00 00:00:00 | |
| post_content_filtered | longtext | NO | | NULL | |
| post_parent | bigint(20) unsigned | NO | MUL | 0 | |
| guid | varchar(255) | NO | | | |
| menu_order | int(11) | NO | | 0 | |
| post_type | varchar(20) | NO | MUL | post | |
| post_mime_type | varchar(100) | NO | | | |
| comment_count | bigint(20) | NO | | 0 | |
+-----+
23 rows in set (0.002 sec)

MariaDB [wordpress]>
```

Muestra los usuarios de WordPress y sus roles.

```
MariaDB [wordpress]> SELECT
  -> u.ID,
  -> u.user_login,
  -> u.user_email,
  -> um.meta_value AS rol
  -> FROM wp_users u
  -> JOIN wp_usermeta um ON u.ID = um.user_id
  -> WHERE um.meta_key = 'wp_capabilities';
+-----+-----+-----+-----+
| ID | user_login | user_email | rol |
+-----+-----+-----+-----+
| 1 | root | ygutierrezm05@educantabria.es | a:1:{s:13:"administrator";b:1;} |
+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [wordpress]> |
```

Consulta los posts publicados

```
MariaDB [wordpress]> SELECT
  -> p.ID,
  -> p.post_title,
  -> u.user_login AS autor,
  -> p.post_date
  -> FROM wp_posts p
  -> JOIN wp_users u ON p.post_author = u.ID
  -> WHERE p.post_type = 'post'
  -> AND p.post_status = 'publish'
  -> ORDER BY p.post_date DESC;
+-----+-----+-----+-----+
| ID | post_title | autor | post_date |
+-----+-----+-----+-----+
| 1 | ¡Hola, mundo! | root | 2025-12-19 18:13:35 |
+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [wordpress]>
```

Cuenta el número de entradas que hay por cada usuario de WordPress

```
MariaDB [wordpress]> SELECT
  -> u.user_login AS autor,
  -> COUNT(p.ID) AS total_posts
  -> FROM wp_users u
  -> LEFT JOIN wp_posts p
  -> ON u.ID = p.post_author
  -> AND p.post_type = 'post'
  -> AND p.post_status = 'publish'
  -> GROUP BY u.ID;
```

autor	total_posts
root	1
admin	0

2 rows in set (0.001 sec)

```
MariaDB [wordpress]> |
```

Muestra cuántos usuarios hay por cada rol

```
MariaDB [wordpress]> SELECT
  -> meta_value AS rol,
  -> COUNT(*) AS total_usuarios
  -> FROM wp_usermeta
  -> WHERE meta_key = 'wp_capabilities'
  -> GROUP BY meta_value;
```

rol	total_usuarios
a:1:{s:13:"administrator";b:1;}	2

1 row in set (0.002 sec)

```
MariaDB [wordpress]>
```

Consulta las páginas publicadas por cada usuario (título y fecha)

```
MariaDB [wordpress]> SELECT
-> login, p.post_da u.user_login AS autor,
-> p.post_title,
-> p.post_date
-> FROM wp_posts p
-> JOIN wp_users u ON p.post_author = u.ID
-> WHERE p.post_type = 'page'
-> AND p.post_status = 'publish'
-> ORDER BY u.user_login, p.post_date;
```

autor	post_title	post_date
root	Página de ejemplo	2025-12-19 18:13:35
root	Pago	2025-12-19 19:00:29
root	Panel	2025-12-19 19:00:29
root	Tienda	2025-12-19 19:00:29

4 rows in set (0.001 sec)

```
MariaDB [wordpress]> |
```

Mostrar y modificar si procede los siguientes parámetros relativos a las conexiones en el archivo de configuración

```
// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'root' );

/** Database password */
define( 'DB_PASSWORD', 'usuario2' );

/** Database hostname */
define( 'DB_HOST', 'fatgram.chgqawmakccd.us-east-1.rds.amazonaws.com' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8mb4' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );
```

Puerto de escucha

```
Database changed
MariaDB [wordpress]> SHOW VARIABLES LIKE 'port';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| port          | 3306  |
+-----+-----+
1 row in set (0.001 sec)

MariaDB [wordpress]> |
```

Número máximo de conexiones

```
MariaDB [wordpress]> SHOW VARIABLES LIKE 'max_connections';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| max_connections | 30    |
+-----+-----+
1 row in set (0.001 sec)

MariaDB [wordpress]> |
```

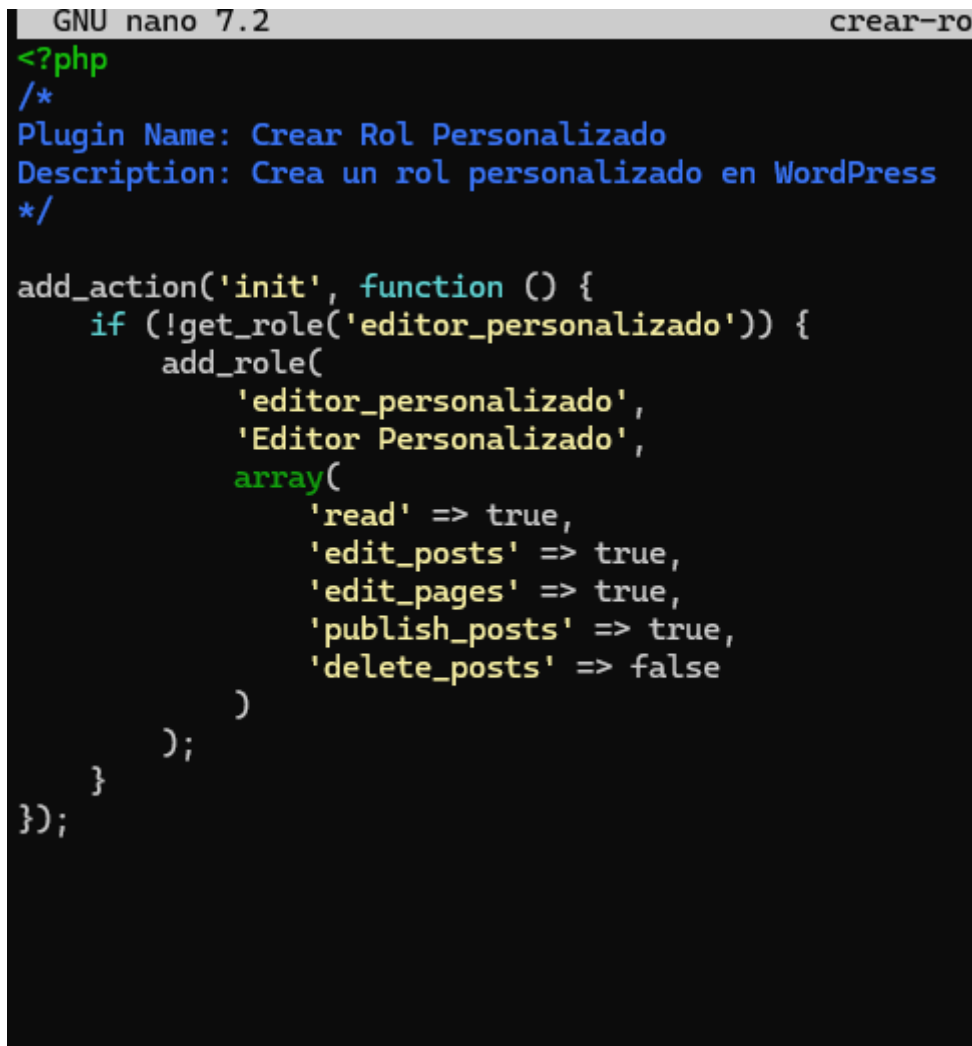
Tiempo de espera antes de cerrar una conexión inactiva

```
MariaDB [wordpress]> SHOW VARIABLES LIKE 'wait_timeout';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| wait_timeout  | 28800 |
+-----+-----+
1 row in set (0.001 sec)

MariaDB [wordpress]> |
```

Asignar nuevo rol y permisos

Para asignar nuevos roles y permisos sobre este, tenemos dos opciones, si tenemos el tema de wordpress asignado, debemos de modificar el fichero wp-function.php. En el caso de no tener asignado todavía el tema tenemos que crear nosotros mismo el plugin desde plugins

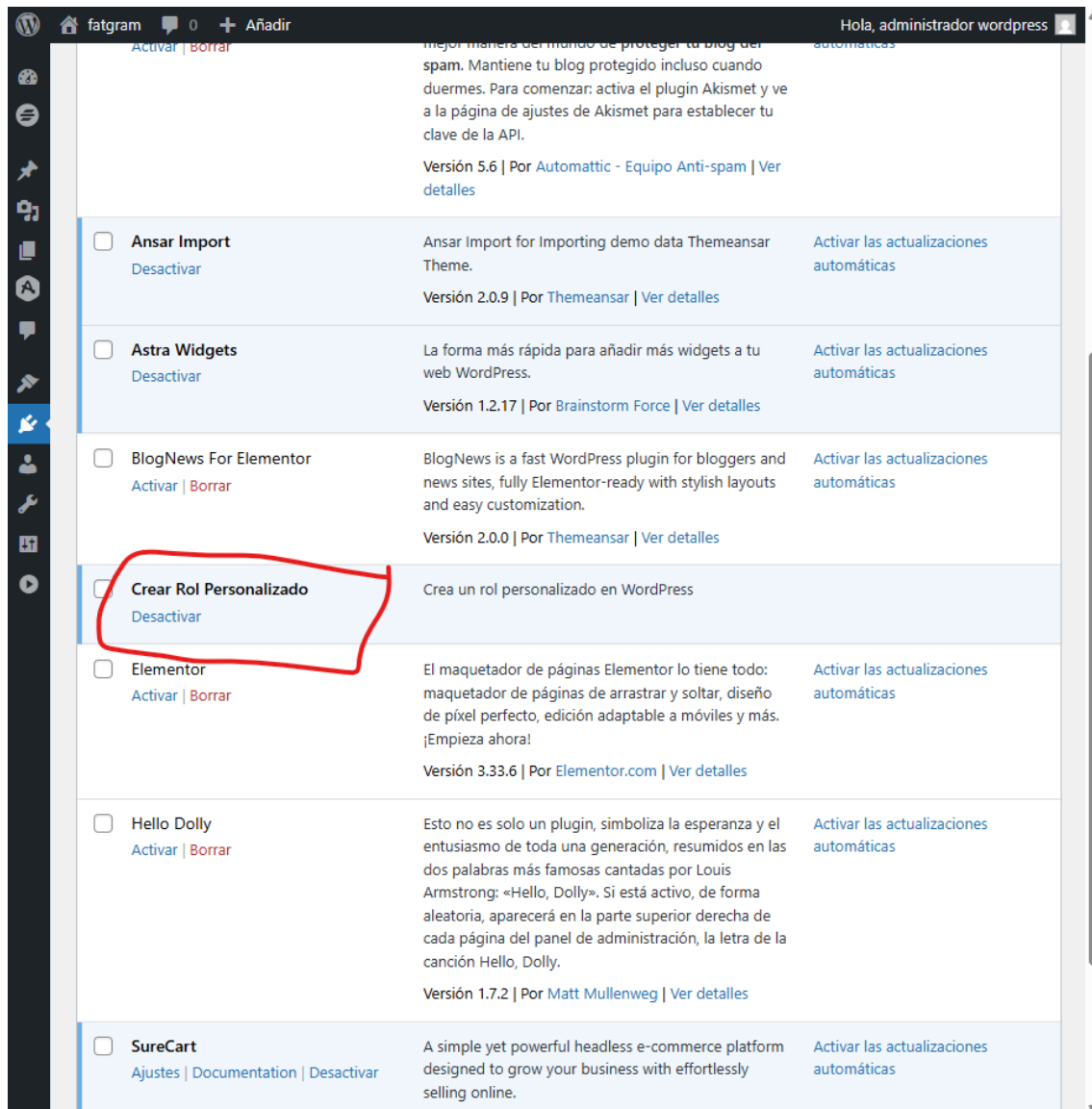
A screenshot of a GNU nano 7.2 text editor window. The title bar shows 'GNU nano 7.2' on the left and 'crear-ro' on the right. The editor contains PHP code for a WordPress plugin. The code starts with a green comment block: `<?php`, `/*`, `Plugin Name: Crear Rol Personalizado`, `Description: Crea un rol personalizado en WordPress`, and `*/`. Below this is a function `add_action('init', function () {`. Inside the function, there is an `if (!get_role('editor_personalizado')) {` statement. This is followed by an `add_role(` function call with three arguments: `'editor_personalizado'`, `'Editor Personalizado'`, and an `array(` containing permissions: `'read' => true,`, `'edit_posts' => true,`, `'edit_pages' => true,`, `'publish_posts' => true,`, and `'delete_posts' => false`. The code ends with closing braces and semicolons: `);`, `}`, and `});`.

```
GNU nano 7.2                                crear-ro
<?php
/*
Plugin Name: Crear Rol Personalizado
Description: Crea un rol personalizado en WordPress
*/

add_action('init', function () {
    if (!get_role('editor_personalizado')) {
        add_role(
            'editor_personalizado',
            'Editor Personalizado',
            array(
                'read' => true,
                'edit_posts' => true,
                'edit_pages' => true,
                'publish_posts' => true,
                'delete_posts' => false
            )
        );
    }
});
```

Después de poner la configuración lo guardamos y accedemos al WordPress desde el navegador

Una vez dentro de plugins activamos el plugin de crear rol personalizado, después wordpress ejecutara el php y el rol quedara guardado



Aquí vemos que el rol esta creado

user_login	rol
root	a:1:{s:13:"administrator";b:1;}
admin	a:1:{s:13:"administrator";b:1;}