

LOS ROLES ARIA

Los Roles ARIA son fundamentales para aplicaciones web interactivas y ayudan a los usuarios con tecnologías de asistencia (como lectores de pantalla).

EJEMPLO 1: ROLE ALERT

Se utiliza para notificar a los usuarios sobre cambios importantes.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo Role ARIA: Alert</title>
</head>
<body>
  <h1>Ejemplo de Role ARIA: alert</h1>
  <button onclick="mostrarAlerta()">Mostrar alerta</button>
  <div id="alerta" role="alert" style="display: none; color: red;">
    ¡Este es un mensaje de alerta importante!
  </div>

  <script>
    function mostrarAlerta() {
      const alerta = document.getElementById('alerta');
      alerta.style.display = 'block';
    }
  </script>
</body>
</html>
```

EXPLICACIÓN:

- **Role alert:** Notifica automáticamente a los lectores de pantalla cuando el contenido del elemento cambia, alertando a los usuarios de información importante.
- **Uso práctico:** Ideal para mostrar mensajes críticos como errores en formularios o alertas de sistema.
- **Comportamiento:** Cuando haces clic en el botón, el contenido de <div> se muestra. Un lector de pantalla lo anunciará automáticamente, incluso si el usuario no está enfocado en ese elemento.

EJEMPLO 2: ROLE BUTTON

Para personalizar elementos como botones accesibles.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo Role ARIA: Button</title>
</head>
```

```

<body>
  <h1>Ejemplo de Role ARIA: button</h1>
  <div role="button" tabindex="0" onclick="alert('Botón activado')"
onkeypress="if(event.key === 'Enter'){ alert('Botón activado'); }">
    Presiona aquí (div con rol de botón)
  </div>
</body>
</html>

```

EXPLICACIÓN:

- **Role button:** Permite que un elemento no interactivo (como <div>) actúe como un botón para usuarios con tecnologías de asistencia.
- **tabindex="0":** Hace que el elemento sea enfocable con el teclado.
- **onkeypress:** Permite activar el "botón" usando la tecla Enter.
- **Uso práctico:** Personalizar botones cuando no se utiliza el elemento <button> nativo.

EJEMPLO 3: ROLE DIALOG

Define un cuadro de diálogo accesible.

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo Role ARIA: Dialog</title>
  <style>
    #dialogo {
      display: none;
      position: fixed;
      top: 50%;
      left: 50%;
      transform: translate(-50%, -50%);
      border: 1px solid #000;
      padding: 20px;
      background: #fff;
      z-index: 1000;
    }
  </style>
</head>
<body>
  <h1>Ejemplo de Role ARIA: dialog</h1>
  <button onclick="abrirDialogo()">Abrir diálogo</button>
  <div id="dialogo" role="dialog" aria-labelledby="tituloDialogo" aria-
hidden="true">
    <h2 id="tituloDialogo">Diálogo accesible</h2>
    <p>Este es un cuadro de diálogo con role ARIA.</p>
    <button onclick="cerrarDialogo()">Cerrar</button>
  </div>

  <script>
    function abrirDialogo() {

```

```

        const dialogo = document.getElementById('dialogo');
        dialogo.style.display = 'block';
        dialogo.setAttribute('aria-hidden', 'false');
    }

    function cerrarDialogo() {
        const dialogo = document.getElementById('dialogo');
        dialogo.style.display = 'none';
        dialogo.setAttribute('aria-hidden', 'true');
    }
</script>
</body>
</html>

```

EXPLICACIÓN:

- **Role dialog:** Define una ventana emergente accesible que funciona como un cuadro de diálogo.
- **aria-labelledby:** Asocia el diálogo con el título (<h2>), lo que ayuda a los lectores de pantalla a identificar el propósito del cuadro.
- **aria-hidden="true":** Indica inicialmente que el diálogo está oculto para los lectores de pantalla.
- **Uso práctico:** Implementar modales o ventanas emergentes que sean fáciles de navegar y entender.

EJEMPLO 4: ROLE NAVIGATION

Define una región de navegación accesible.

```

<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <title>Ejemplo Role ARIA: Navigation</title>
</head>
<body>
    <header>
        <nav role="navigation" aria-label="Menú principal">
            <ul>
                <li><a href="#inicio">Inicio</a></li>
                <li><a href="#servicios">Servicios</a></li>
                <li><a href="#contacto">Contacto</a></li>
            </ul>
        </nav>
    </header>
</body>
</html>

```

EXPLICACIÓN:

- **Role navigation:** Identifica una sección de navegación del sitio web.
- **aria-label:** Proporciona un nombre accesible (en este caso, "Menú principal") para la región de navegación.

- **Uso práctico:** Ayuda a los usuarios a navegar fácilmente por un sitio web usando tecnologías de asistencia.

EJEMPLO 5: ROLE PROGRESSBAR

Indica un progreso en tiempo real.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo Role ARIA: Progressbar</title>
</head>
<body>
  <h1>Ejemplo de Role ARIA: progressbar</h1>
  <div role="progressbar" aria-valuenow="50" aria-valuemin="0" aria-
valuemax="100" style="width: 50%; background-color: blue; height: 20px;">
    <span style="color: white;">50%</span>
  </div>
</body>
</html>
```

EXPLICACIÓN:

- **Role progressbar:** Representa el estado de progreso de una tarea, como una barra de carga.
- **aria-valuenow:** Muestra el progreso actual (en este caso, 50%).
- **aria-valuemin y aria-valuemax:** Especifican los valores mínimo y máximo (0 y 100, respectivamente).
- **Uso práctico:** Hacer que los estados de progreso sean accesibles para los lectores de pantalla, útil en formularios largos o tareas de carga.

RESUMEN DE LOS ROLES ARIA:

- Los **Roles ARIA** ayudan a que las aplicaciones dinámicas sean entendidas por tecnologías de asistencia como lectores de pantalla.
- Cada role (alert, button, dialog, etc.) define claramente el propósito del elemento, mejorando la experiencia para usuarios con discapacidades visuales, motoras o cognitivas.
- Las propiedades adicionales de ARIA (como aria-label, aria-valuenow, aria-hidden, etc.) aportan información complementaria para mejorar aún más la accesibilidad.