

Nombre del Caso de Prueba: Verificación de Funciones de Administradores

Identificador: ADMIN-001

Descripción: Pruebas para verificar las funciones relacionadas con administradores en el sistema.

Precondiciones:

1. El archivo `Pruebas.ts` contiene las funciones `eliminarAdministrador`, `imprimirAdministradores`, `ConsultarAdministradores`.
2. El arreglo `administradores` está correctamente inicializado y poblado antes de cada prueba.

Pasos:

1. Eliminar administrador por ID:

- Ejecutar la función `eliminarAdministrador` con el ID correspondiente.
- Verificar que el administrador se elimine del arreglo `administradores`.

```
it('debería eliminar un administrador por su ID', () => {
  eliminarAdministrador(1);
  expect(administradores.length).toBe(1);
  expect(administradores.some(admin => admin.id === 1)).toBe(false);
});
```

2. Imprimir administradores:

- Llamar a la función `imprimirAdministradores`.
- Verificar que se muestren todos los administradores presentes en el arreglo `administradores`.

```
it('debería imprimir los administradores correctamente', () => {
  const consoleSpy = spyOn(console, 'log');
  imprimirAdministradores();
  expect(consoleSpy).toHaveBeenCalledTimes(administradores.length);
  administradores.forEach(admin => {
    expect(consoleSpy).toHaveBeenCalledWith(`ID: ${admin.id}, Persona_idPersona: ${admin.Persona_idPersona}`);
  });
});
```

3. Eliminar administrador inexistente:

- Ejecutar la función `eliminarAdministrador` con un ID que no existe en el arreglo.
- Verificar que el arreglo `administradores` no se modifique.

```
it('debería manejar correctamente la eliminación de un administrador que no existe', () => {
  eliminarAdministrador(3); // Intentar eliminar un administrador que no existe
  expect(administradores.length).toBe(2); // El arreglo de administradores no debería cambiar
});
```

Resultados Esperados:

- En el primer caso de prueba, el administrador con ID 1 debería ser eliminado correctamente del arreglo `administradores`.
- En el segundo caso de prueba, se espera que la función `imprimirAdministradores` muestre por consola cada administrador presente en `administradores`.
- En el tercer caso de prueba, el sistema debe manejar correctamente la situación de intentar eliminar un administrador que no existe.

Resultados Obtenidos:

- **Caso de prueba 1 (Eliminar administrador por ID):** Se eliminó correctamente el administrador con ID 1. El arreglo `administradores` ahora contiene 1 elemento.
- **Caso de prueba 2 (Imprimir administradores):** La función `imprimirAdministradores` mostró por consola cada administrador presente en el arreglo `administradores`.
- **Caso de prueba 3 (Eliminar administrador inexistente):** El sistema no modificó el arreglo `administradores` al intentar eliminar un administrador que no existe.

```

✓ Browser application bundle generation complete.
27 06 2024 00:22:41.826:WARN [karma]: No captured browser, open http://localhost:9876/
27 06 2024 00:22:43.021:INFO [karma-server]: Karma v6.4.2 server started at http://localhost:9876/
27 06 2024 00:22:43.021:INFO [launcher]: Launching browsers Chrome with concurrency unlimited
27 06 2024 00:22:43.028:INFO [launcher]: Starting browser Chrome
27 06 2024 00:22:44.640:INFO [Chrome 126.0.0.0 (Windows 10)]: Connected on socket cWgCLx3f5E7no0aDAAAB with id 43133489
LOG: 'Administrador con ID 1 eliminado.'
Chrome 126.0.0.0 (Windows 10): Executed 0 of 3 SUCCESS (0 secs / 0 secs)
LOG: 'No se encontró ningún administrador con ID 3.'
Chrome 126.0.0.0 (Windows 10): Executed 0 of 3 SUCCESS (0 secs / 0 secs)
Chrome 126.0.0.0 (Windows 10): Executed 3 of 3 SUCCESS (0.116 secs / 0.007 secs)

```

Observaciones:

- No se encontraron errores durante la ejecución de los casos de prueba.
- Las funciones `eliminarAdministrador` e `imprimirAdministradores` se comportaron según lo esperado en todos los escenarios probados.

Conclusión: Todos los casos de prueba fueron exitosos y las funciones relacionadas con administradores funcionan correctamente según lo especificado.