

AppleScript notification issue

this AppleScript written to close all notifications in Mac OS sequoia suddenly no longer works. think hard about this

On macOS Sequoia a lot of people are finding that old “close all notifications” AppleScripts suddenly stop working — even though they worked fine on Ventura/Sonoma or even earlier Sequoia builds. This is **not** you going crazy; Apple quietly changed things.

Because you didn’t paste your script, I’ll go through the *typical* patterns that break and what’s likely going on, plus workarounds you can try.

1. What has (likely) changed in Sequoia

Most “clear notifications” scripts rely on one of these:

1. UI scripting of Notification Center

Example pattern:

```
applescript
```

```
tell application "System Events"
    tell process "Notification Center"
        click button 1 of window 1 -- or UI element "x" etc.
    end tell
end tell
```

Or:

```
applescript
```

```
tell application "System Events"
    tell process "Notification Centre" -- or "NotificationCenter"
        set theButtons to buttons of scroll area 1 of group 1 of window 1
        repeat with b in theButtons
            if name of b is "Close" then click b
        end if
    end repeat
end tell
```

Problems on Sequoia:

- The **process name** may have changed (e.g. capitalization, or Apple moved notifications into a different process).
- The **AX hierarchy** (windows/groups/scroll areas) has changed, so your script is now pointing at the wrong UI elements.
- The “Close” / “x” buttons may no longer appear in the accessibility tree the same way.

- Sometimes the **Notification Center process isn't running** until you actually open it; scripts that assume it's always there now fail.

2. Using **display notification** and assuming that implies access to Notification Center

Apple's tightened the sandboxing / accessibility on scripted UI interactions. **display notification** still works, but using AppleScript to click around in Notification Center is more fragile.

3. Accessibility permissions reset or invalidated

After OS updates, Settings → Privacy & Security → Accessibility sometimes silently forgets existing permissions for:

- **Script Editor / Script Debugger**
- **osascript**
- Any app/launcher (Keyboard Maestro, Alfred, Automator app, etc.) that runs the script

Symptom: Script runs without error, but no buttons are actually clicked.

2. First sanity checks (these alone fix many cases)

Do these quickly:

1. Open Notification Center manually

On Sequoia, sometimes Notification Center isn't in the UI tree until it's visible.

- Click the time in the menu bar so notifications are actually shown.
- Then run your script.

2. Re-enable Accessibility

- System Settings → Privacy & Security → Accessibility
- Turn OFF then ON:
 - Script Editor / Script Debugger
 - Any app that triggers the script (Keyboard Maestro, Alfred, Automator, FastScripts, etc.)
- Also check **Automation** section and make sure those tools are allowed to control "System Events."

3. Verify process name

In Script Editor, run:

```
applescript
```

```
tell application "System Events" to get name of every process
```

Look for something like:

- "Notification Center"
- Or any similarly named process when Notification Center is open.

If your script says `process "Notification Center"` but the actual name is different (capitalization, extra space, localized language, etc.), that will kill it.

3. The big one: AX hierarchy changes

It's extremely likely the **UI structure of notifications** changed in Sequoia.

How to inspect the actual structure

If you have Xcode installed:

1. Launch **Xcode** → **Open Developer Tool** → **Accessibility Inspector**.
2. Open Notification Center (click time in menu bar).
3. Use the inspector to hover over:
 - Individual notification cards
 - Their close buttons
4. Look at the hierarchy:
 - Which `window`?
 - Which `group` / `scroll area`?
 - What's the role / name of the close button?

Your old script probably references something like:

```
applescript
```

```
buttons of scroll area 1 of group 1 of window 1
```

but on Sequoia it may now be more like:

```
applescript
```

```
buttons of groups of group 1 of scroll area 1 of window "Notification Center"
```

You may also find that the close button:

- Has **no name** now (needs **role description** or **subrole**).
- Or lives in a nested group.

A generic “search and click” pattern that survives small changes

If your old script relied on fixed indexes, try a more defensive pattern like:

applescript

```
tell application "System Events"
    tell process "Notification Center"
        -- adjust window name/index if necessary
        tell window 1
            set closeButtons to (every button whose role description is "close
button" or name is "Close")
            repeat with b in closeButtons
                try
                    click b
                end try
            end repeat
        end tell
    end tell
end tell
```

Sometimes you need to go through **scroll area** / **groups** explicitly:

applescript

```
tell application "System Events"
    tell process "Notification Center"
        tell window 1
            set theButtons to every button of (every group of every group of
scroll area 1)
            repeat with b in theButtons
                try
                    if (role description of b is "close button") or (name of b is
"Close") then
                        click b
                    end if
                end try
            end repeat
        end tell
    end tell
end tell
```

Then iteratively refine based on what Accessibility Inspector shows.

4. If the script errors vs. silently does nothing

- **If you get a real AppleScript error** (e.g. `Can't get window 1 of process "Notification Center"`):
 - Either the process name is wrong, or the window index/name changed.
 - Fix by inspecting the process name and windows:

```
applescript
```

```
tell application "System Events"
    tell process "Notification Center"
        get name of every window
    end tell
end tell
```

- **If there's no error but no notifications close:**
 - Script is targeting the wrong UI elements (wrong groups/scroll areas).
 - Or Accessibility permissions aren't correctly set.

5. Architectural possibility: Apple partially blocked scripting Notification Center

It's also possible Sequoia has:

- Moved notification UI into a more locked-down process / view.
- Marked some elements as non-interactive to accessibility APIs.

You'll see this if, in Accessibility Inspector:

- The close button doesn't show up as a distinct element.
- Or it has no actionable role.

In that case, AppleScript UI scripting will be extremely brittle or impossible for those elements. Then your options are:

- **Keyboard Maestro / BetterTouchTool** macro using screen coordinates + image detection (ugly but often works).
- **Keyboard shortcuts** if Apple ever exposes “Clear all notifications” as a menu item (currently they don’t).

6. What I’d do next in your situation

Since you clearly know what you’re doing, here’s a concrete debugging checklist:

1. **Paste your current script into Script Editor and run it from there**, not from any automation wrapper.
2. **Check Accessibility permissions** for Script Editor & “System Events”.
3. **List processes** and confirm the live name of the Notification Center process.
4. **Use Accessibility Inspector** to:
 - Confirm there *is* a window and scroll area containing notifications.
 - See how the close buttons are represented in the hierarchy.
5. **Rewrite the UI references** using the actual hierarchy, with defensive **every button whose ...** logic instead of hardcoded index chains.

If you’d like, paste your exact AppleScript here and tell me:

- Whether it now fails with an error, or
- Whether it runs but does nothing.

Then I can try to **surgically rewrite** your script for the new Sequoia UI instead of talking in generalities.

```
tell application "System Events"
  try
    set _groups to groups of scroll area 1 of group 1 of
      group 1 of window 1 of application process "NotificationCenter"
    repeat with _group in _groups
      set _actions to actions of _group
      repeat with _action in _actions
        if description of _action is in {"Close", "Clear All"} then
          perform _action
        end if
      end repeat
    end repeat
  end try
end tell
```

Right, this is a known Sequoia breakage – and your script is *almost* the one people have been posting about as “no longer works after Sonoma → Sequoia.”

The good news: there *is* a Sequoia-compatible variant. The bad news: Apple changed the accessibility hierarchy and how the “Close / Clear All” actions are exposed, so your old approach no longer hits them.

What changed vs. your script

Your current script:

```
applescript
```

```
tell application "System Events"
    try
        set _groups to groups of scroll area 1 of group 1 of group 1 of
window 1 of application process "NotificationCenter"
        repeat with _group in _groups

            set _actions to actions of _group

            repeat with _action in _actions
                if description of _action is in {"Close", "Clear
All"} then
                    perform _action
                end if
            end repeat
        end repeat
    end try
end tell
```

Key points that Sequoia broke:

1. Hierarchy

`groups of scroll area 1 of group 1 of group 1 of window 1` is no longer valid – Sequoia moved things around, so you now need to look under `UI element 1 of scroll area 1` ..., not `groups` Scripts that used your exact path are now erroring with “Can’t get scroll area 1 of group 1 ... Invalid index” on Sequoia.

2. Actions exposure

On Sequoia, the “Close” / “Clear All” functionality is best accessed by:

- Finding UI elements whose **name** starts with "Name:Close" or "Name:Clear All", and then
- Performing their actions.

That pattern comes from updated Sequoia scripts shared on BetterTouchTool and later blogged as “works on Sequoia.”

So, instead of iterating **groups** and inspecting **description of _action**, you now ask System Events:

```
“Give me all UI elements under this scroll area whose name starts with Name:Close /  
Name:Clear All, and perform all their actions.”
```

Drop-in Sequoia replacement for your script

Try this version (same spirit, updated to Sequoia’s structure):

applescript

```
tell application "System Events"
    tell process "NotificationCenter"
        try
            -- Make sure the Notification Center window is actually
there
            if exists window "Notification Center" then
                -- This is the new path that works on Sequoia
                if exists scroll area 1 of group 1 of group 1 of
window "Notification Center" then

                    perform (actions of UI elements of ¬
                        UI element 1 of scroll area 1 of
group 1 of group 1 of ¬
                        window "Notification Center" ¬
                        whose name starts with
"Name:Close" ¬
                        or name starts with
"Name:Clear All")

                end if
            end if
        end try
    end tell
end tell
```

Notes / differences vs. your original:

- Uses **window "Notification Center"** instead of **window 1** – less brittle.
- Looks at **UI elements of UI element 1 of scroll area 1** instead of **groups of scroll area 1**.
- Filters by **name starts with "Name:Close" / "Name:Clear All"** and performs all associated actions.
- Keeps the **try** so it fails silently like your original (for debugging, temporarily remove the **try/end try** to see the exact error in Script Editor).

This is essentially the pattern that's been confirmed to work on Sequoia 15.x.

Things to be aware of

1. Notification Center must be visible

The script only sees that window if Notification Center is actually open (click the clock / date or use your usual hotkey first).

2. Language / localization

The `"Name:Close"` / `"Name:Clear All"` strings are from an English-language setup. If your macOS UI is in another language, the prefix may still be `Name:` but the word after might be localized; in that case you'd need to adjust those two strings.

3. Only handles visible notifications

Like your old script, this acts on the notifications currently shown in Notification Center, not the ones that are just momentary banners and already gone from the screen. (There's a more elaborate variant that synthesizes an extra notification so "Clear All" always appears, as in that blog post; I've kept your version simple here.)

If this still does nothing

For deeper debugging, do this once:

```
applescript
```

```
tell application "System Events"
    tell process "NotificationCenter"
        get name of every window
    end tell
end tell
```

- If `"Notification Center"` isn't in that list, Apple changed the window name on your system and we can adjust the script accordingly.
- If you get an error like "Can't get scroll area 1 of group 1... Invalid index", it means the hierarchy has shifted again, and we'd then switch to a more generic recursive search over all `UI elements` in window `"Notification Center"` and look for those `Name:Close/Name:Clear All` elements.

But in current Sequoia builds, the script above should work as a near drop-in replacement for what you had.