

Stylable <select>

Joey Arhar <jarhar@chromium.org>

2024-04-30

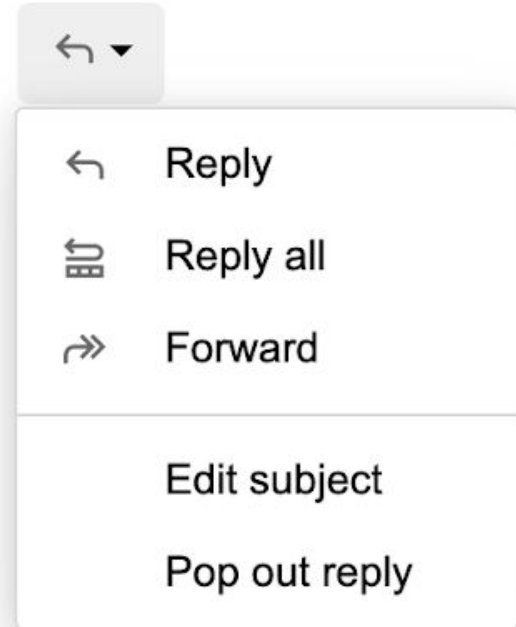
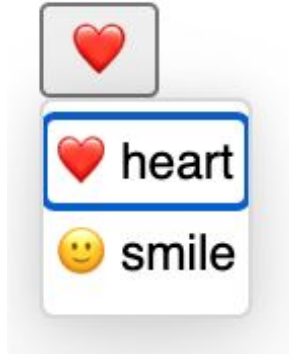
<https://open-ui.org/components/selectlist/>

Background

- Developers don't use the `<select>` element because it isn't stylable enough
 - Developers create their own which have issues:
 - Accessibility problems and inconsistencies
 - They don't work with autofill
 - <https://www.gwhitworth.com/posts/2019/form-controls-components/>
 - <https://github.com/MicrosoftEdge/MSEdgeExplainers/blob/main/ControlUICustomization/explainer.md>
- Microsoft and Google started a project in 2019 to address this by making the `<select>` element more stylable
- We have been incubating this proposal in OpenUI ever since
 - [83 <select> issues discussed so far](#)
- Builds on popover and anchor positioning

Use cases

- Arbitrary content everywhere
- Declaratively clone content into button (via the `<selectedoption>` element)
- Different styling in button
- Animations



Requirements

- Everything is accessible
- Interoperable and well defined default styles, box tree, and pseudo-elements
- Interoperable keyboard/mouse behavior
- Declarative (no javascript needed)

Opt-in history (mostly in OpenUI)

1. New element: `<selectmenu>` then `<selectlist>`
2. Child content: having `<button>` or `<datalist>` in `<select>`
3. HTML attribute: `<select base>`
4. CSS property: `appearance:base-select`

Simple example

```
<select style="appearance:base-select">  
  <option>one</option>  
  <option>two</option>  
</select>
```

Proposed API shape

No HTML changes are needed to use new CSS capabilities, but HTML changes are needed to support arbitrary HTML content:

```
<select style="appearance:base-select">
  <button>
    <selectedoption></selectedoption>
  </button>
  <datalist>
    <option><img src=...>one</option>
    <option><img src=...>two</option>
  </datalist>
</select>
```

<select> open agenda items

1. [\[css-ui\] appearance: base to enable interoperable styling of controls/components #5998](#)
2. [\[css-ui\] Changing UA styles based on the computed value of the appearance property #10028](#)
3. [How to implement and shape API for <selectedoption> element for <select> #10242](#)