Questions and Answers:

**1) Changes observation**

Imagine you have a variable, let’s say a number from 1 to 10:

var a = 1;

And you have some function that do something:

function b() { … }

I want to call function B every time when variable A changes its value.

Question: how can I observe variable A for changes?

Answer: Function called object.watch(). The function will watch the variable. The function will be called whenever the variable value changed.

Another way we can use set intervals for to check variable value.

As per standard to change the variable we need to use get set method. If we do the coding whenever the variable changes.

**2) Drop down menu without JS**

Imagine, that you have a website with a menu. And each menu item has a submenu. And I this submenu to works as a drop down. When I hover over with the mouse over menu item drop down list with submenu shows up. When I move mouse out - drop down disappears.

Now, let’s say I want to make this menu touch friendly. Instead of showing drop down on mouse hover, I want to show it on mouse click or on finger tap. So, I tap menu item - submenu shows up. I tap again - it hides back.

Question: how can I do this without using any JS?

Answer: We can use style to display the drop-down class name: hover. In the following code samples, which is clarified.

**3) Image rendering performance**

Imagine you are building a search engine and all UI you have on the web page is search widget. And your search button is rendered as a simple standard 2 colors magnified glass icon.

Question: What would you use .png or .svg image to render it?

**Follow up:**what if instead of one icon, you have to render 100 different icons on the page, but all of them are small, less than one kilobyte?

Answer:

Subject: Selecting the image type for the search engine UI widget

Image rendering: Image-rendering CSS property provides a hint to the browser about the algorithm it should use to scale images. It applies to the element itself as well as any images supplied in other properties for the element. It has no effect on non-scaled images.

I would be using the .svg over .png because and mentioned the reasons for using the it below :

1) .svg images follow the vector based customizations. we can Make them as big or as small as you want, change the colors, or even change the shapes themselves.

2) .svg images are faster at loading than compared to .png file.

3) We can modify the colors of .svg file but for .png we can’t as they are fixed .

4) SVG’s can be animated and styled with CSS. Animations.

5) SVG’s are just code, which means very small file sizes.

If there are even more icons to be rendered on a page, I would prefer to use the .svg image over .png because of its benefits mentioned above.