Const themeButton = document.getElementById(‘theme-button’)

Const darkTheme = ‘dark-theme’

Const iconTheme = ‘bx-sun’

// Previously selected topic (if user selected)

Const selectedTheme = localStorage.getItem(‘selected-theme’)

Const selectedIcon = localStorage.getItem(‘selected-icon’)

// We obtain the current theme that the interface has by validating the dark-theme class

Const getCurrentTheme = () => document.body.classList.contains(darkTheme) ? ‘dark’ : ‘light’

Const getCurrentIcon = () => themeButton.classList.contains(iconTheme) ? ‘bx-moon’ : ‘bx-sun’

// We validate if the user previously chose a topic

If (selectedTheme) {

// If the validation is fulfilled, we ask what the issue was to know if we activated or deactivated the dark

Document.body.classList[selectedTheme === ‘dark’ ? ‘add’ : ‘remove’](darkTheme)

themeButton.classList[selectedIcon === ‘bx-moon’ ? ‘add’ : ‘remove’](iconTheme)

}

// Activate / deactivate the theme manually with the button

themeButton.addEventListener(‘click’, () => {

// Add or remove the dark / icon theme

Document.body.classList.toggle(darkTheme)

themeButton.classList.toggle(iconTheme)

// We save the theme and the current icon that the user chose

localStorage.setItem(‘selected-theme’, getCurrentTheme())

localStorage.setItem(‘selected-icon’, getCurrentIcon())

})