```javascript
'use strict';
/////////// Lecture 79: Modal window

// We can store the results of the selected element in a variable.
const modal = document.querySelector('.modal');
const overlay = document.querySelector('.overlay');
const btnCloseModal = document.querySelector('.close-modal');
// We would like to select the 'show-modal' class but there are 3 elements with the same class name.
const btnOpenModal = document.querySelector('.show-modal'); //==> Only selects the first button!
console.log(btnOpenModal);
//This is the limitation of the querySelector method. If there are more than one object with the same class, querySelector will select the first one.

// There is a better one however, it's called querySelectorAll.
const btnsOpenModal = document.querySelectorAll('.show-modal'); // ==> Returns a Node-list!
console.log(btnsOpenModal);

for (let i = 0; i < btnsOpenModal.length; i++)
  console.log(btnsOpenModal[i].textContent);

// Just 1 line - no need curly braces!

///////////// Lecture 80: Working with Classes
//Attach event handlers to each of these buttons:
for (let i = 0; i < btnsOpenModal.length; i++)
  btnsOpenModal[i].addEventListener('click', function () {
    console.log(btnsOpenModal[i].textContent);
    // Using JS to modify display settings
    // Classlist has a lots of properties, one of its methods is called 'remove'
    modal.classList.remove('hidden');
    //DO NOT USE THE DOT here.
    // Error: modal.classList(.remove('.hidden')); // No!
    // If you would like more than 1 class, then you can
    // do this:
    // modal.classList.remove('class1', 'class2',...);
    overlay.classList.remove('hidden');
    /*
    This is similar to doing something like:
    modal.style.display = block;
    But imagine the class had like 10 properties:
    then we would have to write all these properties
    manually and change all their values. So that's a lots of work, and we would aggregate all these properties into a class, that we then define here in CSS, and we add or remove these classes s we add or remove each style.
    */
    // We could
  });

// Add functionality to close the window, or add the hidden class back to it.
/*
btnCloseModal.addEventListener('click', function () {
  modal.classList.add('hidden');
  overlay.classList.add('hidden');
});

overlay.addEventListener('click', function () {
  overlay.classList.add('hidden');
  modal.classList.add('hidden');
});
*/
// this works! but yoooo don't repeat yourself.

const closeModal = function () {
  modal.classList.add('hidden');
  overlay.classList.add('hidden');
};
btnCloseModal.addEventListener('click', closeModal);
overlay.addEventListener('click', closeModal);
// You can do this for the one that opens the model too!!
```

```

///////////// Lecture 81: Handling an 'ESC' Keypress event

//Responding keyboard events using addEventListener

// These events are global events, so they listen to the whole DOM.
document.addEventListener('keydown', function () {
  console.log('Hello!');
});
// As we hit any key on the keyboard now, the function will be fired. this is because upon keydown, a key down event is generated and our handler function is waiting for this event to happen. And anytime that an event like this occurs, JS does in fact generate an object, and that object contains all the information about the object itself, and then we can actually access that object in the eventhandler function.

// We can have access to information about that event in the event handler function just like this one. Up until this point, we have never examined the event object, but we need to examine it in order to determine which key was the one that has been pressed.

// Obtaining the EVENT Object
document.addEventListener('keydown', function (e) {
  console.log(e.key); // this is an object generated by JS.
});

// Note that when the Escape key is pressed, JS calls the ESC button 'Escape'

// Adding the closing of the modal window upon keydown Escape:
// To do this, I will also want to know if the modal class is visible, so I will do this only when it DOESN'T contain the class 'hidden'.
```

```javascript
document.addEventListener('keydown', function (e) {
  if (e.key == `Escape`) {
    // We can check if an element already has a certain class.
    if (!modal.classList.contains('hidden'))
      // no dot
      closeModal(); // explicit function call
  }
});

```