import {

$,

swal,

removeSwal,

$$,

CLASS\_NAMES,

} from './utils';

const {

MODAL,

OVERLAY,

MODAL\_TITLE,

MODAL\_TEXT,

ICON,

FOOTER,

} = CLASS\_NAMES;

afterEach(() => removeSwal());

describe("init", () => {

test("adds elements on first call", () => {

expect($$(OVERLAY).length).toEqual(0);

swal("Hello world!");

expect($$(OVERLAY).length).toBe(1);

expect($$(MODAL).length).toBe(1);

});

});

describe("string parameters", () => {

test("shows text when using 1 param", () => {

swal("Hello world!");

expect($$(MODAL\_TEXT).is(':first-child')).toBeTruthy();

expect($$(MODAL\_TEXT).text()).toBe("Hello world!");

expect($$(MODAL\_TEXT).next().hasClass(FOOTER)).toBeTruthy();

});

test("shows title and text when using 2 params", () => {

swal("Title", "text");

expect($$(MODAL\_TITLE).is(':first-child')).toBeTruthy();

expect($$(MODAL\_TITLE).text()).toBe("Title");

expect($$(MODAL\_TEXT).text()).toBe("text");

expect($$(MODAL\_TEXT).next().hasClass(FOOTER)).toBeTruthy();

});

test("shows icon, title and text when using 3 params", () => {

swal("Oops", "text", "error");

expect($$(ICON).is(':first-child')).toBeTruthy();

expect($$(ICON).hasClass(`${ICON}--error`)).toBeTruthy();

expect($$(MODAL\_TITLE).is(':nth-child(2)')).toBeTruthy();

expect($$(MODAL\_TITLE).text()).toBe("Oops");

expect($$(MODAL\_TEXT).is(':nth-child(3)')).toBeTruthy();

expect($$(MODAL\_TEXT).text()).toBe("text");

expect($$(MODAL\_TEXT).next().hasClass(FOOTER)).toBeTruthy();

});

});