**Міністерство освіти і науки України**

**Черкаський державний технологічний університет**

**Факультет інформаційних технологій і систем**

**Кафедра інформаційних технологій проектування**

Дисципліна: Вебпрограмування зJavaScript

**З В І Т**

**з лабораторної роботи №5**

студента 2-го курсу спеціальності 172«Електроні комунікації та радіотехніка»

***Галайди Олександра Вікторовича***

Виконав \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

Оцінка  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Перевірено \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Перевірила \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /к.т.н., доцент Лавданська О.В. /

(підпис) (прізвище та ініціали)

Черкаси – 2023 року

**Мета роботи**: вивчити засоби по роботі з об’єктною моделлю документа; навчитися додавати, змінювати та видаляти елементи веб-сторінки; засвоїти принципи роботи з CSS-стилями.

**Порядок виконання роботи**

1. Ознайомитися із теоретичними відомостями.
2. Виконати завдання до лабораторної роботи.
3. Скласти та оформити звіт.

**Теоретичні відомості**

[DOM дерево](https://uk.javascript.info/dom-nodes)

[Навігація по DOM](https://uk.javascript.info/dom-navigation)

[Пошук: getElement\*, querySelector\*](https://uk.javascript.info/searching-elements-dom)

[Властивості вузлів: тип, тег та вміст](https://uk.javascript.info/basic-dom-node-properties)

[Атрибути та властивості](https://uk.javascript.info/dom-attributes-and-properties)

[Внесення змін в документ](https://uk.javascript.info/modifying-document)

1. **isElementVisible**
2. export function isElementVisible(element) {
3. switch(element){
4. case element === null || element === undefined:
5. return false
6. case element.offsetWidth || element.offsetHeight:
7. return true
8. default:
9. return 'error'
10. }
11. }

**2. isElementScrolle**

export function isElementScrolled(element) {

    if (!element) {

      return {

        scrollTop: null,

        scrollLeft: null,

        isScrolled: null,

      }

    }

    const scrollable = {

      scrollTop: element.scrollTop,

      scrollLeft: element.scrollLeft,

      isScrolled: (element.scrollTop > 0 || element.scrollLeft > 0),

    }

    return scrollable

  }

1. **getPageData**
2. export function getPageData() {
3. const doc = document.documentElement
4. const body = document.body
6. const result = {
7. windowHeight: window.innerHeight || doc.clientHeight || body.clientHeight,
8. windowWidth: window.innerWidth || doc.clientWidth || body.clientWidth,
9. documentHeight: Math.max(
10. body.scrollHeight,
11. body.offsetHeight,
12. doc.clientHeight,
13. doc.scrollHeight,
14. doc.offsetHeight
15. ),
16. documentWidth: Math.max(
17. body.scrollWidth,
18. body.offsetWidth,
19. doc.clientWidth,
20. doc.scrollWidth,
21. doc.offsetWidth
22. ),
23. currentScrollFromTop: (window.pageYOffset || doc.scrollTop) - (doc.clientTop || 0),
24. currentScrollFromLeft: (window.pageXOffset || doc.scrollLeft) - (doc.clientLeft || 0),
25. }
27. return result
28. }
30. **createToast**
31. export function createToast(element) {
32. const toast = document.createElement('div')
33. toast.classList.add('toast')
34. toast.style.position = 'fixed'
35. toast.style.top = '20px'
36. toast.style.right = '20px'
38. if (element) {
39. toast.appendChild(element)
40. }
42. return toast
43. }

**5. createBlurredCoverElement**

export function createBlurredCoverElement(elementToCover) {

    const cover = document.createElement('div');

    cover.classList.add('cover');

    const elementRect = elementToCover.getBoundingClientRect();

    cover.style.position = 'absolute';

    cover.style.top = `${elementRect.top + window.scrollY}px`;

    cover.style.left = `${elementRect.left + window.scrollX}px`;

    cover.style.width = `${elementRect.width}px`;

    cover.style.height = `${elementRect.height}px`;

    return cover;

  }

**Висновок:**Я вивчив засоби по роботі з об’єктною моделлю документа, навчився додавати, змінювати та видаляти елементи веб-сторінки і засвоїв принципи роботи з CSS-стилями