



**ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ  
УНІВЕРСИТЕТ**  
**Факультет інформаційних технологій**  
**Кафедра комп'ютерних наук та інформаційних систем**

**СИЛАБУС (SYLLABUS)**  
**Дисципліна «Програмування JavaScript /**  
**JavaScript programming»**

**ІНФОРМАЦІЯ ПРО ВИКЛАДАЧА**

Викладач	Базурін Віталій Миколайович
Науковий ступінь	Кандидат педагогічних наук
Вчене звання	Доцент
Посада	Доцент кафедри комп'ютерних наук та інформаційних систем
Адреса кафедри	м. Київ, вул. Кіото 19, каб. Б-507, Б-526
E-mail	compdep@knute.edu.ua
Консультації	Відповідно до графіку індивідуальних консультацій на сайті кафедри

**ПОЛІТИКА АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ**

<https://knute.edu.ua/file/NjY4NQ==/bf27ad9293fa2bb6f9b2c3031d4b6e4a.pdf>

**Дотримання академічної доброчесності передбачає:**

- самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);
- посилання на джерела інформації у разі використання не авторських ідей, розробок, тверджень, відомостей і т.п.;
- дотримання норм законодавства про авторське право і суміжні права;
- надання достовірної інформації про результати власної наукової діяльності, використанні методики досліджень і джерела інформації.

**Порушенням академічної доброчесності вважається:**

- академічний плагіат – оприлюднення (частково або повністю) наукових (творчих) результатів, отриманих іншими особами, як результатів власного дослідження (творчості) та/або відтворення опублікованих текстів (оприлюднених творів мистецтва) інших авторів без зазначення авторства;
- самоплагіат – оприлюднення (частково або повністю) власних раніше опублікованих наукових результатів як нових наукових результатів;
- фабрикація – вигадкування даних чи фактів, що використовуються в наукових дослідженнях;
- фальсифікація – свідомо зміна чи модифікація вже наявних даних, що стосуються наукових досліджень.

**За порушення академічної доброчесності здобувачі освіти можуть бути притягнені до академічної відповідальності:**

- повторне проходження оцінювання (модульний контроль, іспит, залік тощо);
- повторне проходження відповідного освітнього компонента освітньо-професійної програми;
- відрахування з Університету;

- позбавлення наданих університетом пільг;
  - відмова у присудженні відповідного ступеня вищої освіти;
- ПОЛІТИКА ЩОДО ВІДВІДУВАННЯ ЗАНЯТЬ**
- відвідування занять є обов'язковим;
  - за об'єктивних причин (наприклад, хвороба, міжнародне стажування та ін.) навчання може відбуватись в он-лайн формі за погодженням із викладачем дисципліни.

## ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назва дисципліни / тип дисципліни	Програмування JavaScript / обов'язкова
Навчальний рік	2023-2024
Факультет	Факультет інформаційних технологій
Курс	3
Семестр	5
Освітній ступінь	Бакалавр
Галузь знань	12 «Інформаційні технології»
Спеціальність	126 «Інформаційні системи та технології»
Загальна характеристика	Кількість годин –180 Кількість кредитів – 6 <b>Види занять:</b> лекції, лабораторні, самостійна робота. <b>Співвідношення аудиторних годин і годин самостійної роботи -</b> 70/110 <b>Мова викладання</b> – українська <b>Форма викладання</b> – очна
Підсумковий контроль	Екзамен
Програмне забезпечення	Visual Studio, Notepad++
Обладнання	Проектор, комп'ютерна техніка із встановленим програмним забезпеченням та доступом до мережі Інтернет.
Необхідні попередні дисципліни	«Основи теорії інформаційних систем»; «Інформаційні системи та технології», «Алгоритмізація та програмування», «Теорія систем і системний аналіз»
Методика вивчення	Методика вивчення дисципліни полягає у набутті студентами знань теоретичного і практично-прикладного характеру під час лекцій, лабораторних занять, самостійної роботи та вивчення першоджерел і навчально-методичної літератури.
Мета і завдання	<b>Метою</b> вивчення дисципліни «Програмування JavaScript» є надання студентам знань з основ синтаксису і семантики мови сценаріїв JavaScript, вивчення основних алгоритмічних конструкцій і особливостей обробки основних типів даних у сценаріях JavaScript, а також принципів і особливостей роботи з Dynamic Object Model і Browser Object Model. <b>Завданням</b> вивчення дисципліни «Програмування JavaScript» є ознайомлення студентів з призначенням і особливостями впровадження сценаріїв JavaScript у код web-сторінки, особливостями програмування сценаріїв на мові JavaScript з використанням основних алгоритмічних конструкцій і основних типів даних (цілі і дійсні числа, рядки, символи, масиви, колекції, множини тощо), основами об'єктно-

	орієнтованого програмування на мові JavaScript, особливостями програмування сценаріїв для роботи з Dynamic Object Model і Browser Object Model.
<b>Місце дисципліни в освітньо-професійній програмі</b>	
<b>Загальні компетентності</b>	КЗ 2 Здатність застосовувати знання у практичних ситуаціях. КЗ 5 Здатність вчитися і оволодівати сучасними знаннями.
<b>Фахові компетентності (результати навчання)</b>	КС 3 Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, Інтернету речей (IoT), комп'ютерно-інтегрованих систем та системної мережної структури, управління ними. КС 4. Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші). КС 12 Здатність управляти та користуватися сучасними інформаційно-комунікаційними системами та технологіями (у тому числі такими, що базуються на використанні Інтернет).
<b>Програмні результати навчання</b>	ПР 3 Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій. ПР 5 Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій. ПР 6 Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності. ПР 7 Обґрунтовувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій.

## ТЕМАТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### Тема 1. Вступ до JavaScript

Мета і завдання дисципліни. Поняття про JavaScript. Перша програма на JavaScript. Виконання коду JavaScript. Підключення зовнішнього файлу JavaScript. Консоль браузера і console.log.

Змінні та константи. Типи даних. Арифметичні операції. Порозрядні операції. Умовні вирази. Умовні оператори ? і ??. Перетворення даних. Масиви. Умовні конструкції. Цикли. Налаштування програм.

### Тема 2. Функційне програмування.

Функції. Параметри. Результат функції. Стрілкові функції. Область видимості змінних. Замикання і функція IIFE. Паттерн Модуль. Рекурсивні функції. Перевизначення функцій. Hoisting. Передача параметрів за значенням і посиланням.

### **Тема 3. Об'єктно-орієнтовне програмування.**

Об'єктно-орієнтовне програмування. Об'єкти. Ключове слово `this`. Вкладені об'єкти і масиви в об'єктах. Копіювання і порівняння об'єктів. Перевірка наявності і перебір методів і властивостей. Об'єкти в функціях. Конструктори об'єктів. Розширення об'єктів. Прототипи. Функція `Object.create`. Конфігурація властивостей об'єктів. Функція як об'єкт. Методи `call` і `apply`. Успадкування прототипів. Деструктуризація. Оператор `?` Константні об'єкти. Заборона зміни об'єкта

Класи. Приватні поля і методи. Статичні поля і методи. Властивості і методи доступу. Успадкування.

### **Тема 4. Масиви.**

Створення масиву і об'єкт `Array`. Масиви і `spread`-оператор. Операції з масивами. Успадкування масивів.

### **Тема 5. Рядки і регулярні вирази.**

Рядки і об'єкт `String`. Шаблони рядків. Об'єкт `RegExp`. Регулярні вирази. Синтаксис регулярних виразів. Квантифікатори в регулярних виразах. Пошук у рядку. Групи в регулярних виразах. Регулярні вирази в методах `String`.

### **Тема 6. Обробка помилок.**

Конструкція `try...catch..finally`. Генерація помилок і оператор `throw`. Типи помилок. Обробка помилок і стек виклику функцій.

### **Тема 7. Вбудовані об'єкти.**

Об'єкт `Date`. Робота з датами. Об'єкт `Math`. Математичні операції. Об'єкт `Number`. Символи. `Proxy`.

### **Тема 8. Колекції і ітератори.**

Ітератори. Генератори. Множини `Set`. `Map`. `WeakSet`. `WeakMap`.

### **Тема 9. Робота з DOM.**

Робота з DOM. Вступ до DOM. Властивості об'єкта `document`. Пошук елементів на веб-сторінці. Об'єкт `Node`. Навігація по DOM. Елементи. Створення, додавання і видалення елементів. Керування атрибутами елементів. Керування стилями і класами елементів. Створення свого елемента в HTML.

Події. Вступ до обробки подій. Обробники подій. Передача даних в обробник подій. Об'єкт `Event`. Поширення подій. Події миші. Події клавіатури. Програмний виклик подій. Визначення своїх подій.

### **Тема 10. Робота з формами**

Форми і їх елементи. Кнопки. Текстові поля. Прапорці і радіокнопки. Список `select`. `Validation API`. Валідація елементів форми. Керування валідацією форм.

### **Тема 11. Робота з браузером і BOM.**

`Browser Object Model` і об'єкт `window`. Діалогові вікна і пошук на сторінці. Відкриття, закриття і позиціонування вікон. Історія браузера. `History API`. Об'єкт `location`. Об'єкт `navigator`. Отримання інформації про браузер. Таймери.

### **Перелік навчальних робіт з дисципліни «Основи теорії інформаційних систем»**

Види робіт	К-сть балів
------------	-------------

Лабораторне заняття №1. Тема: «Прості обчислення».	2
Лабораторне заняття №2. Тема: «Логічні операції. розгалуження».	2
Лабораторне заняття №3. Тема: «Функції».	2
Лабораторне заняття №4. Тема: «Рекурсивні функції».	2
Лабораторне заняття №5. Тема «Основи ООП і JavaScript».	2
Лабораторне заняття №6. Тема: «Створення сценарію з використанням основних концепцій ООП».	2
Лабораторне заняття №7. Тема: «Масиви».	4
Лабораторне заняття №8. Тема: «Двовимірні масиви».	4
Лабораторне заняття №9. Тема: «Рядки і регулярні вирази».	4
Лабораторне заняття №10. Тема: «Обробка помилок».	4
Лабораторне заняття №11. Тема: «Вбудовані об'єкти».	4
Лабораторне заняття №12. Тема: «Колекції і ітератори».	4
Лабораторне заняття №13. Тема: «Робота з DOM».	4
Лабораторне заняття №14. Тема: «Створення власного елемента HTML».	4
Лабораторне заняття №15. Тема: «Обробка подій».	2
Лабораторне заняття №16. Тема: «Створення тесту».	2
Лабораторне заняття №17. Тема: «Створення сценарію для роботи з вікном браузера».	2
Модульний контроль	10
Виконання індивідуального завдання (СР)	20
<b>Разом: Аудиторна робота</b>	<b>80</b>
<b>Самостійна робота (СР)</b>	<b>20</b>
<b>Всього:</b>	<b>100</b>

### КОНТРОЛЬ ТА КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАТЬ СТУДЕНТІВ

При вивченні дисципліни використовуються наступні форми контролю знань студентів: поточний; модульний; підсумковий.

**Поточний контроль** передбачає перевірку теоретичних питань, самостійної роботи, практичних робіт та усне опитування по кожній практичній роботі. По даному виду контролю оцінювання знань здійснюється у відповідності до бального розподілу наведеного в попередній таблиці.

**Модульний контроль** передбачає виконання модульної контрольної роботи. Всі завдання оцінюються в 10 балів. Перше завдання (теоретичне) – 3 бали, друге завдання (практичне) – 3 бали, третє завдання (практичне) – 4 бали.

**Формою підсумкового контролю** є екзамен. Екзаменаційна оцінка (100 балів) є результатом виконання двох теоретичних питань (2 x 20 балів = 40 балів) та практичного завдання (60 балів).

**Результуюча оцінка з дисципліни** визначається як середня від балів набраних протягом семестру та отриманих на іспиті.

### **СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ**

#### **Основний:**

1. Bocharov B.P., Levikov Y.V., Voevodina M.Y. Scripting programming languages. O.M.Beketov National University of Urban Economy in Kharkiv. – Kharkiv : O. M. Beketov NUUE in Kharkiv, 2021. – 109 p.
2. Flanagan D. JavaScript. O'Reilly Media, Incorporated, 2021. – 600 p.
3. Rauschmayer A. Deep JavaScript. Theory and techniques. 2020. – 400 p.
4. Rauschmayer A. JavaScript for impatient programmers (ES2022 edition). 2022. – 560 p.
5. Баран С.В. Основи web-програмування: навчальний посібник. – Кривий Ріг: Державний університет економіки і технологій, 2023. –316 с.
6. Двірничук К.В., Вацек Д.О. Веб-програмування та веб-дизайн : навч. посіб. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича, 2022. – 472 с.