

**ПРИВАТНЕ АКЦІОНЕРНЕ ТОВАРИСТВО «ВИШИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«МІЖРЕГІОНАЛЬНА АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ»**



**МАУП**

**НАВЧАЛЬНА ПРОГРАМА**  
**дисципліни**  
**“МОВА СЦЕНАРІЇВ JAVASCRIPT”**  
**(для бакалаврів)**

Київ 2019

Підготовлено викладачем кафедри комп'ютерних інформаційних систем та технологій  
Т.С. Домків.

Затверджено на засіданні кафедри комп'ютерних інформаційних систем та технологій  
(Протокол № 7 від 16 січня 2019 р.)

Схвалено Вченою радою Міжрегіональної Академії управління персоналом

**Домків Т.С.** Навчальна програма дисципліни “МОВА СЦЕНАРІЇВ JAVASCRIPT” (для  
освітньо-кваліфікаційного рівня «бакалавр»). — К.: МАУП, 2019— 16 с.

Навчальна програма містить пояснювальну записку, тематичний план, зміст дисципліни,  
тематику контрольних робіт, вказівки до виконання контрольних робіт, питання для  
самоконтролю, список літератури.

© ПрАТ «ВНЗ «Міжрегіональна Академія управління персоналом» 2019

## ПОЯСНЮВАЛЬНА ЗАПИСКА

**Метою дисципліни** «Мова сценаріїв JavaScript» є засвоєння необхідних знань з основ веб-технологій, а також формування практичних навичок щодо розробки якісних веб-сайтів.

Для досягнення мети поставлені такі основні **завдання**:

- отримання теоретичних знань з основ мови сценаріїв JavaScript, веб-дизайну та веб-програмування;
- отримання практичних навичок з розробки веб-сайтів.

**Об'єктом** навчальної дисципліни є глобальна мережа Інтернет та процеси, що в ній відбуваються.

**Предметом** вивчення дисципліни є веб-технології, а також методи їх використання при розробці сайтів різноманітного призначення. З

### **Вимоги до знань та вмінь**

*Студент повинен знати:*

- правила побудови документів HTML;
- основні властивості каскадних таблиць стилів;
- основні елементи об'єктної моделі браузера;
- правила побудови і основні елементи мови JavaScript;

*Студент повинен вміти:*

- створити сайт (мінімум з 5 сторінок);
- використати засоби каскадних таблиць стилів для оформлення сторінки;
- використовувати основні об'єкти браузера та основні наперед визначені об'єкти JavaScript

Загальний обсяг дисципліни – 90 годин (3 кредити ЄКТС)

З них: 15 год лекції, 30 год практичні та семінарські заняття, 45 год самостійна робота.

Дана дисципліна не є обов'язковою для вивчення.

## ТЕМАТИЧНИЙ ПЛАН ДИСЦИПЛІНИ

### «МОВА СЦЕНАРІЇВ JAVASCRIPT»

№/п.п.	Назва змістового модуля та теми
Змістовий модуль 1. Основи створення статичних веб-сайтів	
Тема 1	Мова гіпертекстової розмітки HTML
Тема 2	Каскадні таблиці стилів CSS
Тема 3	Створення HTML сторінок
Тема 4	Мова програмування JavaScript
Тема 5	Робота з об'єктами в JavaScript. Бібліотека jQuery
Тема 6	Робота з формами
Тема 7	Адаптивний інтерфейс користувача
Змістовий модуль 2. Основи створення динамічних веб-сайтів	
Тема 8	Основи мови програмування PHP
Тема 9	Об'єктно-орієнтовне програмування в PHP
Тема 10	Розміщення веб-додатків в Інтернеті
Тема 11	Автономна робота
	<b>Разом годин : 90</b>

# **ЗМІСТ ДИСЦИПЛІНИ**

## **Змістовий модуль 1. Основи створення статичних веб-сайтів**

### **Тема 1. Мова гіпертекстової розмітки HTML**

Всесвітнє павутиння. URL-адреса. Веб-браузери. Протокол HTTP. Основи HTML5. Мова гіпертекстової розмітки HTML. Семантична розмітка. Робота з елементами. Створення HTML документа. Вбудовування вмісту. Вбудовування HTML за допомогою фреймів. Робота з гіперпосиланнями. Додавання зображень в HTML документі. Вбудовування вмісту з плагінів.

### **Тема 2. Каскадні таблиці стилів CSS**

Знайомство з CSS. Визначення та застосування стилю. Додавання коментарів у таблицю стилів. Створення внутрішнього стилю. Створення вбудованого стилю. Створення зовнішньої таблиці стилів. Селектори, специфічність і каскадування. Визначення селекторів. Розуміння вбудованих стилів браузера. Розширення стилів браузера стилями користувача. Правило !important. Каскадування стилів. Специфічність. Спадкування. Робота з властивостями CSS. Робота з кольорами CSS. Робота з текстом. Робота з блочною моделлю. Встановлення межі, полів і відступів. Позиціонування елемента <div>. Використання властивості float. Використання властивості clear. Використання властивості box-sizing. Центрування вмісту у вікні браузера.

### **Тема 3. Створення HTML сторінок**

Семантика HTML. Чому семантична розмітка? Підтримка браузерів для HTML5. Створення семантичних документів HTML5. Створення макету сайту HTML5. Управління форматуванням за допомогою елемента <div>. Додавання тематичних розривів. Анотування вмісту. Робота зі списками. Робота з таблицями. Неправильне використання таблиць. Створення базової

таблиці. Додавання заголовків. Стилiзація заголовків таблиці. Визначення верхнього і нижнього колонтитулів та тіла таблиці. Створення нестандартних таблиць. Додавання підпису до таблиці. Стилiзація колонок.

#### **Тема 4. Мова програмування JavaScript**

Основи JavaScript. Розуміння JavaScript. Розуміння ролі даних. Використання інструкцій. Робота з функціями. Область видимості змінних. Вкладені функції і вкладені локальна змінна оглядовий. Перетворення до іншого типу. Умовне програмування. Реалізація циклів. Обробка помилок. Написання, тестування і відлагодження JavaScript. Привіт світ від JavaScript. Використання тегу script. Робота з браузерами, які не підтримують JavaScript. Вбудований JavaScript проти зовнішніх файлів JavaScript. Розміщення ваших елементів скрипту. Використання відладчиків JavaScript.

#### **Тема 5. Робота з об'єктами в JavaScript. Бібліотека jQuery**

Робота з об'єктами. Робота з масивами. Доступ до об'єктів DOM. Створення об'єктів JavaScript. Використання об'єктно-орієнтованої термінології. Розуміння об'єктно-орієнтованого застереження JavaScript. Використання літерального шаблону об'єкта JavaScript. Створення динамічних об'єктів за допомогою шаблону factory. Створення класу. Використовуючи властивості prototype. Обговорення компромісу prototype/private. Реалізація простору імен. Реалізація спадкування Робота з бібліотекою jQuery. Основи jQuery. Початок роботи з jQuery. Використання jQuery. Створення оболонки посиланням на елемент DOM для jQuery. Додавання обробників подій. Запуск обробників подій. Ініціалізація коду коли браузер готовий.

#### **Тема 6. Робота з формами**

Розуміння форм. Розуміння веб-комунікації. Відправка даних форми на вебсервер. Передача даних при відправці форми. Використання елемента <label>. Визначення батьківських форм. Запуск подання форми. Серіалізація

форми. Використання атрибуту autofocus. Використання обмежень представлення даних. Використання POST або GET. Перевірка форм. Необхідна перевірка. Перевірка вхідних URL. Перевірка чисел і діапазонів. Стилзація перевірки.

## **Тема 7. Адаптивний інтерфейс користувача**

Підтримка декількох форм-факторів. Для чого розробляти адаптивний інтерфейс користувача? Міркування для підтримки різних типів пристроїв. Створення адаптивного інтерфейсу користувача. Типи media в CSS. Виявлення можливостей пристрою за допомогою запитів media. Виявлення старої версії Internet Explorer за допомогою умовних коментарів. Визначення стилів для друку.

## **Змістовий модуль 2. Основи створення динамічних веб-сайтів**

### **Тема 8. Основи мови програмування PHP**

Включення PHP в HTML. Синтаксис мови програмування PHP. Вирази. Оператори. Управління процесом виконання. Перетворення типів. Функції. Масиви. Обробка форм.

### **Тема 9. Об'єктно-орієнтовне програмування в PHP**

Термінологія. Об'ява класу. Створення об'єкту. Доступ до об'єктів. Конструктори та деструктори. Написання методів та властивостей. Наслідування

### **Тема 10. Розміщення веб-додатків в Інтернеті**

Віртуальні хостинги. Віртуальні виділені сервери. Хмарні сервіси.

## **Тема 11. Автономна робота**

Робота з Web SQL. Питання підтримки Web SQL. Створення та відкриття бази даних. Використання оновлення схеми. Використання транзакцій. Робота з HTTP офлайн кешем додатків. Підтримка браузерів. Файл маніфесту кешу. Оновлення кешу. Розуміння події.



## **ТЕМАТИКА КОНТРОЛЬНИХ РОБІТ.**

1. Мова гіпертекстової розмітки HTML
2. Каскадні таблиці стилів CSS
3. Створення HTML сторінок
4. Мова програмування JavaScript
5. Робота з об'єктами в JavaScript. Бібліотека jQuery
6. Робота з формами
7. Адаптивний інтерфейс користувача
8. Основи мови програмування PHP
9. Об'єктно-орієнтовне програмування в PHP
10. Розміщення веб-додатків в Інтернеті
11. Автономна робота

## ВКАЗІВКИ ДО ВИКОНАННЯ КОНТРОЛЬНИХ РОБІТ

Задачами контрольної роботи є:

- систематизація і закріплення теоретичних та практичних фахових знань, виявлення уміння студента застосовувати ці знання при вирішенні конкретних наукових, технічних, економічних і виробничих задач;
- перевірка уміння студента самостійно освоювати та використовувати сучасні інформаційні технології, програмно-апаратні засоби обчислювальної техніки;
- розвинення у студента навичок ведення самостійного науково-практичного пошуку, оволодіння методикою дослідження й експериментування при вирішенні проблем і питань, поставлених на курсове проектування;
- закріплення знань і навичок виконання графічних робіт та інших конструкторських документів у відповідності до вимог і правил, встановлених державними стандартами, Єдиною системою конструкторської документації (ЄСКД), Єдиною системою проектної документації (ЄСПД), іншими чинними нормативно-технічними документами;

Головна мета контрольної роботи – визначення рівня теоретичної та практичної підготовки студента з курсу „Мова сценаріїв JavaScript”.

Контрольна робота сприяє розвитку у студента творчої ініціативи і самостійності в проведенні аналізу, добору й обґрунтування найбільш раціональних інженерних рішень.

Контрольна робота надає студентіві таких навичок виконання виробничих завдань, які допоможуть йому швидко адаптуватися до умов праці у професійному колективі.

## ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ

1. Що таке елемент HTML та як він записується?
2. В чому полягає особливість порожніх елементів?
3. Що таке атрибут HTML та як він записується?
4. Як визначаються логічні атрибути?
5. Як записати власний атрибут?
6. Як створюються гіперпосилання?
7. Як задати ціль гіперпосилання?
8. Яким чином розмінюються зображення на веб-сторінці?
9. Що таке фрейм та як його додати в HTML документ?
10. Що таке каскадна таблиця стилів CSS?
11. Як додати внутрішній стиль CSS?
12. Як додати вбудовану таблицю стилів CSS?
13. Як додати посилання на зовнішню таблицю стилів CSS?
14. Як використовувати селектори класів для стилізації елементів?
15. Як використовувати селектори ідентифікаторів для стилізації елементів?
16. Які ще селектори можна використовувати?
17. Як виконується каскадування стилів CSS?
18. Як враховується специфічність стилів?
19. Як виконується спадкування стилів CSS?
20. Яким чином задається колір в CSS?
21. Як задати параметри шрифту в CSS?
22. Що таке блочна модель і які параметри вона має?
23. Як можна позиціонувати елемент `<div>`?
24. Що таке семантика HTML?
25. Який семантичний елемент HTML визначає заголовок?
26. Який семантичний елемент HTML визначає нижній колонтитул?
27. Який семантичний елемент HTML визначає блок навігації?

28. Який семантичний елемент HTML визначає бокову панель?
29. Який семантичний елемент HTML визначає розділ вмісту?
30. Який семантичний елемент HTML визначає частину цілого?
31. Як створюється нумерований список?
32. Як створюється маркований список?
33. Як створюється список визначень?
34. Яка базова структура таблиці в HTML?
35. Як об'єднувати комірки таблиці по горизонталі та вертикалі?
36. Як створюються програми на JavaScript?
37. Які типи даних існують в JavaScript?
38. Як виконуються інкремент/декремент змінної?
39. Як екранувати символи в рядках?
40. Які логічні оператори існують в JavaScript?
41. Назвіть основні правила іменування змінних в JavaScript.
42. Як оголошуються та використовуються функції?
43. Як оголошуються та використовуються функційні вирази?
44. Як вбудовані функції JavaScript ви знаєте?
45. Які визначаються області видимості змінних?
46. Як виконується перетворення типів?
47. Як реалізувати умовні оператори в JavaScript?
48. Як реалізувати цикли в JavaScript?
49. Як обробляються помилки в JavaScript?
50. Як додати JavaScript в HTML документ?
51. Як зневаджувати JavaScript програму?
52. Як працювати з масивами в JavaScript?
53. Що таке DOM?
54. Які методи JavaScript дозволяють знаходити бажаний елемент в DOM?
55. Як працювати з подіями DOM в JavaScript?
56. Як створюється об'єкт JavaScript в літеральній формі?

57. Як створюється об'єкт JavaScript за допомогою шаблону factory?
58. Як створити клас в JavaScript?
59. Як оголошуються приватні та публічні змінні в класах JavaScript?
60. Як реалізувати простір імен в JavaScript?
61. Як використовується правило CSS @media?
62. Які критерії запитів media можна використовувати?
63. Що «асинхронна операція»?
64. Як створюються проміси в jQuery?
65. Як використовувати проміси jQuery?
66. В чому полягає призначення протоколу WebSocket?
67. Які властивості та методи має об'єкт WebSocket?
68. Як реалізувати простий чат за допомогою WebSocket?
69. Як працювати з таймаутами в WebSocket?
70. Які бібліотеки для роботи WebSocket ви знаєте?
71. Як працює перетягування елементів на веб-сторінці?

Відповідність підсумкової семестрової рейтингової оцінки в балах оцінці за національною шкалою та шкалою ECTS

Оцінка в балах	Оцінка за національною шкалою	Оцінка за шкалою ECTS	
		Оцінка	Пояснення
<b>90-100</b>	<b><i>Відмінно</i></b>	<b>A</b>	<b>Відмінно</b> (відмінне виконання лише з незначною кількістю помилок)
<b>82 – 89</b>	<b>Добре</b>	<b>B</b>	<b>Дуже добре</b> (вище середнього рівня з кількома помилками)
<b>75 – 81</b>		<b>C</b>	<b>Добре</b> (в загальному вірне виконання з певною кількістю суттєвих помилок)
<b>67 – 74</b>	<b>Задовільно</b>	<b>D</b>	<b>Задовільно</b> (непогано, але зі значною кількістю недоліків)
<b>60 – 66</b>		<b>E</b>	<b>Достатньо</b> (виконання задовольняє мінімальним критеріям)
<b>35 – 59</b>	<b>Незадовільно</b>	<b>FX</b>	<b>Незадовільно</b> (з можливістю повторного складання)
<b>1 – 34</b>		<b>F</b>	Незадовільно (з обов'язковим повторним курсом)

Підсумкова семестрова рейтингова оцінка в балах, за національною шкалою та шкалою ECTS заноситься до заліково-екзаменаційної відомості, навчальної картки та залікової книжки студента.

Підсумкова семестрова рейтингова оцінка заноситься до залікової книжки та навчальної картки студента, наприклад, так: **92/Відм./A**, **87/Добре/B**, **79/Добре/C**, **68/Задов./D**, **65/Задов./E** тощо..

Підсумкова рейтингова оцінка з дисципліни визначається як середньоарифметична оцінка з підсумкових семестрових рейтингових оцінок у балах (з цієї дисципліни – за перший та другий семестри) з наступним її переведенням в оцінки за національною шкалою та шкалою ECTS.

## СПИСОК ЛІТЕРАТУРИ

### Основна

1. Гоше Х.Д. HTML5. Для профессионалов / Х.Д. Гоше. – [2-е изд.]. – СПб.: Питер, 2015. – 560 с.
2. Роббинс Д.Н. HTML5, CSS3 и JavaScript. Исчерпывающее руководство / Д.Н. Роббинс. – [4-е изд.]. – М.: Эксмо, 2014. – 516 с.
3. Флэнаган Д. JavaScript. Подробное руководство / Д. Флэнаган. – [6-е изд.]. – СПб.: Символ-Плюс, 2012. – 1080 с.
4. Фрейн Б. HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Б. Фрейн. – СПб.: Питер, 2014. – 304 с.

### Додаткова

5. Макфарланд Д. Большая книга CSS3 / Д. Макфарланд. – [3-е изд.]. – СПб.: Питер, 2014. – 608 с.
6. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. – [3-е изд.]. – СПб.: Питер, 2015. – 688 с.
7. Холмогоров В. Основы Web-мастерства: [учебный курс] / В. Холмогоров. – СПб.; М.; Х.: Питер, 2001. – 352 с.
8. Пьюривал С. Основы разработки веб-приложений / С. Пьюривал. – СПб.: Питер, 2015. – 272 с.

## **ЗМІСТ**

<u>ПОЯСНЮВАЛЬНА ЗАПИСКА</u> .....	3
<u>ТЕМАТИЧНИЙ ПЛАН</u> .....	4
<u>ЗМІСТ ДИСЦИПЛІНИ</u> .....	6
<u>ТЕМАТИКА КОНТРОЛЬНИХ РОБІТ</u> .....	9
<u>ВКАЗІВКИ ЩОДО ВИКОНАННЯ КОНТРОЛЬНИХ РОБІТ</u> .....	10
<u>ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ</u> .....	11
<u>СПИСОК ЛІТЕРАТУРИ</u> .....	15