

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ**  
**ІМЕНІ СЕМЕНА КУЗНЕЦЯ**

**Робоча програма**  
**навчальної дисципліни**  
**"ВЕБ-ТЕХНОЛОГІЇ ТА ВЕБ-ДИЗАЙН"**  
для студентів напряму підготовки  
6.050101 "Комп'ютерні науки"  
всіх форм навчання

Затверджено на засіданні кафедри інформаційних систем.  
Протокол № 6 від 09.12.2013 р.

**Укладачі:** Огурцов В. В.  
Гриньов Д. В.  
Щербаков О. В.

P58 Робоча програма навчальної дисципліни "Веб-технології та веб-дизайн" для студентів напряму підготовки 6.050101 "Комп'ютерні науки" всіх форм навчання / укл. В. В. Огурцов, Д. В. Гриньов, О. В. Щербаков. – Х. : Вид. ХНЕУ ім. С. Кузнеця, 2014. – 54 с. (Укр. мов.)

Подано тематичний план навчальної дисципліни та її зміст за модулями й темами. Вміщено плани лекцій, лабораторних занять, матеріали для закріплення знань (самостійну роботу, контрольні запитання), критерії оцінювання знань студентів.

Рекомендовано для студентів напряму підготовки 6.050101 "Комп'ютерні науки".

# Вступ

Нова інформаційна технологія досягла такого розвитку, що, мабуть, не залишилося сфер людського життя, які незачеплені глобальною мережею Internet. У цей час інтерес до мережі Internet продовжує зростати. Розроблений у роки інформаційного вибуху Internet стає невід'ємною частиною життя більшості людей усього світу.

Якісний сайт стає важливим, а у деяких галузях – єдиним, засобом досягнення економічних, політичних, соціальних, рекламних та інших цілей. Якісний сайт відрізняє від інших сайтів в глобальній мережі такі риси: висока якість інформаційного наповнення й грамотність його подачі; оригінальність і естетична привабливість зовнішнього вигляду сторінок; доступність змісту сайту для максимально широкого кола користувачів поза залежністю від застосовуваних ними типів пристроїв і версій браузерів, а також від обмежень по стану здоров'я; ергономічність елементів користувацького інтерфейсу сайту, що забезпечує високу ефективність, але в той же час легкість і невимушеність взаємодії відвідувача з веб-ресурсом; надійність і безпека використовуваних технологічних рішень, чітка погодженість роботи всіх компонентів; бездоганне пророблення всіх деталей і нюансів.

Але для створення якісного сайту, тобто відповідності цим рисам, потрібна плідна робота висококваліфікованих спеціалістів із різних веб-технологій, які повинні розуміти не лише вузьку галузь знань, але і добре представляти весь спектр веб-технологій.

Необхідним елементом успішного засвоєння навчального матеріалу дисципліни є самостійна робота студентів з технічною літературою, та сучасним програмними засобами розробки програм.

Навчальна дисципліна "Веб-технології та веб-дизайн" є нормативною навчальною дисципліною та вивчається згідно з навчальним планом підготовки фахівців освітньо-кваліфікаційного рівня "бакалавр" напряму підготовки 6.050101 "Комп'ютерні науки" для всіх форм навчання.

## Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 9	Галузь знань 0501 "Інформатика та обчислювальна техніка"	Нормативна	
Змістових модулів – 6	Напрямок підготовки 6.050101 "Комп'ютерні науки"	Рік підготовки	
Індивідуальне науково-дослідне завдання "Серверні веб-застосування"		3, 4-й	3, 4-й
Загальна кількість годин – 324		Семестр	
		5, 6, 7-й	6, 7-й
Тижневих годин для денної форми навчання за семестрами: аудиторних – 4, 6, 4; самостійної роботи студента – 4, 4, 4	Освітньо-кваліфікаційний рівень: бакалавр	Лекції	
		48 год	16 год
		Практичні, семінарські	
			16 год
		Лабораторні	
		114 год	
		Самостійна робота	
		144 год	130 год
		Індивідуальні завдання:	
		18 год	
Вид контролю:			
іспит	залік		

*Примітка.* Співвідношення кількості годин аудиторних занять до самостійної та індивідуальної роботи становить:

для денної форми навчання – 45 %;

для заочної форми навчання – 80 %.

## 2. Мета та завдання навчальної дисципліни

**Метою** навчальної дисципліни є засвоєння необхідних знань з основ веб-технологій та веб-дизайну, а також формування твердих практичних навичок щодо розробки якісних сайтів.

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

отримання теоретичних знань з основ веб-технологій, веб-дизайну та веб-програмування;

отримання практичних навичок з розробки веб-сайтів.

**"Веб-технології та веб-дизайн"** – навчальна дисципліна, що вивчає інструментарій розробки веб-сайтів з використанням веб-програмування та веб-дизайну.

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

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

З метою найкращого засвоєння матеріалу студенти повинні до початку вивчення дисципліни засвоїти теоретичні знання та опанувати практичні вміння з дисциплін: "Алгоритмізація та програмування", "Об'єктно-орієнтоване програмування", "Організація баз даних і знань", а також мати навички роботи з персональним комп'ютером.

У процесі навчання студенти отримують необхідні знання під час лекційних занять та виконання лабораторних завдань. Також велике значення в процесі вивчення та закріплення знань має самостійна робота студентів. Усі види занять розроблені відповідно до кредитно-модульної системи організації навчального процесу.

У результаті вивчення навчальної дисципліни 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,

правила побудови і основні елементи серверних сценаріїв,

правила взаємодії серверних сценаріїв з серверами БД та веб-сервісами;

правила побудови і основні елементи архітектури "модель – відображення – контролер";

**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,

написати нескладний сценарій обробки події,

створити серверний сценарій та забезпечити його взаємодію з БД та веб-сервісами;

створити веб-застосування ASP.NET MVC;

**буди здатним (компетентності)** до вирішення складних спеціалізованих задач та практичних проблем у певній галузі професійної діяльності або навчання, що пов'язані із застосуванням певних теорій та методів відповідних наук і характеризуються комплексністю та недетермінованістю умов.

У процесі викладання навчальної дисципліни основна увага приділяється оволодінню студентами професійними компетентностями згідно з Національною рамкою кваліфікацій України, що наведені в табл. 2.1.

Таблиця 2.1

**Професійні компетентності, які отримують студенти після вивчення навчальної дисципліни**

Знання	Вміння	Комунікація	Автономність і відповідальність
<p>Концептуальні знання в галузі навчання та професійної діяльності, включаючи певні знання сучасних досягнень.</p> <p>Критичне осмислення основних теорій, принципів, методів і понять в галузі навчання та професійної діяльності</p>	<p>Вирішення складних непередбачуваних задач і проблем у спеціалізованих сферах професійної діяльності та/або навчання, яке передбачає збирання та інтерпретацію інформації (даних), вибір методів та інструментальних засобів, застосування інноваційних підходів</p>	<p>Донесення до фахівців і нефахівців інформації, ідей, проблем, рішень та власного досвіду в галузі професійної діяльності.</p> <p>Здатність ефективно формувати комунікаційну стратегію</p>	<p>Управління комплексними діями або проектами, відповідальність за прийняття рішень у непередбачуваних умовах.</p> <p>Відповідальність за професійний розвиток окремих осіб та/або груп осіб.</p> <p>Здатність до подальшого</p>

### **3. Програма навчальної дисципліни**

#### **Змістовий модуль 1.**

#### **Основи веб та веб-дизайн**

##### **Тема 1. Структура і принципи Веб. Уведення в HTML**

Архітектура та принципи організації всесвітнього павутиння. Поняття гіпертексту. Зв'язок між гіпертекстом, Інтернетом та всесвітнім павутинням. Клієнт-серверна технологія та її роль у всесвітньому павутинні.

Розвиток та сучасний стан мови HTML. Структура документа HTML. Мінімальний документ. Теги для форматування документа. Посилання як основа гіпертексту. Списки. Використання графіки. Графічні формати. Таблиці в документі HTML. Форми в документі HTML.

##### **Тема 2. Технологія CSS та її підтримка браузерами**

Каскадні таблиці стилів (CSS). Ідеологія і способи використання. Властивості форматування елементів засобами CSS: шрифт, колір і фон, форматування блоків, форматування списків.

##### **Тема 3. Блокова верстка сторінок веб-сайта**

Блокова модель елементів веб-сторінок. Блокові, строкові та інші види html-елементів. Зміна виду html-елементів за допомогою CSS. Види позиціонування елементів веб-сторінки. Верстка веб-сторінок з фіксованою шириною, резинова верстка та еластична верстка.

#### **Змістовий модуль 2.**

#### **Веб-програмування на боці клієнта**

##### **Тема 4. Основи javascript**

Загальний огляд мови сценаріїв javascript. Синтаксис та особливості мови. Оператори javascript. Розгалуження і цикли. Оператори роботи з об'єктами у javascript. Особливості роботи з масивами. Функції у javascript. Створення об'єктів користувача у javascript. Організація передачі інформації між сторінками сайта засобами мови javascript.

##### **Тема 5. Програмна взаємодія з HTML документами на основі**

## **DOM API**

Об'єктна модель документа (DOM). Види вузлів DOM. Способи доступу до вузлів DOM за допомогою javascript. Маніпуляція вузлами DOM за допомогою javascript. Поняття події. Процедури обробки події.

### **Тема 6. Використання бібліотек javascript для розробки веб-сайтів. Бібліотека jQuery**

Огляд популярних javascript-бібліотек. Порівняльний аналіз популярних javascript-бібліотек. Концепція бібліотеки jQuery. Доступ (вибір) до вузлів DOM засобами бібліотеки jQuery. Маніпуляція вузлами DOM за допомогою бібліотеки jQuery.

## **Змістовий модуль 3.**

### **Веб-програмування на боці сервера. Основи php**

#### **Тема 7. Взаємодія веб-клієнта з веб-сервером. Протокол HTTP та засоби моніторингу http-трафіка. Мови розроблення серверних сценаріїв**

Протокол HTTP. Загальна структура http-повідомлень. Методи протоколу HTTP. Заголовки http-повідомлень. Кодування даних http-повідомлень. Засоби моніторингу http-трафіка. Поняття серверного сценарію. Стандарти взаємодії серверних сценаріїв з веб-сервером. Огляд мов розроблення серверних сценаріїв.

#### **Тема 8. Розробка веб-застосунків за допомогою мови PHP**

Методи вбудовування PHP-коду. Вивід результатів роботи скрипта. Змінні у PHP. Типи даних та ініціалізація змінних. Перевірка існування змінних. Видалення змінних. Створення та використання констант. Оператори PHP: математичні, присвоювання, двійкові, конкатенації строк, виклику зовнішніх програм. Перетворення типів даних. Масиви (одномірні, багатомірні, асоціативні) та робота з ними. Функції для роботи зі строками, числами, датою та часом. Функції користувача, глобальні та локальні змінні. Умовні оператори. Оператори циклів. Завершення виконання сценарію. Помилки у програмі.

#### **Тема 9. Обробка даних http-запитів мовою PHP. Механізми управління станом веб-застосунків**

Змінні оточення. Глобальні масиви. Обробка даних форми. Передача файлів в http-повідомленнях та їх обробка мовою PHP.



Встановлення http-заголовків за допомогою PHP. Cookie, їх обробка та встановлення. Механізм управління сесансом користувача. Способи ідентифікації сесії (сесансу). Ініціювання сесії. Додавання та видалення даних (змінних) в сесії.

## **Змістовий модуль 4.**

### **Веб-програмування на боці сервера.**

#### **Механізми роботи з об'єктами і даними в веб**

##### **Тема 10. Особливості ООП в PHP**

Створення класів. Визначення властивостей класів. Робота із методами. Визначення метода-конструктора. Аргументи та типи. Управління доступом до класів та елементів класів. Статичні методи та властивості. Постійні властивості. Абстрактні класи та методи. Інтерфейси. Виключення. Закінчені класи та методи. Клонування об'єктів. Простори імен. Автозавантаження. Reflection API.

##### **Тема 11. Інтерфейси взаємодії PHP-застосувань з СУБД**

Огляд засобів PHP для роботи з текстовими, DBM та реляційними базами даних. PDO. З'єднання із БД SQL. Виконання запитів до БД SQL. Вибір, додавання, оновлення та видалення даних з таблиць БД. Визначення кількості рядків у наборі даних. Типи даних полів таблиць у MySQL. Створення бази даних в MySQL. Створення облікових записів користувачів бази даних. Створення таблиць. Видалення таблиці і бази даних.

##### **Тема 12. DOM XML. Програмна обробка XML-документів з допомогою XML DOM та Simple XML**

Структура XML-документів. Приклади ефективного використання XML. Огляд PHP API для робіт з XML. Створення XML-документа за допомогою PHP DOM XML та Simple XML. Доступ до xml-елементів за допомогою PHP DOM XML та Simple XML. Додавання, зміна, видалення елементів та їх атрибутів в XML-документах за допомогою PHP DOM XML та Simple XML. Порівняння PHP DOM XML та Simple XML. XPath та PHP.

## **Змістовий модуль 5.**

### **Веб-програмування на боці сервера.**

#### **Основи ASP.NET**

### **Тема 13. Розміщення веб-сайта на сервері. Створення інтерактивних елементів**

Загальне уявлення про сервер Internet Information Service (IIS). Базові поняття про налаштування IIS. Використання IIS при розробці веб-сайта.

Основні компоненти веб-застосування ASP.NET. Основні елементи веб-форми. Серверні елементи управління. Збереження стану відображення елементів управління форми.

Методи реалізації коду: код програмної частини, змішаний код, вбудований код. Обробка подій на сервері та клієнті. Типи, компоненти і класи. Життєвий цикл веб-сторінки. Етапи та події життєвого циклу веб-сторінки. Процес зворотної передачі даних. Автоматична зворотня відправка даних.

Використання майстра веб-сторінки та елементів управління користувача. Елементи управління перевірки правильності вводу даних.

Управління станом в ASP.NET на боці клієнта та на боці сервера. Стан відображення, приховані поля, Cookies, строки запиту, стан сеансу, стан додатку, кешування.

Налагодження та трасування веб-сайта, розгорнення веб-сайта. Модель безпеки ASP.NET. Рівні безпеки, авторизація та автентифікація.

### **Тема 14. Інтерфейси взаємодії ASP.NET-застосувань з СУБД**

Основи технології доступу до даних ADO.NET. Об'єктна модель. Постачальники даних ADO.NET. Класи, які основані на з'єднанні та класи, які основані на змістовності. Компоненти даних. Прив'язка даних. З'єднання з сховищем даних та SQL-запити. Багатофункціональні елементи управління даними. Об'єктно-реляційне відображення.

Мова інтегрованих запитів LINQ. Типи реалізації LINQ. Стандарти операторів запитів. Запитоподібний та процедурний формат написання LINQ. Відкладене та негайне виконання запитів.

## **Змістовий модуль 6.**

### **Розробка веб-порталів. CMF ASP.NET MVC**

**Тема 15. Уведення в технологію AJAX. Реалізація асинхронної взаємодії браузера з веб-сервером з допомогою технології AJAX**

Загальні принципи технології AJAX. Особливості реалізації AJAX в різних браузерах. Властивості та методи класу XMLHttpRequest. ASP.NET AJAX на боці клієнта та на боці сервера. Серверні елементи управління. Часткова візуалізація та синхронізація по часу оновлення. Клієнтські бібліотеки. Модель клієнта. Об'єктно-орієнтоване програмування в *javascript*. Інструментарій ASP.NET AJAX Control Toolkit.

#### **Тема 16. Веб-сервіси та мови їх описування**

Роль веб-сервісів XML. Компоненти веб-сервісів. Мова опису веб-сервісів WSDL. Формат повідомлень. Протоколи зв'язку веб-сервісів. Введення в WCF. Роль веб-служб. Поняття контрактів WCF, прив'язок та адрес. Основи побудови служби WCF.

#### **Тема 17. Основи розробки веб-порталів за допомогою CMF ASP.NET MVC**

Архітектура "модель – відображення – контролер" в ASP.NET MVC. Контролери та дії. Отримання даних, які введені. Генерація виводу. Не типізовані та типізовані відображення. Використання часткового відображення. Моделі даних та моделі відображення. Синтаксис Razor. Допоміжні методи AJAX. Використання бібліотеки jQuery в ASP.NET MVC.

### **4. Структура навчальної дисципліни**

Із самого початку вивчення навчальної дисципліни кожен студент має бути ознайомлений як з робочою програмою навчальної дисципліни і формами організації навчання, так і зі структурою, змістом та обсягом кожного з її навчальних модулів, а також з усіма видами контролю та методикою оцінювання сформованих професійних компетентностей.

Вивчення студентом навчальної дисципліни відбувається шляхом послідовного і ґрунтовного опрацювання змістовних модулів. Змістовний модуль – це окремий, відносно самостійний блок дисципліни, який логічне об'єднує кілька навчальних елементів дисципліни за змістом та взаємозв'язками. Тематичний план дисципліни складається з шести змістових модулів (табл. 4.1).

Таблиця 4.1

#### **Структура залікового кредиту навчальної дисципліни**

Назви змістових модулів і тем	Кількість годин	
	денна форма	заочна форма

1	усього о	у тому числі					усього о	у тому числі				
		л	п	лаб.	інд.	с. р.		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Змістовий модуль 1. ОСНОВИ ВЕБ ТА ВЕБ-ДИЗАЙН</b>												
Тема 1. Структура і принципи Веб. Уведення в HTML	16	2		4		10	20	2	2			16

Продовження табл. 4.1

1	2	3	4	5	6	7	8	9	10	11	12	13
Тема 2. Технологія CSS та її підтримка браузерами	18	4		6		8	20	2	2			16
Тема 3. Блокова верстка сторінок веб-сайта	14	2		6		6	20	2	2			16
<i>Разом за змістовим модулем 1</i>	<i>48</i>	<i>8</i>		<i>16</i>		<i>24</i>	<i>60</i>	<i>6</i>	<i>6</i>			<i>48</i>
<b>Змістовий модуль 2. ВЕБ-ПРОГРАМУВАННЯ НА БОЦІ КЛІЄНТА</b>												
Тема 4. Основи JavaScript	14	2		4		8	20	2	2			16
Тема 5. Програмна взаємодія з HTML документами на основі DOM API	20	4		6		10	20	2	2			16
Тема 6. Використання бібліотек JavaScript для розробки веб-сайтів. Бібліотека jQuery	18	2		6		10						
<i>Разом за змістовим модулем 2</i>	<i>52</i>	<i>8</i>		<i>16</i>		<i>28</i>		<i>4</i>	<i>4</i>			<i>32</i>
<b>Змістовий модуль 3. ВЕБ-ПРОГРАМУВАННЯ НА БОЦІ СЕРВЕРА. ОСНОВИ PHP</b>												
Тема 7. Взаємодія веб-клієнта з веб-сервером. Протокол	20	2		6		10	20	2	2			16

HTTP та засоби моніторингу http-трафіка. Мови розроблення серверних сценаріїв												
Тема 8. Розробка веб-застосувань за допомогою мови PHP	20	4		6		10	20	2	2			16

Продовження табл. 4.1

1	2	3	4	5	6	7	8	9	10	11	12	13
Тема 9. Обробка даних http-запитів. Механізми управління станом веб-застосувань	28	2		14	6	8	22	2	2			18
<i>Разом за змістовим модулем 3</i>	68	8		26	6	28	62	6	6			50
<b>Змістовий модуль 4.</b> ВЕБ-ПРОГРАМУВАННЯ НА БОЦІ СЕРВЕРА. МЕХАНІЗМИ РОБОТИ З ОБ'ЄКТАМИ І ДАНИМИ В ВЕБ												
Тема 10. Особливості ООП в PHP	28	4		6	6	12						
Тема 11. Інтерфейси взаємодії PHP-застосувань з СУБД	26	2		10	6	8						
Тема 12. DOM XML. Програмна обробка XML-документів з допомогою XML DOM та Simple XML	20	2		8		10						
<i>Разом за змістовим модулем 4</i>	74	8		24	12	30						
<b>Змістовий модуль 5.</b> ВЕБ-ПРОГРАМУВАННЯ НА БОЦІ СЕРВЕРА. ОСНОВИ ASP.NET												
Тема 13. Розміщення веб-сайта на сервері. Створення інтерактивних елементів	20	4		8		8						
Тема 14. Інтерфейси взаємодії ASP.NET-застосувань з СУБД	20	4		8		8						
<i>Разом за змістовим модулем 5</i>	40	8		16		16	40	4	4			32

**Змістовий модуль 6.**  
**ВЕБ-ПРОГРАМУВАННЯ НА БОЦІ СЕРВЕРА.**  
**МЕХАНІЗМИ РОБОТИ З ОБ'ЄКТА-МИ І ДАНИМИ В ВЕБ**

Тема 15. Уведення в технологію AJAX. Реалізація асинхронної взаємодії браузера з веб-сервером з допомогою технології AJAX	10	2	2	6								
---	----	---	---	---	--	--	--	--	--	--	--	--

Закінчення табл. 4.1

1	2	3	4	5	6	7	8	9	10	11	12	13
Тема 16. Веб-сервіси та мови їх описування	10	2		2		4						
Тема 17. Основи розробки веб-порталів за допомогою C# ASP.NET MVC	24	4		12		8						
<i>Разом за змістовим модулем 6</i>	42	8		16		18						
<b>Усього годин</b>	<b>324</b>	<b>48</b>		<b>114</b>		<b>144</b>	<b>162</b>	<b>16</b>	<b>16</b>			<b>130</b>

## 5. Теми лабораторних занять

Лабораторне заняття – це організаційна форма навчального заняття, на якому студенти під керівництвом викладача формують уміння й навички з практичного застосування основних теоретичних положень навчальної дисципліни шляхом виконання завдань до лабораторних робіт.

Лабораторні заняття з дисципліни "Веб-технології та веб-дизайн" проводяться в спеціально обладнаному навчальному класі з використанням комп'ютерного устаткування пристосованого до навчального процесу.

З метою підвищення якості навчального процесу, під час проведення лабораторного заняття призначається ще один викладач і навчальна група ділиться на дві підгрупи. Кожний студент працює самостійно, виконуючи індивідуальне завдання для лабораторного дослідження.

За результатами виконаної на занятті лабораторної роботи студенти оформлюють індивідуальні звіти з її виконання та захищають ці звіти перед викладачем. Результати виконання лабораторних досліджень оцінюються викладачем.

Тематика проведення лабораторних занять наведена у табл. 5.1.

Таблиця 5.1

### План проведення лабораторних занять

Назва теми	Назва лабораторного заняття та питання що опрацьовуються	Кількість годин	Література
1	2	3	4
<b>Змістовний модуль 1. Основи веб та веб-дизайн</b>			
Тема 1. Структура і принципи Веб. Уведення в HTML	Розробка веб-сторінок з використанням мови HTML	4	Основна: [1; 8; 9]. Додаткова: [18; 20 – 22; 24]
Тема 2. Технологія CSS та її підтримка браузерами	Розробка веб-сайта з використанням CSS та табличної верстки	6	Основна: [2 – 7; 9 – 12]. Додаткова: [18; 23]
Тема 3. Блокова верстка сторінок веб-сайта	Розробка веб-сайта з використанням блокової верстки	6	Основна: [2 – 7; 9 – 12]. Додаткова: [19 – 21; 24]
<b>Змістовний модуль 2. Веб-програмування на боці клієнта</b>			
Тема 4. Основи Javascript	Розробка динамічних веб-сторінок за допомогою мови Javascript	4	Основна: [6; 9]. Додаткова: [19 – 21; 24]
Тема 5. Програмна взаємодія з HTML документами на основі DOM API	Розробка динамічних веб-сторінок за допомогою мови Javascript та DOM API	6	Основна: [9; 13]. Додаткова: [27]

Тема 6. Використання бібліотек JavaScript для розробки веб-сайтів. Бібліотека jQuery	Розробка динамічних веб-сторінок за допомогою Javascript-бібліотеки jQuery	8	Основна: [1; 5; 9; 13; 15]. Додаткова: [22; 26]
<b>Змістовний модуль 3. Веб-програмування на боці сервера. Основи PHP</b>			
Тема 7. Взаємодія веб-клієнта з веб-сервером. Протокол HTTP та засоби моніторингу http-трафіка. Мови розроблення серверних сценаріїв	Моніторинг та аналіз http-трафіка	4	Основна: [1; 2; 4; 6; 12; 16]. Додаткова: [23; 25]

Продовження табл. 5.1

1	2	3	4
Тема 8. Розробка веб-застосувань за допомогою мови PHP	Розробка веб-застосувань за допомогою мови PHP	8	Основна: [9; 5]. Додаткова: [18]
Тема 9. Обробка даних http-запитів. Механізми управління станом веб-застосувань	Розробка веб-застосувань мовою PHP з використанням механізмів управління станом	8	Основна: [5; 9]. Додаткова: [18]
<b>Змістовний модуль 4. Веб-програмування на боці сервера. Механізми роботи з об'єктами і даними в Веб</b>			
Тема 10. Особливості ООП в PHP	Розробка веб-застосувань з використанням ООП в PHP	6	Основна: [9; 13; 15]. Додаткова: [22]
Тема 11. Інтерфейси взаємодії PHP-застосувань з СУБД	Розробка веб-застосувань з використанням серверу БД MySQL з використанням процедурного підходу та об'єктно-орієнтованого (модуль PDO)	8	Основна: [9; 13]. Додаткова: [27]
Тема 12. DOM XML. Програмна обробка XML-документів з допомогою XML DOM	Розробка веб-застосувань мовою PHP з використанням API (XML DOM та Simple XML) для обробки XML-документів	8	Основна: [1; 5; 9; 13; 15]. Додаткова: [22; 26]



та Simple XML			
<b>Змістовний модуль 5.</b> <b>Веб-програмування на боці сервера. Основи ASP.NET</b>			
Тема 13. Розміщення веб-сайта на сервері. Створення інтерактивних елементів	Створення веб-форми з використанням серверних елементів управління. Реалізація обробки подій на сервері та клієнті	4	Основна: [1; 5; 7; 14; 17]. Додаткова: [23; 25; 28]
	Використання майстра веб-сторінки та елементів управління користувача. Управління станом в ASP.NET. Реалізація авторизації та автентифікації	4	Основна: [2; 7; 14; 17]. Додаткова: [27 – 28]

Закінчення табл. 5.1

1	2	3	4
Тема 14. Інтерфейси взаємодії ASP.NET-застосувань з СУБД	Створення бази даних та використання технології доступу до даних ADO.NET	4	Основна: [7; 14; 17]. Додаткова: [28 – 29]
	Використання об'єктно-реляційного відображення для взаємодії з базою даних. Застосування LINQ	4	Основна: [7; 17]. Додаткова: [29]
<b>Змістовний модуль 6.</b> <b>Розробка веб-порталів. CMF ASP.NET MVC</b>			
Тема 15. Уведення в технологію AJAX	Використання технології AJAX	2	Основна: [7; 14]. Додаткова: [26]
Тема 16. Веб-сервіси та мови їх описування	Розробка веб-служби в ASP.NET та використання веб-сервісів Інтернет	2	Основна: [7; 14]. Додаткова: [30]
Тема 17. Основи розробки веб-порталів за допомогою CMF ASP.NET MVC	Створення веб – додатку з використанням патерну "модель – відображення – контролер" в ASP.NET MVC	6	Основна: [7; 10]. Додаткова: [31]
	Реалізація доступу до бази даних веб-дodatка ASP.NET MVC	6	Основна: [7; 10] Додаткова: [31]
Разом годин		114	

## 6. Самостійна робота

**Самостійна робота студента (СРС)** – це форма організації навчального процесу, за якої заплановані завдання виконуються студентом самостійно під методичним керівництвом викладача.

**Мета СРС** – засвоєння в повному обсязі навчальної програми та формування у студентів загальних і професійних компетентностей, які відіграють суттєву роль у становленні майбутнього фахівця вищого рівня кваліфікації.

Навчальний час, відведений для самостійної роботи студентів денної форми навчання, визначається навчальним планом і становить 45 % (144 годин) від загального обсягу навчального часу на вивчення дисципліни (324 годин). У ході самостійної роботи студент має перетворитися на активного учасника навчального процесу, навчитися свідомо ставитися до оволодіння теоретичними і практичними знаннями, вільно орієнтуватися в інформаційному просторі, нести індивідуальну відповідальність за якість власної професійної підготовки. СРС включає: опрацювання лекційного матеріалу; опрацювання та вивчення рекомендованої літератури, основних термінів та понять за темами дисципліни; підготовку до лабораторних занять; поглиблене опрацювання окремих лекційних тем або питань; виконання індивідуальних завдань (вирішення розрахункових індивідуальних та комплексних завдань) за вивченою темою; пошук (підбір) та огляд літературних джерел за заданою проблематикою дисципліни; аналітичний розгляд наукової публікації; контрольну перевірку студентами особистих знань за запитаннями для самодіагностики; підготовку до контрольних робіт та інших форм поточного контролю; підготовку до модульного контролю; систематизацію вивченого матеріалу з метою підготовки до семестрових заліків та екзамену.

Необхідним елементом успішного засвоєння матеріалу навчальної дисципліни є самостійна робота студентів з вітчизняною та закордонною спеціальною технічною літературою в області веб-програмування та веб-дизайна. Основні види самостійної роботи, які запропоновані студентам для засвоєння теоретичних знань з навчальної дисципліни, наведені в табл. 6.1.

## Завдання для самостійної роботи студентів та форми її контролю

Назва теми	Зміст самостійної роботи студентів	Кількість годин	Форми контролю СРС	Література
1	2	3	4	5
<b>Змістовий модуль 1. Основи веб та веб-дизайн</b>				
<i>Тема 1.</i> Структура і принципи Веб. Мова HTML	Універсальний ідентифікатор – URI. Схеми адресації ресурсів в Internet. Принципи роботи HTTP. Особливості HTML 5. Використання шарів в HTML. Підходи щодо оптимізації HTML-сторінок	10	Захист індивідуального завдання	Основна: [1; 3 – 5; 8 – 9]. Додаткова: [18; 20 – 22; 24]

Продовження табл. 6.1

1	2	3	4	5
<i>Тема 2.</i> Технологія CSS та її підтримка браузерами	Створення меню, що випадає. CSS 3: елементи із згладженими кутами; лінійні і сферичні градієнти; додавання до елементів і до тексту елементів тіні; небезпечні шрифти; анімація і різні ефекти переходів	8	Захист індивідуального завдання	Основна: [1; 3 – 5; 8 – 9]. Додаткова: [18; 23]
<i>Тема 3.</i> Блокова верстка сторінок веб-сайта	Колапсуючу відступи, потік документу, накладення і порядок шарів, плаваючі елементи, позиціонування елементів	6	Захист індивідуального завдання	Основна: [3 – 6; 11 – 12]. Додаткова: [20 – 22]
<b>Усього за змістовим модулем 1</b>		<b>24</b>		
<b>Змістовий модуль 2. Веб-програмування на боці клієнта</b>				
<i>Тема 4.</i> Основи JavaScript	Динамічні форми. Читання та запис строк у файли cookie. Перетворення символів строки в Unicode та у зворотному напрямку. Кодування та декодування URL. Обробка помилок сценарію. Кодування та	8	Захист індивідуального завдання	Основна: [6; 9]. Додаткова: [19; 24]

	декодування строк у форматі Base64			
<i>Тема 5.</i> Програмна взаємодія з HTML документами на основі DOM API	Прив'язування звуків до подій. Запобігання попаданню сторінки до історії браузера. Створення контекстного меню. Меню на основі XML	10	Захист індивідуального завдання	Основна: [2; 6; 9]. Додаткова: [19; 24]
<i>Тема 6.</i> Використання бібліотек JavaScript для розробки веб-сайтів. Бібліотека jQuery	Ядро jQuery, селектори, атрибути. Traversing. Маніпуляції з елементами DOM. Таблиці стилів CSS. Події, ефекти. Взаємодія з Ajax. Інструменти внутрішні функції.	10	Захист індивідуального завдання	Основна: [2; 6]. Додаткова: [18; 24]
<b>Усього за змістовим модулем 2</b>		<b>28</b>		

Продовження табл. 6.1

1	2	3	4	5
<b>Змістовий модуль 3.</b>				
<b>Веб-програмування на боці сервера. Основи PHP</b>				
<i>Тема 7.</i> Взаємодія веб-клієнта з веб-сервером. Протокол HTTP та засоби моніторингу http-трафіка. Мови розроблення серверних сценаріїв	Відправка електронної пошти із сайту. Автентифікація за допомогою PHP. Заголовки HTTP та робота з ними. Робота з файлами та каталогами.	10	Захист індивідуального завдання	Основна: [9; 13; 15]. Додаткова: [23; 24]

<i>Тема 8.</i> Розробка веб-застосувань за допомогою мови PHP	Робота із графікою. Способи перевірки відповідності формату файлу, який завантажує користувач на сервер за допомогою форми (на боці клієнту) та PHP (на боці серверу)	10	Захист індивідуального завдання	Основна: [9; 13; 15]. Додаткова: [23; 24]
<i>Тема 9.</i> Обробка даних http-запитів. Механізми управління станом веб-застосувань	Непрямі посилання на змінні, XML і веб-сервіси, додаткові можливості PHP	8	Захист індивідуального завдання	Основна: [9; 13; 15]. Додаткова: [23; 24]
<b>Усього за змістовим модулем 3</b>		<b>28</b>		
<b>Змістовий модуль 4.</b> <b>Веб-програмування на боці сервера.</b> <b>Механізми роботи з об'єктами і даними в веб</b>				
<i>Тема 10.</i> Особливості ООП в PHP	Можливості перевантаження, перехоплення звернень до властивостей і методів, віртуальні масиви, ітератори. Шаблони проектування	12	Захист індивідуального завдання	Основна: [9; 13; 15]. Додаткова: [23; 24]

Продовження табл. 6.1

1	2	3	4	5
<i>Тема 11.</i> Інтерфейси взаємодії PHP-застосувань з СУБД	Параметри настроювання сервера MySQL. Видалення таблиць за маскою. Багатотабличні запити. Організація повнотекстового пошуку в MySQL. Проблеми з використанням різних кодових сторінок в MySQL та на Web-сторінках. Розмежування прав доступу на сайті за допомогою MySQL та PHP. Оптимізація використання БД MySQL у PHP. Підключення до	8	Захист індивідуального завдання	Основна: [9; 12; 13; 15]. Додаткова: [23; 24]

	MySQL іншого домену. Організація пошуку в БД MySQL за допомогою PHP			
<i>Тема 12.</i> DOM XML. Програмна обробка XML-документів з допомогою XML DOM та Simple XML	Переклад даних XML-файла в об'єкти і класи PHP. Обхід дерева об'єктів XML	10	Захист індивідуального завдання	Основна: [9; 13; 15]. Додаткова: [23; 28]
<b>Усього за змістовим модулем 4</b>		<b>30</b>		
<b>Змістовий модуль 5.</b> <b>Веб-програмування на боці сервера. Основи ASP.NET</b>				
<i>Тема 13.</i> Розміщення веб-сайта на сервері. Створення інтерактивних елементів	Налаштування IIS. Особливості, які відрізняють веб-сайт та веб-додаток. Відкладена обробка подій на сервері. Динамічне створення серверних елементів управління. Домен додатку ASP.NET. Час життя додатку. Архітектура сеансу. Конфігурація ASP.NET. Обробники HTTP. Модулі HTTP. Обробка помилок додатку, спеціальні сторінки помилок.	8	Захист індивідуального завдання	Основна: [7; 14 ;17]. Додаткова: [26; 29; 30]

Закінчення табл. 6.1

1	2	3	4	5
<i>Тема 14.</i> Інтерфейси взаємодії ASP.NET-застосувань з СУБД	Стандартизація в ADO.NET. Тестування з'єднання з базою даних. Транзакції в ADO.NET. Створення фабрики. Інтеграція з XML	8	Захист індивідуального завдання	Основна: [7; 14 ;17]. Додаткова: [26; 29; 30]

<b>Усього за змістовим модулем 5</b>		<b>16</b>		
<b>Змістовий модуль 6. Розробка веб-порталів. CMF ASP.NET MVC</b>				
<i>Тема 15.</i> Уведення в технологію AJAX. Реалізація асинхронної взаємодії браузера з веб-сервером з допомогою технології AJAX	Спеціальні елементи управління з JavaScript-кодом. Атаки впровадженням сценаріїв JavaScript. Перевірка достовірності запиту. Фрейми. Клієнтські зворотні виклики в спеціальних елементах управління	6	Захист індивідуального завдання	Основна: [7; 14 ;17]. Додаткова: [26; 29; 30]
<i>Тема 16.</i> Веб-сервіси та мови їх описування	Побудова служби WCF. Хостинг служби WCF. Конфігураційні настройки в WCF. Тестування служби WCF. Проектування контрактів даних WCF	4	Захист індивідуального завдання	Основна: [7; 14; 17]. Додаткова: [26; 29]
<i>Тема 17.</i> Основи розробки веб-порталів за допомогою CMF ASP.NET MVC	URL і механізм маршрутизації. Формат JSON. Використання фільтрів для підключення повторно використаної поведінки. Контролери як частина конвеєру обробки запитів. Допоміжні методи Ajax в ASP.NET MVC. Використання JQuery в ASP.NET MVC. Використання технології WebForms в додатку MVC. Безпечне використання MVC Framework	8	Захист індивідуального завдання	Основна: [10; 14]. Додаткова: [31]
<b>Усього за змістовим модулем 6</b>		<b>18</b>		
<b>Усього</b>		<b>144</b>		

### 6.1. Індивідуальне науково-дослідне завдання

Індивідуальне науково-дослідне (ІНДЗ) "Серверні веб-застосування" передбачає: систематизацію, закріплення, розширення теоретичних знань і практичних навичок із дисципліни та застосування їх при

вирішенні конкретних виробничих ситуацій; розвиток навичок самостійної роботи з науково-технічною документацією.

ІНДЗ "Серверні веб-застосування" з дисципліни "Веб-технології та веб-дизайн" видається студенту викладачем на початку вивчення дисципліни. Індивідуальне завдання виконується студентом самостійно. Студент має надати індивідуальне завдання для перевірки наприкінці останнього семестру. Оцінка за виконання індивідуального завдання враховується при виставленні загальної оцінки з дисципліни.

Тематика індивідуального завдання має носити проблемний характер. Студент має право самостійно обрати тему та зміст роботи з обов'язковим її узгодженням з викладачем.

У протилежному випадку тема має бути запропонована викладачем (варіанти тем наведені нижче).

У процесі виконання індивідуального завдання студент має опрацювати не менш п'яти літературних джерел з посиланнями на використання певної інформації з них у тексті роботи. При цьому робота має носити творчий характер і бути спрямованою на вирішення певної проблеми чи на висловлення особистого погляду автора роботи на питання, яке розглядається в роботі.

Індивідуальне завдання складається з: титульної сторінки; змісту; вступу; основної частини; висновків; списку використаної літератури, додатків до індивідуального завдання (при необхідності).

**Вступ** має розкривати актуальність обраної студентом теми, її проблематику, мету написання роботи.

**Основна частина** роботи (може включати декілька підрозділів) має включати характеристику сучасного стану проблеми, опис нормативної бази, погляд різних авторів на цю проблему, позитивні та негативні наслідки проблеми.

**Висновки** мають включати обґрунтовані відомості студента щодо досягнення мети роботи, можливо, рекомендації щодо підвищення ефективності веб-програмування, розробки веб-сайтів та веб-дизайну.

Обсяг індивідуального завдання повинен становити у друкованому варіанті 10 – 15 сторінок. Орієнтовна кількість сторінок у розділах: вступ – 1 с.; основна частина – 8 – 12 с.; висновки – 1 – 2 с.

## 6.2. Тематика індивідуального завдання



1. Аналіз поведінки браузерів при переключенні режимів за допомогою Doctype.
2. Безпечне завантаження зображень на сервер по HTTP.
3. Особливості CMS Joomla.
4. Порівняльний аналіз PHP та ASP.
5. Ефективне використання .htaccess.
6. Використання різних PHP-фреймворків.
7. Digest-автентифікація в PHP.
8. Переваги та недоліки ООП в PHP.
9. Порівняльний аналіз Smarty та Twig.
10. Тенденції розвитку сучасного Web-програмування.
11. Ефективність вибору засобів перевірки роботи із даними у формі між Javascript та PHP.
12. Фільтрування даних за допомогою PHP.
13. Організація безпечного пошуку на сайті.
14. Робота з електронною поштою за допомогою PHP.
15. Побудова дерева ієрархій за допомогою PHP та MySQL.
16. PHP та регулярні вирази.
17. Посторінковий вивід у PHP.
18. Визначення поточної кількості користувачів, які присутні на сайті та ефективне використання цих даних.
19. Універсальні скрипти для обробки web-форм.
20. Фотомонтаж за допомогою PHP.
21. Організація повної статистики відвідувань сайту.
22. Робота з банерними системами.
23. Захист сайту від повного завантаження.
24. Захист web-форм від автоматичної обробки.
25. Створення гостевої книги за допомогою PHP та MySQL.
26. Захист e-mail адрес на web-сторінках.
27. Використання XML в PHP.
28. Сортування стовбців таблиці без перевантаження.
29. Затиснення даних за допомогою PHP.
30. Нові можливості PHP5-Zend Engine 2.0.
31. Захист сайту: блокування IP-адреси.
32. Аспектно-орієнтована web-розробка та PHP.
33. Можливості JpGraph.

34. Специфікація і функції DOM в PHP.
35. Хешування паролів в PHP.
36. Абстрактні класи та інтерфейси у PHP 5.
37. Реалізація шаблонів на PHP.
38. Створення автоматичних тестів та середовище PHPUnit.
39. Створення таблиць Excel засобами PHP.
40. Використання XML Parser Functions для роботи із шаблонами.
41. Використання сокетів в PHP.
42. Особливості створення internet-магазину .
43. Система голосувань за допомогою PHP.
44. Особливості Javascript-бібліотеки Glassbox.
45. Підтвердження реєстрації у PHP.
46. RSS-парсер за допомогою PHP.
47. Аналіз та відображення логів.
48. Розширення PHP для роботи з mp3.
49. Створення водяних знаків за допомогою PHP.
50. Парсинг URL для "User Friendly URLs".
51. Вразливість використання SSI (include) в PHP.
52. Робота зі строками у файлі за допомогою PHP.
53. Трансформування XML + XSLT.
54. Стиснення сторінок у PHP.
55. Реалізація тем на сайті за допомогою PHP.
56. Обчислення часу виконання SQL-запиту.
57. Використання сервісу Whois у PHP.
58. Обробка XML-документу за допомогою Javascript.
59. Базове налаштування IIS.
60. Використання основних елементів веб-форми ASP.NET.
61. Застосування серверних елементів управління.
62. Збереження стану відображення елементів управління форми.
63. Методи реалізації коду в ASP.NET.
64. Обробка подій на сервері та клієнті.
65. Етапи та події життєвого циклу веб-сторінки.
66. Використання автоматичної зворотної відправки даних.
67. Процес зворотної передачі даних.
68. Використання майстра веб-сторінки.

69. Використання елементів управління користувача.
70. Використання елементів управління перевірки вводу даних.
71. Управління станом в ASP.NET на боці клієнта.
72. Управління станом в ASP.NET на боці сервера.
73. Налаштування та трасування веб-сайта.
74. Розгорнення веб-сайта.
75. Використання авторизації та автентифікації у веб-додатках.
76. З'єднання з сховищем даних ADO.NET.
77. Застосування багатофункціональних елементів управління даними.
78. Використання об'єктно-реляційного відображення даних.
79. Використання запитоподібного формату написання LINQ.
80. Використання процедурного формату написання LINQ.
81. AJAX на боці клієнта та на боці сервера.
82. Інструментарій ASP.NET AJAX Control Toolkit.
83. Створення веб-сервісів XML.
84. Реалізація веб-служб WCF.
85. Використання синтаксису Razor.
86. Робота з архітектурою "модель – відображення – контролер" в ASP.NET MVC.

### **6.3. Контрольні запитання для самодіагностики**

#### **Тема 1. Основи Web-програмування. Розробка Web-сторінок за допомогою HTML та CSS.**

1. З яких частин складається документ HTML?
2. У чому полягають правила сумісності синтаксису HTML з XML?
3. Чим відрізняється логічне і фізичне форматування тексту в документі HTML?
4. Поясніть різницю між правильним та дійсним документом.
5. Як використовується службова інформація в блоці

заголовка документа HTML?

6. Перелічіть основні елементи структури сторінки.
7. Які типи списків можуть зустрічатися в документі HTML?
8. За яких умов для зображення краще обрати один з графічних форматів GIF або JPG?
9. Чому слід завжди визначати альтернативний текст для зображень?
10. Поясніть різницю між глобальним і локальним посиланням. Яка роль тега <base>?
11. Чому слід завжди передбачати частину <noframes> в документі, що описує фреймову структуру?
12. Які є переваги і недоліки використання фреймів в документі HTML?
13. Для чого можуть використовуватися форми в документі HTML? Наведіть приклади.
14. Для чого призначені каскадні таблиці стилів (CSS)?
15. Якими способами можна додати інформацію про стилі у документ HTML?
16. Як визначити, яке правило задання стилів діє в конкретному випадку?
17. Поясніть різницю між абсолютним і відносним позиціонуванням.
18. Які є рівні організації сайту?
19. З чого складається інтерфейс веб-сайту?

## **Тема 2. Технологія CSS та її під-тримка браузером.**

1. Що таке каскадні таблиці стилів (CSS)?
2. Ідеологія і способи використання CSS?
3. Що таке складні селектори і правила їх використання?
4. Які властивості форматування елементів засобами CSS?
5. Які правила роботи зі шрифтами, кольором і фоном?
6. Які правила форматування блоків, форматування списків?

## **Тема 3. Блокова верстка сторінок веб-сайту.**

1. Що таке блокова модель елементів веб-сторінок?
2. Особливості блокових, строкових та інших видів html-елементів?
3. Як реалізувати зміну виду html-елементів за допомогою CSS.
4. Які види позиціонування елементів веб-сторінки?
5. Як здійснюється верстка веб-сторінок з фіксованою шириною, резиноюю версткою та еластичною версткою?

#### **Тема 4. Основи javascript.**

1. Які особливості мови сценаріїв javascript?
2. Основні оператори javascript?
3. Для чого призначені розгалуження і цикли?
4. Які ви знаєте оператори роботи з об'єктами у javascript.
5. Які особливості роботи з масив?
6. Для чого призначені функції у javascript та їх основні типи?
7. Як створюються об'єкти користувача у javascript?
8. Як організується передача інформації між сторінками сайту засобами мови javascript?
9. Які є способи включення сценаріїв JavaScript у документ HTML?
10. Чому сценарії не завжди слід описувати на початку документа?
11. Назвіть основні операції у мові сценаріїв JavaScript.
12. Порівняйте оператори розгалуження у мові сценаріїв JavaScript та у мові C++.
13. Порівняйте оператори циклів у мові сценаріїв JavaScript та у мові C++.
14. Назвіть оператори керування об'єктами у мові сценаріїв JavaScript.
15. Опишіть особливості масивів у мові сценаріїв JavaScript.
16. Як передаються параметри у функції у мові сценаріїв JavaScript?
17. Скільки складає час життя локальних і глобальних

змінних сценарія JavaScript?

18. Назвіть наперед визначені об'єкти JavaScript.

19. Яким чином можна створити власний об'єкт у мові сценаріїв JavaScript?

20. Яким способом можна передати інформацію між сторінками сайта?

### **Тема 5. Програмна взаємодія з HTML документами на основі DOM API.**

1. Переваги об'єктної моделі документа (DOM)?

2. Назвіть види вузлів DOM?

3. Які способи доступу до вузлів DOM за допомогою javascript?

4. Які принципи маніпуляція вузлами DOM за допомогою javascript?

5. Охарактеризуйте поняття події?

6. Які процедури обробки події ви знаєте?

### **Тема 6. Використання бібліотек javascript для розробки веб-сайтів. Бібліотека jQuery**

1. Які переваги популярних javascript-бібліотеки?

2. Охарактеризуйте концепції бібліотеки jQuery?

3. Які здійснюється доступ до вузлів DOM засобами бібліотеки jQuery?

4. Які принципи маніпуляції вузлами DOM за допомогою бібліотеки jQuery?

### **Тема 7. Взаємодія веб-клієнта з веб-сервером. Протокол HTTP та засоби моніторингу http-трафіка. Мови розроблення серверних сценаріїв.**

1. Які основні характеристики протоколу HTTP?

2. Назвіть загальну структуру http-повідомлень?

3. Назвіть основні методи протоколу HTTP та особливості.

4. Складова частина заголовку http-повідомлень?

5. Які особливості кодування даних http-повідомлень?

6. Основні засоби моніторингу http-трафіка?
7. Яка сутність поняття серверного сценарію?
8. Назвіть основні стандарти взаємодії серверних сценаріїв з веб-сервером.
9. Порівняйте мови розроблення серверних сценаріїв.

#### **Тема 8. Розробка веб-застосувань за допомогою мови PHP**

1. Які принципи дії методів вбудовування PHP-коду?
2. Як здійснюється вивід результатів роботи скрипта?
3. Які особливості змінних у PHP?
4. Які типи даних та особливості ініціалізації змінних?
5. Як провести перевірку існування змінних?
6. Як здійснюється перетворення типів даних?
7. Які принципи роботи з масивами?
8. Які особливості функцій для роботи зі строками, числами, датою та часом?
9. Що таке функції користувача?
10. Як виявити помилки у програмі?

#### **Тема 9. Обробка даних http-запитів мовою PHP. Механізми управління станом веб-застосувань**

1. Що таке змінні оточення?
2. Що таке глобальні масиви?
3. Які особливості обробки даних форми?
4. Які принципи передачі файлів в http-повідомленнях та їх обробка мовою PHP?
5. Як встановити http-заголовки за допомогою PHP?
6. Які принципи використання Cookie, їх обробки та встановлення?
7. Які особливості механізму управління сеансом користувача?
8. Назвіть способи ідентифікації сесії (сеансу)?
9. Як реалізувати ініціювання сесії?
10. Які принципи додавання та видалення даних (змінних) в сесії?

## **Тема 10. Особливості ООП в PHP**

1. Які особливості створення класів?
2. Які принципи визначення властивостей класів?
3. Призначення метода-конструктора?
4. Назвіть особливості управління доступом до класів та елементів класів?
5. В яких випадках використовуються статичні методи та властивості?
6. Для чого необхідні постійні властивості?
7. Чим відрізняються абстрактні класи та інтерфейси?
8. Для чого використовуються закінчені класи та методи?
9. Що таке простори імен?

## **Тема 11. Інтерфейси взаємодії PHP-застосувань з СУБД**

1. Які засоби PHP для роботи з текстовими, DBM та реляційними базами даних?
2. Що таке PDO та його основні переваги?
3. Які принципи з'єднання із БД SQL?
4. Як виконуються запити до БД SQL?
5. Як здійснюються основні операції вибору, додавання, оновлення та видалення даних з таблиць БД?
6. Як визначити кількості рядків у наборі даних?
7. Які основні типи даних полів таблиць у MySQL?
8. Опишіть правила створення бази даних в MySQL.
9. Як створити облікові записи користувачів бази даних?
10. Опишіть механізм створення та видалення таблиці і бази даних.

## **Тема 12. DOM XML. Програмна обробка XML-документів з допомогою XML DOM та Simple XML**

1. Яка Структура XML-документів?
2. Назвіть приклади ефективного використання XML.
3. Назвіть основні PHP API для робіт з XML.
4. Опишіть процес створення XML-документа за допомогою PHP DOM XML та Simple XML.
5. Як здійснюється доступ до xml-елементів за допомогою PHP DOM XML та Simple XML?



6. Як реалізується додавання, зміна, видалення елементів та їх атрибутів в XML-документах за допомогою PHP DOM XML та Simple XML?

7. Проведіть порівняння PHP DOM XML та Simple XML. XPath та PHP.

### **Тема 13. Розміщення веб-сайта на сервері. Створення інтерактивних елементів ASP.NET.**

1. Які базові принципи налаштування IIS?

2. Які особливості використання IIS при розробці веб-сайта?

3. Які основні компоненти веб-додатка ASP.NET?

4. Назвіть основні елементи веб-форми.

5. Що таке серверні елементи управління та для чого вони розроблені?

6. В чому полягає особливість збереження стану відображення елементів управління форми?

7. Які методи реалізації коду ви знаєте?

8. В чому полягає особливість обробки подій на сервері та клієнті?

9. Яка різниця між типами, компонентами і класами?

10. Назвіть етапи та події життєвого циклу веб-сторінки.

11. Що таке процес зворотної передачі даних?

12. В яких випадках використовується автоматична зворотна відправка даних?

13. Для чого існує майстер веб-сторінки?

14. В яких випадках застосовуються елементи управління користувача?

15. Які способи управління станом в ASP.NET ви знаєте?

16. Для чого використовується трасування веб-сайта?

17. В чому полягає різниця авторизація та автентифікація?

### **Тема 14. Інтерфейси взаємодії ASP.NET-застосувань з СУБД.**

1. В чому полягає технологія доступу до даних ADO.NET?

2. Які ви знаєте постачальники даних ADO.NET?

3. Що таке об'єктна модель ADO.NET?

4. Назвіть класи, які основані на з'єднанні та класи, які

основані на змістовності.

5. Що таке прив'язка даних?
6. Для чого використовується мова SQL?
7. Назвіть основні багатофункціональні елементи управління даними.
8. В чому полягає сутність об'єктно-реляційного відображення?
9. Яка різниця між LINQ та SQL?
10. Назвіть типи реалізації LINQ.
11. Які формати написання LINQ ви знаєте?
12. В чому полягає відкладене та негайне виконання запитів?

### **Тема 15. Уведення в технологію AJAX**

1. В чому полягає різниця ASP.NET AJAX на боці клієнта та на боці сервера?
2. Що таке асинхронний JavaScript?
3. Які серверні елементи управління AJAX ви знаєте?
4. Що таке часткова візуалізація та синхронізація по часу оновлення?
5. В чому полягає особливість об'єктно-орієнтованого програмування в JavaScript.?
6. Для чого призначений інструментарій ASP.NET AJAX Control Toolkit?

### **Тема 16. Уведення в технологію AJAX. Реалізація асинхронної взаємодії браузера з веб-сервером з допомогою технології AJAX.**

1. В чому полягає роль веб-сервісів XML?
2. Що таке WSDL?
3. Які компоненти веб-сервісів ви знаєте?
4. Які формати повідомлень ви знаєте?
5. Які протоколи зв'язку веб-сервісів ви знаєте?
6. Яка роль веб-служб WCF?
7. Для чого використовуються контракти WCF, прив'язки та адреси?
8. Назвіть основні принципи побудови служби WCF.

### **Тема 17. Основи розробки веб-порталів за допомогою CMF ASP.NET MVC.**

1. Назвіть основну ідею архітектуру "модель - відображення - контролер"?
2. Що таке контролери та дії?
3. Як отримати дані, які введені?
4. Що таке генерація виводу?
5. В чому різниця між не типізованими та типізованими відображеннями?
6. Для чого використовується часткове відображення?
7. Які особливості моделі даних та моделі відображення?
8. Які переваги синтаксису Razor.

## **7. Індивідуально-консультативна робота**

Індивідуально-консультативна робота здійснюється за графіком індивідуально-консультативної роботи у формі індивідуальних занять, консультацій, перевірки виконання індивідуальних завдань, перевірки та захисту завдань, що винесені на поточний контроль, тощо.

Формами організації індивідуально-консультативної роботи є:

- а) за засвоєнням теоретичного матеріалу:  
консультації: індивідуальні (запитання – відповідь), групові (розгляд типових прикладів – ситуацій);
- б) за засвоєнням практичного матеріалу:  
консультації індивідуальні та групові;
- в) для комплексної оцінки засвоєння програмного матеріалу:  
індивідуальне здавання виконаних робіт.

## **8. Методи навчання**

У процесі викладання навчальної дисципліни для активізації навчально-пізнавальної діяльності студентів передбачене застосування як активних, так і інтерактивних навчальних технологій, серед яких: лекції проблемного характеру, міні-лекції, робота в малих групах, мозкові атаки, кейс-метод, презентації, ознайомлювальні (початкові) ігри, метод проектної роботи, комп'ютерні симуляції, метод Дельфі, метод сценаріїв, банки візуального супроводу (табл. 8.1 і 8.2).

Таблиця 8.1

## Розподіл форм та методів активізації процесу навчання за темами навчальної дисципліни

Методики активізації процесу навчання	Практичне застосування навчальних технологій
<p>Проблемні лекції направлено на розвиток логічного мислення студентів, коло питань теми обмежується двома-трьома ключовими моментами, використовується досвід закордонних навчальних закладів з роздачею студентам під час лекцій друкованого матеріалу та виділенням головних висновків з питань, що розглядаються. При читанні лекцій студентам даються питання для самостійного розміркування, на які лектор відповідає сам, не чекаючи відповідей студентів</p>	<p>Проблемна лекція з питання: "Структура і принципи Веб. Уведення в HTML" (за темою 1).</p> <p>Проблемна лекція з питань: "Блокова верстка сторінок веб-сайта" (за темою 3)</p>
<p>Робота в малих групах дає змогу структурувати лабораторні заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду соціального спілкування</p>	<p>Робота в малих групах при розробленні корпоративного веб-сайта підприємства з використанням відповідних технологій (Л.р. № 4 за модулем 1 та л.р. № 6 за модулем 2)</p>
<p>Семінари-діскусії передбачають обмін думками і поглядами учасників з приводу даної теми, а також розвивають мислення, допомагають формувати погляди і переконання, виробляють вміння формулювати думки і висловлювати їх, вчать оцінювати пропозиції інших людей, критично підходити до власних поглядів</p>	<p>Проблемне повідомлення та дискусія при роботі студентів в проектній команді з питань: "Розроблення корпоративного веб-сайта підприємства з використанням всіх вивчених технологій". (лабораторна робота № 4 за модулем 5)</p>
<p>Презентації – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи</p>	<p>Презентація індивідуально вирішеного завдання</p>

Основні відмінності активних та інтерактивних методів навчання від традиційних визначаються не тільки методикою і технікою викладання, але й високою ефективністю навчального процесу, який виявляється у: високій мотивації студентів; закріпленні теоретичних знань на практиці; підвищенні самосвідомості студентів; формуванні здатності приймати самостійні рішення; формуванні здатності до ухвалення колективних рішень; формуванні здатності до соціальної інтеграції; набуття навичок вирішення конфліктів; розвитку здатності до знаходження компромісів.

**Лекції проблемного характеру** – один із найважливіших елементів проблемного навчання студентів. Вони передбачають поряд із розглядом основного лекційного матеріалу встановлення та розгляд кола проблемних питань дискусійного характеру, які недостатньо розроблені

в науці й мають актуальне значення для теорії та практики. Лекції проблемного характеру відрізняються поглибленою аргументацією матеріалу, що викладається. Вони сприяють формуванню у студентів самостійного творчого мислення, прищеплюють їм пізнавальні навички. Студенти стають учасниками наукового пошуку та вирішення проблемних ситуацій.

**Міні-лекції** передбачають викладення навчального матеріалу за короткий проміжок часу й характеризуються значною ємністю, складністю логічних побудов, образів, доказів та узагальнень. Вони проводяться, як правило, як частина заняття-дослідження. Міні-лекції відрізняються від повноформатних лекцій значно меншою тривалістю. Зазвичай міні-лекції тривають не більше 10 – 15 хвилин і використовуються для того, щоб стисло донести нову інформацію до всіх слухачів. Міні-лекції часто застосовуються як частини цілісної теми, яку бажано викладати повноформатною лекцією, щоб не втомлювати аудиторію. Тоді інформація надається по черзі кількома окремими сегментами, між якими застосовуються інші форми й методи навчання.

**Робота в малих групах** дає змогу структурувати практично-семінарські заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду соціального спілкування.

**Мозкові атаки** – метод розв'язання невідкладних завдань, сутність якого полягає в тому, щоб висловити якомога більшу кількість ідей за дуже обмежений проміжок часу, обговорити і здійснити їх селекцію.

**Презентації** – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи звіту про виконання індивідуальних завдань, проектних робіт. Презентації можуть бути як індивідуальними, наприклад виступ одного слухача, так і колективними, тобто виступи двох та більше слухачів.

**Метод Дельфі** використовується з метою досягнення консенсусу в експертних оцінках і передбачає надання можливості висловити свої думки групі експертів, що працюють індивідуально в різних місцях. При виборі управлінського рішення за цим методом академічну групу

розділяють, наприклад, на п'ять малих груп. Чотири групи є робочими, вони розробляють і приймають управлінське рішення, а п'ята група є експертною. Аналіз та варіанти управлінських рішень робочих груп усереднюються цією групою. Експертна група може бути поділена за спеціалізаціями.

**Комп'ютерна симуляція (гра)** – це метод навчання, що спирається на використання спеціальних комп'ютерних програм, за допомогою яких можливе віртуальне моделювання бізнес-процесу. Студенти можуть змінювати параметри й дані, приймати рішення та аналізувати наслідки таких рішень. Метою використання даного методу є розвиток системного мислення студентів, їх здібностей до планування, формування вмінь розпізнавати й аналізувати проблеми, порівнювати й оцінювати альтернативи, приймати оптимальні рішення й діяти в умовах обмеженого часу.

**Метод сценаріїв** полягає в розробці ймовірних моделей поведінки та розвитку конкретних явищ у перспективі.

**Банки візуального супроводу** сприяють активізації процесу навчання за темами навчальної дисципліни за допомогою наочності.

## 9. Методи контролю

Система оцінювання сформованих компетентностей (див. табл. 2.1) у студентів враховує види занять, які згідно з програмою навчальної дисципліни передбачають лекційні, лабораторні заняття, а також виконання самостійної роботи. Оцінювання сформованих компетентностей у студентів здійснюється за накопичувальною 100-бальною системою. Відповідно до Тимчасового положення "Про порядок оцінювання результатів навчання студентів за накопичувальною бально-рейтинговою системою" ХНЕУ ім. С. Кузнеця, контрольні заходи включають:

**поточний контроль**, що здійснюється протягом семестру під час проведення лекційних, лабораторних занять і оцінюється сумою набраних балів (в 5-му та 6-му семестрі, в яких метод контролю є залік, максимальна сума – 100 балів, а мінімальна сума, що дозволяє студенту отримати залік – 60 балів; в 7-му семестрі, в якому метод контролю є іспит максимальна сума – 60 балів, а мінімальна сума, що дозволяє студенту скласти іспит – 35 балів);

**модульний контроль**, що проводиться з урахуванням поточного контролю за відповідний змістовий модуль і має на меті *інтегровану* оцінку результатів навчання студента після вивчення матеріалу з логічно завершеної частини дисципліни – змістового модуля;

**підсумковий/семестровий контроль** в 5-му та 6-му семестрі проводиться у формі заліку, а в 7-му семестрі у формі семестрового екзамену, відповідно до графіку навчального процесу.

Приклад тестових завдань за модулем 1

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс третій  
Модуль № 1

### **Завдання № 1**

#### **1. Дайте відповідь на такі запитання:**

- а) Для чого використовується DOCTYPE у HTML-кодi?
- б) Як використовується службова інформація в блоці заголовка документа HTML?
- в) Поясніть різницю між глобальним і локальним посиланням. Яка роль тега <base>?
- г) Які селектори CSS вам відомі?

#### **2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:**

При реалізації сайт, який включає достатньо багато сторінок, найбільш ефективним способом використання CSS є:

- а) визначення стилю в елементі розмітки;
- б) розміщення опису стилю в заголовку документа в елементі STYLE;
- в) розміщення посилання на зовнішній опис через елемент LINK.

Затверджено на засіданні кафедри "Інформаційних систем"

протокол № від р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,  
доц. Д. В. Гриньов



## Приклад тестових завдань за модулем 2

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс третій  
Модуль № 2

### Завдання № 1

#### 1. Дайте відповідь на такі запитання:

- а) Назвіть способи розміщення коду JavaScript на HTML -сторінці.
- б) Які види масивів існують в JavaScript?
- в) Які Вам відомі оператори в JavaScript?
- г) Для чого використовують оператор var?

#### 2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:

При реалізації сайт, який включає достатньо багато сторінок, найбільш ефективним способом використання JavaScript є:

- а) включення JavaScript в елемент розмітки;
- б) розміщення JavaScript в тілі HTML-документа в тегах `<script>`  
`</script>`;
- в) розміщення посилання на зовнішній файл з кодом JavaScript в якості значення атрибута `src` тега `<script>`.

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол № \_\_\_\_\_ від \_\_\_\_\_ р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,  
доц. Д. В. Гриньов

## Приклад тестових завдань за модулем 3

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс третій  
Модуль № 3

### Завдання № 1

#### 1. Дайте відповідь на такі запитання:

- а) Яким чином можна звернутися до змінної в PHP?
- б) Чим відрізняються оператори PHP `include` і `require` від операторів `include_once` і `require_once`?
- в) У чому особливість області видимості змінних в PHP порівняно з такими мовами як C++ і C#?
- г) Назвіть методи протоколу HTTP.

#### 2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:

Більш надійний метод передачі змінних форми до оброблювача, особливо при авторизації користувачів, є:

- а) GET;
- б) POST;
- в) PUT.

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол № \_\_\_\_\_ від \_\_\_\_\_ р.

Зав. кафедри \_\_\_\_\_

проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_

доц. В. В. Огурцов,  
доц. Д. В. Гриньов

## Приклад тестових завдань за модулем 4

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс третій  
Модуль № 4

### Завдання № 1

#### 1. Дайте відповідь на такі запитання:

- а) За допомогою якої PHP-функції можна підключитися до сервера MySQL?
- б) За допомогою якої PHP-функції можна вибрати необхідну БД MySQL?
- в) За допомогою якої PHP-функції можна виконати запит до БД MySQL?
- г) За допомогою якої PHP-функції можна зберегти рядок результату виконання запиту в асоціативний масив?

#### 2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:

Що реалізує наступний SQL запит "SELECT FirstName FROM User":

- а) запис імені FirstName в таблицю User;
- б) читання даних всіх строк з поля FirstName з таблиці User;
- в) видалення імені FirstName з таблиці User.

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол №     від                     р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,  
доц. Д. В. Гриньов

## Приклад тестових завдань за модулем 5

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс четвертий  
Модуль № 5

### Завдання № 1

#### 1. Дайте відповідь на такі запитання:

- а) Що таке серверні елементи управління та для чого вони розроблені?
- б) В яких випадках використовується автоматична зворотна відправка даних?
- в) Назвіть етапи та події життєвого циклу веб-сторінки.
- г) Для чого використовується трасування веб-сайта?

#### 2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:

Який варіант створення в кодї розмітки HTML-елементу управління створює об'єкт на боці серверу та дозволяє керувати ним в кодї логіки:

- а) `<input type="submit" name="Btn" runat="server" value="Отправить"/>`;
- б) `<input type="submit" id="Btn" runat="server" value="Отправить"/>`;
- в) `<input type="submit" name="Btn" id="Btn" value="Отправить"/>`.

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол № \_\_\_\_\_ від \_\_\_\_\_ р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,  
доц. Д. В. Гриньов

## Приклад тестових завдань за модулем 6

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс четвертий  
Модуль № 6

### Завдання № 1

#### 1. Дайте відповідь на такі запитання:

- а) Що таке асинхронний JavaScript?
- б) Для чого призначений інструментарій ASP.NET AJAX Control Toolkit?
- в) В чому полягає роль веб-сервісів?
- г) Назвіть основну ідею архітектуру "модель – відображення – контролер"?

#### 2. Оберіть правильний варіант твердження та обґрунтуйте свій вибір:

При стандартній конфігурації маршрутизації URL в ASP.NET MVC оберіть правильні відповіді при яких буде визвано метод дії Index() контролера Home:

- а) `http://вашСайт/;`
- б) `http://вашСайт/Home;`
- в) `http://вашСайт/Home/Index.`

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол № \_\_\_\_\_ від \_\_\_\_\_ р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,  
доц. Д. В. Гриньов

## Проведення підсумкового іспиту

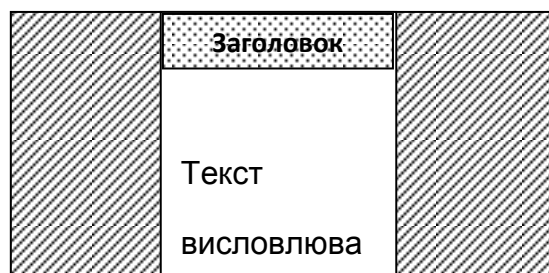
Іспит здійснюється на комп'ютерах за екзаменаційними білетами. Екзаменаційний білет складається з шести завдань. Результаті іспиту оцінюються за 40-бальною системою. Підсумкова оцінка за іспит є сумою оцінок за кожне завдання.

Приклад екзаменаційного білета з дисципліни:

Міністерство освіти і науки України  
Харківський національний економічний університет імені Семена Кузнеця  
Кафедра інформаційних систем  
Дисципліна: "Веб-технології та веб-дизайн"  
Напрямок підготовки "Комп'ютерні науки"  
Курс четвертий

### Білет № 1

**Завдання 1.** Створити HTML-документ (назвати **page1.html**), який містить висловлювання відомою людини. Контент на сторінці розмістити в блоці фіксованої ширини. Блок для заголовку повинен відрізнятися кольором та шрифтом, причому текст повинен форматуватися по центру.



**Завдання 2.** Створити HTML-сторінку (назвати **page2.html**). Вставити в створений HTML-документ таблицю такого виду:




**Завдання 3.** Створити HTML-сторінку (назвати **page3.html**). Вставити в створений документ HTML-код, що відповідає селектору у CSS-правилі, що наведено нижче.

***#content p.fio***{font-weight: bold;}

**Завдання 4.** Створити HTML-сторінку (назвати **page4.html**). За допомогою JavaScript вивести поточний день тижня. Бажано розташувати javascript-код у окремому файлі (назвати **javascript.js**).

Обрати бажану серверну технологію розробки веб-сайтів: PHP або ASP.NET. В залежності від обраної технології реалізувати наступні завдання 5 і 6.

**Завдання 5.** Створити веб-додаток який повинен формувати звіт відповідно до таблиці 1. Для виконання завдання використати базу даних, яка знаходиться на сервері (isserv) у каталозі exam-web та її назва відповідає варіанта білета.

Таблиця 1

Інформація про закріплених тварин за ветеринарами

Ім'я ветеринара	Прізвище ветеринара	Прізвисько тварини	Порода тварини

**Завдання 6.** Реалізувати у веб-додатку, що створений у завданні 4 функції створення, видалення, редагування, пошуку (фільтрації) записів БД, посторінкової навігації.

Затверджено на засіданні кафедри "Інформаційних систем"  
протокол № \_\_\_\_\_ від \_\_\_\_\_ р.

Зав. кафедри \_\_\_\_\_ проф. В. С. Пономаренко

Екзаменатор \_\_\_\_\_ доц. В. В. Огурцов,

доц. Д. В. Гриньов



## Критерії оцінювання екзаменаційної роботи

Для оцінки рівня виконання студентами завдань використовуються такі критерії:

Завдання 1 та 6 оцінюється від 0 до 8 балів відповідно до наступної шкали:

- |         |   |
|---------|---|
| 8 балів | Завдання виконано в повному обсязі. Програма працює правильно на усіх тестах. Інтерфейс та вихідний код програми задовольняють встановленим вимогам. При виконанні задачі були використані нестандартні підходи які не суперечать існуючим правилам та дозволяють оптимізувати код. |
| 7 балів | Завдання виконано в повному обсязі. Програма працює правильно на усіх тестах. Інтерфейс та вихідний код програми задовольняють встановленим вимогам.  |
| 6 балів | Завдання виконано в повному обсязі. Програма працює правильно. Є невеликі зауваження до організації інтерфейсу користувача та (або) вихідного коду програми.  |
| 5 бала  | Завдання в основному виконано. Програма працює правильно, але одна з її функціональних можливостей реалізована з порушенням вимог, вказаних в завданні.   |
| 4 бали  | Завдання виконано, але не в повному обсязі. Програма працює, але не реалізована одна з функціональних вимог, вказаних в завданні  |
| 3 бали  | Завдання виконано, але не в повному обсязі. Програма працює, але дві та більше з функціональних можливостей програми реалізовані з порушенням вимог, вказаних в завданні.   |
| 2 бали  | Завдання не виконано. Програма запускається, але не реалізовані дві та більше з функціональних вимог, вказаних в завданні.  |
| 1 бал   | Програма не запускається, але є програмний код, розроблений студентом.  |
| 0 балів | Програма відсутня.  |

Завдання 2, 3, 4, 5 оцінюється від 0 до 6 балів відповідно до наступної шкали:

- |         |  |
|---------|--|
| 6 балів | Завдання виконано в повному обсязі. Програма працює правильно на усіх тестах. Інтерфейс та вихідний код програми |
|---------|--|

задовольняють встановленим вимогам.

- 5 балів Завдання виконано в повному обсязі. Програма працює правильно. Є невеликі зауваження до організації інтерфейсу користувача та (або) вихідного коду програми.
- 4 бали Завдання в основному виконано. Програма працює правильно, але одна з її функціональних можливостей реалізована з порушенням вимог, вказаних в завданні.
- 3 бали Завдання виконано, але не в повному обсязі. Програма працює, але не реалізована одна з функціональних вимог, вказаних в завданні, або дві та більше з функціональних можливостей програми реалізовані з порушенням вимог, вказаних в завданні.
- 2 бали Завдання не виконано. Програма запускається, але не реалізовані дві та більше з функціональних вимог, вказаних в завданні.
- 1 бал Програма не запускається, але є програмний код, розроблений студентом.
- 0 балів Програма відсутня.

## 10. Розподіл балів, які отримують студенти

Отримання балів студентами у семестрах 5 і 6 за видами занять здійснюється за наступною схемою.

Таблиця 10.1

### Схема оцінювання за 100-бальною накопичувальною шкалою у семестрах 5 і 6

Форми навчання	Навчальні тижні																	Сума
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Лекції	0,5		0,5		0,5		0,5		0,5		0,5		0,5		0,5		0,5	<b>4,5</b>
Лабораторні заняття	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	<b>8,5</b>
Захист лабораторних робіт			6			8			12			10			12		6	<b>54</b>
Експрес-опитування	1	1		1	1		1			1	1		1	1				<b>9</b>
Поточні КР								12									12	<b>24</b>
Сума балів	<b>2</b>	<b>1,5</b>	<b>7</b>	<b>1,5</b>	<b>2</b>	<b>8,5</b>	<b>2</b>	<b>13</b>	<b>13</b>	<b>1,5</b>	<b>2</b>	<b>11</b>	<b>2</b>	<b>1,5</b>	<b>13</b>	<b>13</b>	<b>7</b>	<b>100</b>

Отримання балів студентами у семестру 7 за видами занять здійснюється за наступною схемою.

Таблиця 10.2

**Схема оцінювання за 100-бальною накопичувальною шкалою у семестру 7**

Форми навчання	Навчальні тижні																	Сума
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Лекції	0,5		0,5		0,5		0,5		0,5		0,5		0,5		0,5		0,5	<b>4,5</b>
Лабораторні заняття	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	<b>8,5</b>
Захист лабораторних робіт			7		7			7			7			7			7	<b>42</b>
Експрес-опитування						1								1				<b>2</b>
Поточні КР																	3	<b>3</b>
Екзамен																		<b>40</b>
Сума балів	<b>1</b>	<b>0,5</b>	<b>8</b>	<b>0,5</b>	<b>8</b>	<b>1,5</b>	<b>1</b>	<b>7,5</b>	<b>1</b>	<b>0,5</b>	<b>8</b>	<b>0,5</b>	<b>1</b>	<b>8,5</b>	<b>1</b>	<b>3,5</b>	<b>8</b>	<b>100</b>

Отримання балів студентами у семестру 5 за темами змістовних модулів здійснюється за наступною схемою.

Таблиця 10.3

**Отримання балів студентами для заліку в 5 семестрі**

Поточне тестування та самостійна робота						Сума
Змістовий модуль 1			Змістовий модуль 2			
T1	T2	T3	T6	T7	T6	100
16	17	17	16	17	17	

T1, T2 ... T6 – теми змістових модулів.

Отримання балів студентами у семестру 6 за темами змістовних модулів здійснюється за наступною схемою.

Таблиця 10.4

**Отримання балів студентами для заліку в 6 семестрі**

Поточне тестування та самостійна робота						Сума
Змістовий модуль 3			Змістовий модуль 4			
T7	T8	T9	T10	T11	T12	100

16	17	17	16	17	17	
----	----	----	----	----	----	--

T7, T8 ... T12 – теми змістових модулів.

Отримання балів студентами у семестру 7 за темами змістовних модулів здійснюється за такою схемою.

Таблиця 10.5

### Отримання балів студентами для іспиту в 7 семестрі

Поточне тестування та самостійна робота					Підсумковий тест (екзамен)	Сума
Змістовий модуль 5		Змістовий модуль 6			40	100
T13	T14	T15	T16	T17		
12	12	12	12	12		

T13, T14 ... T17 – теми змістових модулів.

Максимальну кількість балів, яку може накопичити студент протягом тижня за формами та методами навчання, наведено в табл. 10.6.

Таблиця 10.6

### Розподіл балів за тижнями

Теми змістового модуля		Лекційні заняття	Лабораторні заняття	Захист лабораторних робіт	Експрес-опитування	Поточні КР	Захист ІНДЗ	Усього
1		2	3	4	5	6	7	8
Змістовий модуль 1	Тема 1	1 тиждень	0,5	0,5		1		2
		2 тиждень		0,5		1		1,5
	Тема 2	3 тиждень	0,5	0,5	6			7
		4 тиждень		0,5		1		1,5
		5 тиждень	0,5	0,5		1		2
	Тема 3	6 тиждень		0,5	8			8,5
		7 тиждень	0,5	0,5		1		2
		8 тиждень		0,5			12	12,5
Змістовий модуль 2	Тема 4	9 тиждень	0,5	0,5	12			13
		10 тиждень		0,5		1		1,5
		11 тиждень	0,5	0,5		1		2
	Тема 5	12 тиждень		0,5	10			10,5
		13 тиждень	0,5	0,5		1		2
		14 тиждень		0,5		1		1,5
	Тема 6	15 тиждень	0,5	0,5	12			13
		16 тиждень		0,5			12	12,5

		17 тиждень	0,5	0,5	6				7
Усього			4,5	8,5	54	9	24	0	100

Закінчення табл. 10.6

		1	2	3	4	5	6	7	8
Змістовий модуль 3	Тема 7	1 тиждень	0,5	0,5		1			2
		2 тиждень		0,5		1			1,5
	Тема 8	3 тиждень	0,5	0,5	6				7
		4 тиждень		0,5		1			1,5
		5 тиждень	0,5	0,5		1			2
	Тема 9	6 тиждень		0,5	8				8,5
		7 тиждень	0,5	0,5		1			2
		8 тиждень		0,5				9	9,5
Змістовий модуль 4.	Тема 10	9 тиждень	0,5	0,5	12				13
		10 тиждень		0,5		1			1,5
		11 тиждень	0,5	0,5		1			2
	Тема 11	12 тиждень		0,5	10				10,5
		13 тиждень	0,5	0,5		1			2
		14 тиждень		0,5		1		6	7,5
	Тема 12	15 тиждень	0,5	0,5	12				13
		16 тиждень		0,5				9	9,5
		17 тиждень	0,5	0,5	6				7
Усього			4,5	8,5	54	9	18	6	100
Змістовий модуль 3.	Тема 13	1 тиждень	0,5	0,5					1
		2 тиждень		0,5					0,5
		3 тиждень	0,5	0,5	7				8
		4 тиждень		0,5					0,5
	Тема 14	5 тиждень	0,5	0,5	7				8
		6 тиждень		0,5		1			1,5
		7 тиждень	0,5	0,5					1
		8 тиждень		0,5	7				7,5
Змістовий модуль 4.	Тема 15	9 тиждень	0,5	0,5					1
		10 тиждень		0,5					0,5
		11 тиждень	0,5	0,5	7				8
	Тема 16	12 тиждень		0,5					0,5
		13 тиждень	0,5	0,5					1
		14 тиждень		0,5	7	1			8,5
	Тема 17	15 тиждень	0,5	0,5					1
		16 тиждень		0,5				3	3,5
		17 тиждень	0,5	0,5	7				8
Усього			4,5	8,5	42	2	3	0	60

Підсумкова оцінка з навчальної дисципліни визначається відповідно до Тимчасового положення "Про порядок оцінювання результатів навчання студентів за накопичувальною бально-рейтинговою системою" ХНЕУ ім. С. Кузнеця (табл. 10.7).

Оцінки за цією шкалою заносяться до відомостей обліку успішності,

індивідуального навчального плану студента та іншої академічної документації.

Таблиця 10.7

### Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
82 – 89	B	добре	
74 – 81	C		
64 – 73	D	задовільно	
60 – 63	E		
35 – 59	FX	незадовільно	не зараховано
1 – 34	F		

## 11. Рекомендована література

### 11.1. Основна

1. Будилов В. А. Основы программирования для Интернета / В. А. Будилов. – СПб. : БХВ-Петербург, 2003. – 736 с.
2. Глушков С. В. Программирование Web-страниц / С. В. Глушков, И. А. Жакин, Т. С. Хачиров. – Х. : Фолио, 2005. – 390 с.
3. Калиновский А. И. Юзабилити: как сделать сайт удобным / А. И. Калиновский. – Мн. : "Новое знание", 2005. – 220 с.
4. Кристофер Шмитт CSS. Рецепты программирования / Кристофер Шмитт. – СПб. : "БХВ-Петербург", 2007. – С. 592.
5. Лещев Д. Создание интерактивного web-сайта : учебный курс / Д. Лещев. – СПб. : Питер, 2003. – 544 с.
6. Ломов А. Ю. HTML, CSS, скрипты: практика создания сайтов / А. Ю. Ломов. – СПб. : БХВ-Петербург, 2006. – 416 с.
7. Мак-Дональд М. Microsoft ASP.NET 4 с примерами на C# 2010 для профессионалов, 4-е изд. / М. Мак-Дональд, А. Фримен, М. Шпуста – М. : ООО "И.Д. Вильямс", 2011. – 1424 с.: ил. Парал. тит. англ.
8. Пономаренко В. С. Основы технологий Internet : учебное

пособие / В. С. Пономаренко, С. В. Минухин, И. А. Торохтий. – Х. : Изд. ХГЭУ, 2001. – 256 с.

9. Прохоренок Н. HTML, Javascript, PHP и MySQL. Джентльменский набор Web-мастера / Н. Прохоренок. – СПб. : БХВ-Петербург, 2008. – 640 с.

10. Сандереон Стивен. ASP.NET MVC Framework с примерами на C# для профессионалов / Сандереон Стивен ; пер. с англ. – М. : ООО "И.Д. Вильямс", 2010. – 560 с.

11. Создание Web-страниц и Web-сайтов. Самоучитель / под ред. В. Н. Печникова. – М. : Изд-во Триумф, 2006. — 464 с.

12. Соколов С. А. HTML и CSS в примерах, типовых решениях и задачах. Профессиональная работа / С. А. Соколов. – М. : "Вильямс", 2007. – 416 с.

13. Томсон Л. Разработка Web-приложений на PHP и MySQL – пер. с англ. – 2-е изд., испр. – СПб. : "ДиаСофтЮП", 2003. – 672 с.

14. Троелсен Э. Язык программирования C# 2010 и платформа .NET 4.0, 5-е изд. / Э. Троелсен ; пер. с англ. – М. : ООО "И.Д. Вильямс", 2011. – 1392 с.

15. Ульман Л. Основы программирования на PHP / Л. Ульман. – М. : ДМК Пресс, 2001. – 288 с.

16. Эрик А. Мейер CSS-каскадные таблицы стилей: подробное руководство / А. Эрик Мейер. – М. : "Символ", 2006. — С. 576.

17. Microsoft ASP.NET 2.0. Базовый курс. Мастер-класс / пер. с англ. – М. : Издательство "Русская Редакция" ; СПб. : Питер, 2007. – 688 с.

## **11.2. Додаткова**

18. Валентайн Ч. XHTML / Ч. Валентайн, К. Минник. – М. : Издательский дом "Вильямс", 2001. – 480 с.

19. Дарнелл Р. JavaScript: справочник / Р. Дарнелл. – СПб. :

"Питер", 2000. – 192 с.

20. Зельдман Д. Web-дизайн по стандартам / Д. Зельдман. – СПб. : Изд-во: НТ Пресс, 2005. – 440 с.

21. Кирсанов Д. Веб-дизайн / Д. Кирсанов. – СПб. : "Символ-Плюс", 1999. – 376 с.

22. Круг С. Веб-дизайн / С. Круг. – СПб. : "Символ-Плюс", 2001. – 195 с.

23. Кузнецов М. Объектно-ориентированное программирование на PHP / М. Кузнецов, И. Симдянов. – СПб. : "БХВ-Петербург", 2007. – 608 с.

24. Моррисон М. HTML и XML. Быстро и эффективно / М. Моррисон. – СПб. : "Питер", 2005. – 302 с.

25. Нильсен Я. Веб-дизайн / Я. Нильсен. – СПб. : "Символ-Плюс", 2000. – 512 с.

26. Рихтер Дж. Программирование на платформе Microsoft .NET Framework / Дж. Рихтер ; пер. с англ. – 2-е изд., испр. – М. : Издательско-торговый дом "Русская Редакция", 2003. – 512 с.

27. Хольцшлаг М. Э. Использование HTML и XHTML / М. Э. Хольцшлаг. – М. : ВИЛЬЯМС, 2003. – 728 с.

28. Шелдон Р. MySQL: базовый курс / Р. Шелдон, Д. Мойе. – М. : "Диалектика", 2007. – 880 с.

29. Шилдг Г. Полный справочник по C#. / Г. Шилдг ; пер. с англ. – М. : Издательский дом "Вильяме", 2004. – 752 с.

30. Microsoft Corporation Разработка Web-сервисов XML и серверных компонентов на Microsoft Visual Basic .NET и Microsoft Visual C# .NET. Учебный курс MCAD/MCSD / пер. с англ. – М. : Издательско-торговый дом "Русская Редакция", 2004. – 576 стр.: ил.

31. Rob Conery, Scott Hanselman, Phil Haack, Scott Guthrie. Professional ASP.NET MVC 1.0. – Indianapolis, Indiana: Wiley Publishing, Inc., 2009. – 196 pages.

### **11.3. Інформаційні ресурси**

32. Довідкове керівництво по ASP.NET [Електронний ресурс]. – Режим доступу : <http://msdn.microsoft.com/ru-ru/asp.net/default.aspx>.



33. Довідкове керівництво по MySQL [Електронний ресурс]. – Режим доступу : [www.mysql.ru/docs/man/](http://www.mysql.ru/docs/man/).
34. Довідник по Web-мовам [Електронний ресурс]. – Режим доступу : [www.spravkaweb.ru/](http://www.spravkaweb.ru/).
35. Підручник PHP [Електронний ресурс]. – Режим доступу : [php.rus-phpnuke.com/](http://php.rus-phpnuke.com/).
36. Підручник по PHP 4 [Електронний ресурс]. – Режим доступу : [www.softtime.ru/bookphp/gl1\\_1.php](http://www.softtime.ru/bookphp/gl1_1.php).
37. Сайт ASP.NET. – Режим доступу : <http://www.asp.net/>.
38. Сайт компанії Microsoft. – Режим доступу : [www.microsoft.com](http://www.microsoft.com).
39. Форум PHP програмістів. – Режим доступу : [php.ru](http://php.ru).
40. HTML довідник. – Режим доступу : [html.manual.ru/](http://html.manual.ru/).
41. Internet інститут інформаційних технологій. – Режим доступу : [www.intuit.ru](http://www.intuit.ru).
42. PHP: Hypertext Preprocessor. – Режим доступу : [www.php.net](http://www.php.net).

## Зміст

Вступ.....	3
1. Опис навчальної дисципліни.....	4
2. Мета та завдання навчальної дисципліни.....	5
3. Програма навчальної дисципліни.....	7
4. Структура навчальної дисципліни.....	11
5. Теми лабораторних занять.....	14
6. Самостійна робота.....	17
6.1. Індивідуальне науково-дослідне завдання.....	23
6.2. Тематика індивідуального завдання.....	24
6.3. Контрольні запитання для самодіагностики.....	26
7. Індивідуально-консультативна робота.....	33
8. Методи навчання.....	33
9. Методи контролю.....	36
10. Розподіл балів, які отримують студенти.....	46
11. Рекомендована література.....	50
11.1. Основна.....	50
11.2. Додаткова.....	51
11.3. Інформаційні ресурси.....	52

НАВЧАЛЬНЕ ВИДАННЯ

**Робоча програма**  
**навчальної дисципліни**  
**"ВЕБ-ТЕХНОЛОГІЇ ТА ВЕБ-ДИЗАЙН"**  
**для студентів напряму підготовки**  
**6.050101 "Комп'ютерні науки"**  
**всіх форм навчання**

Укладачі: **Огурцов Віталій В'ячеславович**  
**Гриньов Денис Валерійович**  
**Щербаков Олександр Всеволодович**

Відповідальний за випуск **Пономаренко В. С.**

Редактор **Бутенко В. О.**

Коректор **Бутенко В. О.**

План 2014 р. Поз. № 118 ЕВ. Обсяг 54 стор.

---

Видавець і виготівник – видавництво ХНЕУ ім. С. Кузнеця, 61166, м. Харків, пр. Леніна, 9а

*Свідоцтво про внесення до Державного реєстру суб'єктів видавничої справи*  
**Дк № 481 від 13.06.2001 р.**

