

	<p><b>«ЗАТВЕРДЖУЮ»</b> Голова вченої ради факультету _____ _____/_____ Протокол № <u>5</u> «<u>17</u>» <u>лютого</u> <u>2020</u></p>
--	--

**СИЛАБУС**  
навчальної дисципліни  
**«Об`єктно-орієнтоване програмування(друга мова)»**  
Шифр за ОПІ – ВІПА4

Освітній рівень -	бакалаврський
Галузь знань -	12 – інформаційні технології
Спеціальність -	126 – інформаційні системи та технології
Освітня програма -	«Web-технології, Web-дизайн»

Силабус навчальної дисципліни «Об'єктно-орієнтоване програмування»  
(*назва навчальної дисципліни*)

підготовки здобувачів освітнього ступеня «бакалавр» за спеціальністю 126 – Інформаційні системи та технології, освітня програма «Web-технології , Web-дизайн» - 11 стор.

Силабус складений на основі програми навчальної дисципліни «Об'єктно-орієнтоване програмування (друга мова)», шифр (за ОПП) – ВППА4.

Розробник силабусу:

Рудницький Сергій Володимирович, к.т.н., старший викладач кафедри ІТП  
(*ПІБ, наук.ст., вчене зв., посада НПП кафедри, що розробив силабус*)

Силабус затверджений на засіданні кафедри інформаційних технологій проектування

Протокол № 8 від «10» січня 2020 року

Обговорено та рекомендовано до затвердження методичною комісією факультету інформаційних технологій і систем

«14» лютого 2020 р., протокол № 4

Голова методичної комісії факультету інформаційних технологій і систем \_\_\_\_\_ /А.Р. Карапетян/  
*підпис* *ПІБ*

### **1. ІНФОРМАЦІЯ ПРО ВИКЛАДАЧА**

Прізвище, ім'я, по батькові	Рудницький Сергій Володимирович
Науковий ступінь	к.т.н.
Наукове звання	-
Посада	старший викладач
Місце роботи	Черкаський державний технологічний університет
Адреса кафедри	18006, м. Черкаси, бул. Шевченка 460, каб. 603-1 корпус

Контактний телефон	(0472)51-15-86
Профайл викладача	<a href="https://chdtu.edu.ua/fitis/kitp/staff/item/1171-rudnytskyi-serhii-volodymyrovych">https://chdtu.edu.ua/fitis/kitp/staff/item/1171-rudnytskyi-serhii-volodymyrovych</a>
e-mail:	s.v.rudnitskiy@gmail.com
Профайл дисципліни	<a href="http://fitis.moodle.chdtu.edu.ua/course/view.php?id=562">http://fitis.moodle.chdtu.edu.ua/course/view.php?id=562</a>
Розклад консультацій	

## 2. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Галузь знань, спеціальність, освітня програма, освітній рівень	Загальні характеристики		Навчальне навантаження з дисципліни	
			денна форма навчання	заочна форма навчання
<u>Галузь знань</u> 12 – інформаційні технології	Вибіркова		Курс підготовки:	
			3-й	
<u>Спеціальність</u> 126 – інформаційні системи та технології	Загальна кількість кредитів ЄКТС	4	Семестр підготовки:	
	Загальна кількість годин	120	6-й	
<u>Освітня програма</u> «Web-технології, Web- дизайн»	Кількість аудиторних годин	54	Лекції	
	Кількість годин самостійної роботи	66	18 год.	
			Практичні, семінарські	
<u>Освітній рівень</u> бакалаврський	Мова навчання - українська		Лабораторні	
			36 год.	
			Самостійна робота	
			66 год	
			Форма підсумкового контролю	
			Залік	

## 3. МЕТА І ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

<b>Мета викладання дисципліни</b>	отримання студентами базових знань і практичних навиків основ сучасної технології розробки і реалізації складних і програмних продуктів. Отримані знання можуть бути використані в розробці прикладних мобільних програм.
<b>Завдання вивчення дисципліни</b>	підготовка фахівців, що володіють сучасними методами і засобами розробки алгоритмів та програм, що знають сучасну технологію програмування і

	уміючи застосовувати її при рішенні складних прикладних задач.
--	--

#### 4. РЕЗУЛЬТАТИ НАВЧАННЯ

№ з/п	Результати навчання
1	Здатність здійснювати розробку алгоритмів і програм з використанням об'єктно-орієнтованого підходу.
2	Здатність будувати та зображати різними способами функціональні та структурні схеми програмного забезпечення.
3	Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.
4	Проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші)

#### 5. ПРЕРЕКВІЗИТИ

«Алгоритмізація та програмування».

#### 6. ПОСТРЕКВІЗИТИ

«Виробнича практика».

#### 7. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Змістовий модуль №1	
<i>Програмування мовою JavaScript. Основи.</i>	
<b>Тема 1</b> <i>Введення в JavaScript.</i>	
<i>1.1. Введення в JavaScript. 1.2. Історія створення. 1.3. Довідники та специфікації. 1.4. Редактори кода. 1.5. Консоль розробника.</i>	
<b>Тема 2</b> <i>Основи JavaScript.</i>	
<i>2.1. Структура кода. 2.2. Use strict. 2.3. Змінні. 2.4. Типи даних. Перетворення типів.</i>	
<b>Тема 3</b> <i>Оператори.</i>	
<i>3.1. Арифметичні оператори. 3.2. Оператор ділення по модулю. Інкремент і декремент. 3.3. Бітові оператори. Оператори порівняння. 3.4. Взаємодія: alert, prompt, confirm. Умовні оператори: if, '?. 3.5. Логічні оператори. 3.6. Цикли while та for. 3.7. Конструкція "switch"..</i>	
<b>Тема 4</b> <i>Функції (основи). Обробка помилок.</i>	
<i>4.1. Функції. Function Expression. 4.2. Функції-стрілки. 4.3. Обробка помилок, "try..catch". 4.4. Користувацькі помилки, розширення Error.</i>	
<b>Тема 5</b> <i>Об'єкти (основи).</i>	

5.1. Об'єкти. Прибирання сміття. 5.2. Тип даних <i>Symbol</i> . 5.3. Методи об'єкта. 5.4. Ключове слово <i>this</i> . 5.5. Перетворення об'єктів в примитиви. 5.6. Конструктори. Ключове слово <i>new</i> .
<b>Тема 6</b> Типи даних.
6.1. Методи примитивів. 6.2. Числа. 6.3. Строки. 6.4. Масиви. Методи масивів. 6.5. Ітеруємі об'єкти. <i>Set</i> , <i>Map</i> , <i>WeakMap</i> і <i>WeakSet</i> . <i>Object.keys</i> , <i>values</i> , <i>entries</i> . 6.6. Час та дата.
<b>Тема 7</b> Розширена робота з функціями.
7.1. Рекурсія та стек. 7.2. Остаточні параметри та оператор розширення. 7.3. Замкнення. 7.4. Глобальний об'єкт. Об'єкт функції <i>NFE</i> . Синтаксис " <i>new Function</i> ". 7.5. Планування: <i>setTimeout</i> and <i>setInterval</i> . 7.6. Декоратори та переадресація виклику, <i>call/apply</i> . 7.7. Прив'язка контекста до функції.
<b>Змістовий модуль №2</b> Об'єктно-орієнтоване програмування мовою <i>JavaScript</i> .
<b>Тема 8</b> Властивості об'єктів та їх конфігурування.
8.1. Флаги та дескриптори сайтів. 8.2. Властивості - герери та сетери.
<b>Тема 9</b> Прототипи та наслідування.
9.1. Прототипне наслідування. 9.2. <i>F.prototype</i> . 9.3. Вбудовані прототипи. 9.4. Методи прототипів, об'єкти без властивості <code>__proto__</code> . 9.5. Домішки.
<b>Тема 10</b> Класи.
10.1. Базовий синтаксис класа. Наслідування класів 10.2. Статичні властивості та методи. 10.3. Приватні та захищені властивості та методи. 10.4. Розширення вбудованих класів. 10.5. Перевірка класа: " <i>instanceof</i> ".
<b>Тема 11</b> Модулі. Генератори. Продвинута терація.
11.1. Генератори. 11.2. Асинхронні ітератори та генератори. 11.3. Експорт та імпорт. 11.4. Динамічні імпорти..

## 8. ТЕМАТИЧНИЙ ПЛАН НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ теми	Назва модулів і тем	Форми організації навчання, кількість годин						Література, інформаційні ресурси
		Денна форма			Заочна форма			
		Лекції	Практичні, лабораторні роботи	Самостійна робота	Лекції	Практичні, лабораторні роботи	Самостійна робота	
Змістовий модуль №1. Основи об'єктно-орієнтоване програмування мовою <i>Java</i> .								
1	<b>Тема 1.</b> Введення в <i>JavaScript</i> .	1		8	-	-	-	1, 2
2	<b>Тема 2.</b> Основи <i>JavaScript</i> .	1	2	-	-	-	-	6, 8, 9
3	<b>Тема 3.</b> Оператори.	2	4	6	-	-	-	3, 4
4	<b>Тема 4.</b> Функції (основи). Обробка помилок.	2	2	-	-	-	-	9, 10, 11
5	<b>Тема 5.</b> Об'єкти (основи).	2	4	8	-	-	-	5, 9, 14
6	<b>Тема 6.</b> Типи даних.	1	4	-	-	-	-	6, 15
7	<b>Тема 7.</b> Розширена робота з функціями.	1	4	8	-	-	-	3, 7, 11

Змістовий модуль №2. <i>Пакели об'єктно-орієнтоване програмування мовою Java.</i>								
8	<b>Тема 8.</b> Властивості об'єктів та їх конфігурування.	2	4	8	-	-	-	1, 4, 9
9	<b>Тема 9.</b> Прототипи та наслідування.	2	4	8	-	-	-	5, 6
10	<b>Тема 10.</b> Класи.	2	4	8	-	-	-	15
11	<b>Тема 11.</b> Модулі. Генератори. Продвинута терація.	2	4	12	-	-	-	11, 12
	<b>Разом</b>	18	36	66	-	-	-	

## 9. ПРАКТИЧНІ / СЕМІНАРСЬКІ ЗАНЯТТЯ, ЛАБОРАТОРНІ РОБОТИ

№ з/п	Назва лабораторної роботи	Кількість годин	
		Денна	Заочна
1	Технологія виконання простого додатка в середовищі JDK	2	-
2	Поняття об'єкта	4	-
3	Динамічні об'єкти	2	-
4	Відношення між об'єктами	4	-
5	Спадкування	4	-
6	Форми спадкування	4	-
7	Методологія об'єктно-орієнтованого аналізу	4	-
8	Методологія об'єктно-орієнтованого проектування	4	-
9	Технології об'єктно-орієнтованого проектування	4	-
10	Експлуатація і супровід об'єктноорієнтованих систем	4	-

### МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

## 10. САМОСТІЙНА РОБОТА

Поглиблене опрацювання розглянутих на лекціях та розгляд суміжних тем.

№ з/п	Назва теми	Кількість годин	
		Денна	Заочна
1	Інкапсуляція	8	-
2	Введення-виведення, цикли, масиви.	6	-
3	Порівняння спадкування реалізації та композиції.	8	-
4	Поліморфізм. Інтерфейси.	8	-
5	Багатопотокове програмування	8	-
6	Послідовні колекції	8	-
7	Серілізація.	8	-
8	Самостійне опрацювання лекційного матеріалу	12	-

Разом	66	-
-------	----	---

## 11. СИСТЕМА ОЦІНЮВАННЯ НАВЧАЛЬНИХ ДОСЯГНЕНЬ

### 11.1 МЕТОДИ КОНТРОЛЮ

В організації навчального процесу застосовуються контрольні заходи у формі вхідного, поточного, модульного, рейтингового і підсумкового контролю.

Вхідний контроль проводиться перед вивченням нового курсу з метою визначення рівня підготовки здобувачів вищої освіти з дисциплін, які забезпечують цей курс. За результатами вхідного контролю розробляються заходи з надання індивідуальної допомоги здобувачам вищої освіти, коригування навчального процесу з відповідного курсу.

Поточний контроль здійснюється під час проведення лекцій та лабораторних занять і має на меті перевірку рівня підготовленості здобувача вищої освіти до виконання конкретних видів навчальної діяльності.

Модульний контроль успішності здобувачів вищої освіти здійснюється для перевірки рівня засвоєння навчального матеріалу в кінці кожного навчального модуля.

Рейтинговий контроль є інструментом комплексного оцінювання якості навчальної роботи здобувача вищої освіти з усіх кредитних модулів на певному етапі навчання. Рейтинговий контроль успішності здобувачів вищої освіти проводиться на 8-9 навчальних тижнях.

Семестровий контроль з дисципліни проводиться відповідно до навчального плану у вигляді заліку в терміни, встановлені графіком навчального процесу, та в обсязі навчального матеріалу, визначеному робочою програмою дисципліни.

Залік – це вид підсумкового контролю, за якого засвоєння здобувачем вищої освіти навчального матеріалу з дисципліни оцінюється на підставі результатів поточного, проміжного контролів (тестування, поточного опитування, виконання індивідуальних завдань та певних видів робіт на лабораторних заняттях) протягом семестру і модульного контролю.

Іспити - це підсумковий етап вивчення усієї дисципліни з метою перевірки знань студентів по теорії і виявлення навичок застосування отриманих знань при вирішенні практичних завдань, а також навиків самостійної роботи з навчальною і науковою літературою.

Іспит дає можливість кожному студенту у порівняно короткий проміжок часу осмислити весь пройдений курс у цілому, сконцентрувати увагу на вузлових його моментах, закріпити у пам'яті його основний зміст.

Оцінка навчальних досягнень здобувачів вищої освіти за всіма видами контролю здійснюється за національною системою та ECTS:

#### Шкала оцінювання: національна та ECTS

Сума балів за всі види	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену,	для заліку

навчальної діяльності		курсowego проекту (роботи), практики	
90-100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
64-73	D	задовільно	
60-63	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

## 11.2 ПИТАННЯ ДО ЗАЛІКУ

1. У чому полягає суть тегу SCRIPT?
2. Описати сучасну розмітку для тегу SCRIPT.
3. Що таке зовнішні скрипти?
4. У чому полягає суть асинхронних скриптів: defer/async?
5. Для чого потрібні побітові оператори?
6. Які існують види перетворення типів для примітивів?
7. Подати синтаксис альтернативного оголошення та виклику функції «Function Expression».
8. Що таке «контекст виконання» (execution context) для виклику функції?
9. Що може стек контекстів у js?
10. Описати іменованій функціональний вираз в js.
11. Що отримуватиме функція, якщо її запускати в контексті різних об'єктів?
12. Як потрібно викликати функцію, щоб this передався?
13. Який метод дозволяє перетворити рядок на масив, розділивши його за розділювачем s?
14. Назвати ще один синтаксис для створення масиву, окрім квадратних дужок [ ].
15. Чи може метод splice вставляти елементи без видалення? Якщо так, то що для цього потрібно?
16. Що виконує метод slice(begin, end)?
17. Чи однаковий синтаксис методу slice для рядків і для масивів?
18. Що буде результатом функції, якщо ця функція не вирішить повернути свій об'єкт?
19. Що поверне виклик return з об'єктом? Що поверне виклик return із чим завгодно, крім об'єкта?



20. Чи можна в JS створювати для зручності єдиний метод у контексті ООП, який називається так само, як властивість, і відповідає і за запис, і за читання?

21. Чи можна властивості, записані в this, уважати публічними?

22. Чи має клас-наслідувач доступ до приватних властивостей батьківського класу?

23. Які властивості класу називаються захищеними?

24. Чи дозволяють сучасні браузері викликати підпроцеси Web Workers, які виконуються паралельно та можуть відправляти/приймати повідомлення, але не мають доступу до DOM?

25. Як називається найглибший елемент, який викликає подія, що є доступний як event.target?

26. Які браузерні події є синхронними?

### 11.3 КРИТЕРІЇ ОЦІНЮВАННЯ

#### ДЕННА ФОРМА

Для студентів денної форми навчання	
Вид навчальної роботи	Кількість балів максимум
<b><u>Постійна частина</u></b>	
ЗМІСТОВИЙ МОДУЛЬ №1	
«Програмування мовою JavaScript. Основи.» – 60 годин	
Захист лабораторної роботи № 1	5
Захист лабораторної роботи № 2	5
Захист лабораторної роботи № 3	5
Захист лабораторної роботи № 4	5
Захист лабораторної роботи № 5	5
Захист лабораторної роботи № 6	5
Модульна контрольна робота № 1	5
<i>Всього за змістовим модулем № 1</i>	30
ЗМІСТОВИЙ МОДУЛЬ №2	
«Об'єктно-орієнтоване програмування мовою JavaScript» – 60 годин	
Захист лабораторної роботи № 6	5
Захист лабораторної роботи № 7	5
Захист лабораторної роботи № 8	5
Захист лабораторної роботи № 9	5
Захист лабораторної роботи № 10	5
Модульна контрольна робота № 2	10
<i>Всього за змістовим модулем № 2</i>	30
<b><u>Додаткова частина</u></b>	
Підготовка та захист реферату за індивідуальною темою	20
Участь у Днях студентської науки	20
Участь у науковій конференції чи семінарі за темою дисципліни	20
Оформлення наочного стенда за індивідуальною темою	20
<b><u>Штрафна частина</u></b>	
Пропуск одного заняття без поважної причини	-5
Несвоєчасний захист звіту з лабораторної роботи	-5

<b>ІСПИТ</b>	<b>30</b>
<b>ПІДСУМКОВА СЕМЕСТРОВА ОЦІНКА</b>	<b>100</b>

## **12. РЕКОМЕНДОВАНА ЛІТЕРАТУРА**

### **Основна**

1. Арнольд, К., Гослинг, Дж., Холмс, Д. Язык программирования JavaScript. - 3-е изд. — М. : Вильямс, 2001. — 624 с.
2. Блинов, И. Н., Романчик, В. С. JavaScript. Промышленное программирование. - Минск : УниверсалПресс, 2007. — 704 с.
3. Перри, Б. У. JavaScript сервлеты и JSP. Сборник рецептов. — М. : Кудиц-пресс, 2009. — 768 с.
4. Васильев А. Н. JavaScript. Объектно-ориентированное программирование: Учебное пособие. - СПб.: Питер, 2011. — 400 с.
5. Хорстманн, К. С., Корнелл, Г. Библиотека профессионала. JavaScript 2 : Том 1. Основы. — 8-е изд. — М. : Вильямс, 2013. — 816 с.
6. Хорстманн, К. С., Корнелл, Г. Библиотека профессионала. JavaScript 2. : Том 2. Тонкости программирования. — 8-е изд. — М. : Вильямс, 2012. — 992 с.
7. Герберт Шилдт. JavaScript 8. Полное руководство; 9-е изд.: Пер. с англ. - М. : ООО "И.Д. Вильямс", 2015. - 1376 с.
8. Патрик Нимейер. Программирование на JavaScript / Патрик Нимейер, Дэниел Леук : [пер. с англ. М.А.Райтмана]. – Москва : Эксмо, 2014 – 1216 с.
9. Nicholas C. Zakas The Principles of Object-Oriented JavaScript - San Francisco: No starch press, 2014. - 122с
10. Eric Elliott. Programming JavaScript Applications - O'Reilly Media, Inc., 2014 - 253с.
11. Хэррон Дэвид. Node.js Разработка серверных веб-приложений на javascript. – ДМК Пресс, 2012 – С.-146.
12. Джамеа Крис. Эффективный самоучитель по креативному Web-дизайну. HTML, XHTML, CSS, JavaScript, PHP, ASP, ActiveX. Текст, графика, звук и анимация. Пер с англ./Крис Джамса, Конрад Кинг, Энди Андерсон - М.: ООО «ДиаСофтЮП», 2005.- 672 с.
13. Флэнаган Д. JavaScript. Подробное руководство.– Пер. с англ.– СПб: СимволП2004.– 960 с.
14. Роббинс Дженнифер. HTML5, CSS3 и JavaScript. Исчерпывающее руководство / Дженнифер Роббинс. – 4-е издание. – Эксмо, 2014. – 516 с.
15. Бранденбау Д. JavaScript: сборник рецептов / Д. Бранденбау - СПб.: Питер, - 2000, - 416 с.

## **13. ІНФОРМАЦІЙНІ РЕСУРСИ**

1. <http://www.JavaScript2s.com/>
2. <https://JavaScriptrush.ru>
3. <http://www.progs.in.ua/>
4. <https://metanit.com/JavaScript/tutorial/1.1.php>
5. <https://learn.javascript.ru/>

## **14. ПЕРЕЛІК НОРМАТИВНИХ ДОКУМЕНТІВ**

1. Положення про організацію контролю та оцінювання якості навчання студентів (<https://chdtu.edu.ua/normative/regulations/item/420-polozhennya-pro-organizatsiyu-kontrolyu-ta-otsinyuvannya-yakosti-navchannya-studentiv>).

2. Положення про організацію освітнього процесу в Черкаському державному технологічному університеті (<https://chdtu.edu.ua/normative/regulations/item/3636-polozhennya-pro-orhanizatsiyu-osvitnoho-protsesu-v-cherkaskomu-derzhavnomu-tekhnologichnomu-universyteti>).

3. Кодекс академічної доброчесності Черкаського державного технологічного університету (<https://chdtu.edu.ua/normative/regulations/item/8892-kodeks-akademichnoyi-dobrochesnosti-cherkaskoho-derzhavnoho-tekhnologichnoho-universytetu-zimamy>).

## 15. ПОЛІТИКА ДИСЦИПЛІНИ

Для успішного вивчення дисципліни та проходження контрольних заходів здобувачі вищої освіти зобов'язані:

- не запізнюватися на заняття;
- не пропускати заняття (у разі хвороби надати довідку або її ксерокопію);
- своєчасно і самостійно виконувати всі передбачені програмою завдання до лабораторних робіт;
- брати очну участь у контрольних заходах;
- оволодіти навчальним матеріалом для самостійного вивчення з дисципліни у вільний від обов'язкових занять час;
- підтримувати зворотній зв'язок з викладачем на всіх етапах вивчення дисципліни;
- дотримуватися академічної доброчесності.