

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА
ПРИРОДОКОРИСТУВАННЯ**

Навчально-науковий інститут кібернетики, інформаційних технологій та інженерії

04-01-149S

СИЛАБУС	Мова програмування Python (практикум)	
SYLLABUS	Python programming language (workshop)	
Шифр за ОП Code in Degree Programme	ВВ 39	
Освітній рівень Level of Education	Бакалаврський (перший) Bachelor's (first)	
Галузь знань Field of Knowledge	12	Інформаційні технології Information Technology
Спеціальність Field of Study	122	Комп'ютерні науки Computer Science
Освітня програма Degree Programme	Комп'ютерні науки Computer Science	

Силабус навчальної дисципліни «Мова програмування Python (практикум)» для здобувачів вищої освіти ступеня «бакалавр», які навчаються за освітньо-професійною програмою «Комп'ютерні науки», спеціальність 122 «Комп'ютерні науки». Рівне. НУВГП. 2025. 15 стор.

ОП на сайті університету: <https://ep3.nuwm.edu.ua/23461/>.

Розробник силабусу: *Климюк Юрій Євгенійович*, кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук та прикладної математики

Силабус схвалений на засіданні кафедри комп'ютерних наук та прикладної математики
Протокол № 8 від "03" січня 2025 року

Завідувач кафедри: *Турбал Юрій Васильович*, доктор технічних наук, професор

Керівник (гарант) освітньої програми: *Каштан Сергій Степанович*, кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук та прикладної математики

Схвалено науково-методичною радою з якості ННІ кібернетики, інформаційних технологій та інженерії
Протокол № 3 від "06" січня 2025 року


Голова науково-методичної ради з якості ННІ: *Мартинюк Петро Миколайович*, доктор технічних наук, професор, директор ННІ кібернетики, інформаційних технологій та інженерії

© НУВГП, 2025

ПРОГРАМА ОСВІТНЬОГО КОМПОНЕНТА	
Навчальна дисципліна «Мова програмування Python (практикум)»	
ЗАГАЛЬНА ІНФОРМАЦІЯ	
Ступінь вищої освіти	<i>Бакалавр</i>
Освітня програма	<i>Комп'ютерні науки</i>
Спеціальність	<i>122 Комп'ютерні науки</i>

Рік навчання, семестр	2-й рік, 4-й семестр
Кількість кредитів	6
Лекції:	0 год. / 0 год.
Практичні заняття:	60 год. / 18 год.
Самостійна робота:	120 год. / 162 год.
Курсова робота:	-
Форма навчання	денна / заочна
Форма підсумкового контролю	залік
Мова викладання	Державна

ІНФОРМАЦІЯ ПРО РОЗРОБНИКА

Лектор	Климюк Юрій Євгенійович, кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук та прикладної математики
	
Вікіситет	https://wiki.nuwm.edu.ua/index.php/Климюк_Юрій_Євгенійович
ORCID	https://orcid.org/0000-0003-3672-8469
Як комунікувати	yu.ye.klymiuk@nuwm.edu.ua

ІНФОРМАЦІЯ ПРО ОСВІТНІЙ КОМПОНЕНТ

Мета та завдання

Освітній компонент «Мова програмування Python (практикум)» призначений для ознайомлення із сучасними технологіями розробки додатків для вирішення прикладних задач в галузі комп'ютерних наук мовою програмування Python з використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка (CLI) і редактора Visual Studio Code.

Метою освітнього компонента є формування у студентів базових професійних знань, практичних умінь та навичок стосовно розробки додатків для вирішення прикладних задач в галузі комп'ютерних наук мовою програмування Python. Основними завданнями є: ознайомлення із сучасними технологіями розробки додатків мовою програмування Python, оволодіння сучасними навичками по розробці додатків мовою програмування Python з використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.

Посилання на розміщення освітнього компонента на навчальній платформі Moodle, на платформі освітніх програм та їхніх освітніх компонентів

<https://exam.nuwm.edu.ua/course/view.php?id=5264>

<https://nuwm.edu.ua/nni-akot/kaf-pm/osvitni-proghrami/item/kompiuterni-nauky>

**Передумови вивчення*
(місце освітнього компонента в структурно-логічній схемі)**

Освітні компоненти, які передують вивченню даного освітнього компонента:

- Вступ до спеціальності «Комп'ютерні науки»
- Програмування
- Операційні системи
- Об'єктно-орієнтоване програмування

Освітні компоненти, які вивчаються одночасно з даним освітнім компонентом:

- Алгоритми та структури даних

Освітні компоненти, вивчення яких спирається на даний освітній компонент:

- Курсова робота з комп'ютерних технологій
- Виробнича практика
- Кваліфікаційна робота

Компетентності

ЗК 1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК 2. Здатність застосовувати знання у практичних ситуаціях.

ЗК 4. Здатність спілкуватися державною мовою як усно, так і письмово.

ЗК 6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК 7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК 8. Здатність генерувати нові ідеї (креативність).

ЗК 9. Здатність працювати в команді.

ЗК 11. Здатність приймати обґрунтовані рішення.

ЗК 12. Здатність оцінювати та забезпечувати якість виконуваних робіт.

ФК 8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

ФК 10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.

Програмні результати навчання (ПРН). Результати навчання (РН)*

ПРН 9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

Структура та зміст освітнього компонента

Лекції	Практичні заняття	Самостійна робота
<u> 0 </u> год. / <u> 0 </u> год.	<u> 60 </u> год. / <u> 18 </u> год.	<u> 120 </u> год. / <u> 162 </u> год.

Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1-4, 14, 15, 18-20]	Опис: Ознайомитися з налаштуванням Visual Studio Code для розробки додатків мовою Python, набути базових практичних умінь та навичок стосовно розробки додатків мовою Python у редакторі Visual Studio Code.
Тема 2. Розробка додатків мовою Python з використанням команд операційної системи.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 2, 6, 14, 15, 18-20]	Опис: Ознайомитися з налаштуванням середовища операційної системи та Visual Studio Code для розробки додатків мовою Python з використанням команд операційної системи інтерфейсу командного рядка (CLI), набути базових практичних умінь та навичок стосовно розробки додатків мовою Python з використанням команд операційної системи інтерфейсу командного рядка у редакторі Visual Studio Code.
Тема 3. Розробка файлів сценаріїв Bash з використанням фрагментів коду мовою Python.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 2, 14, 15, 18-20]	Опис: Ознайомитися з розробкою файлів сценаріїв Bash з використанням фрагментів коду мовою Python, набути базових практичних умінь та навичок стосовно розробки файлів сценаріїв Bash з використанням фрагментів коду мовою Python у редакторі Visual Studio Code.
Тема 4. Форматування коду.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з вимогами згідно з посібником зі стилю PEP8 щодо форматування коду при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно форматування коду згідно з посібником зі стилю PEP8 при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 5. Автоматизація форматування коду.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з автоматизованим форматування коду згідно з посібником зі стилю PEP8 при розробці додатків мовою Python за допомогою команд інтерфейсу командного рядка Black, набути базових практичних умінь та навичок стосовно автоматизації форматування коду згідно з посібником зі стилю PEP8 при розробці додатків мовою Python за допомогою команд інтерфейсу командного рядка Black у редакторі Visual Studio Code.
Тема 6. Автоматизація редагування та форматування коду.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15, 18]	Опис: Ознайомитися з автоматизованим форматування коду згідно з посібником зі стилю PEP8 при розробці додатків мовою Python у редакторі Visual Studio Code за допомогою розширень IntelliSense та BlackFormatter, набути базових практичних умінь та навичок стосовно автоматизації редагування та форматування коду згідно з посібником зі стилю PEP8 при розробці додатків мовою Python у редакторі Visual Studio Code за допомогою розширень IntelliSense та BlackFormatter.
Тема 7. Дотримання правил іменування.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з правилами іменування згідно з посібником зі стилю PEP8 при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно дотримання правил іменування (змінних, констант, функцій тощо) згідно з посібником зі стилю PEP8 при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 8. Пошук та усунення недоліків коду.			

Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з пошуком та усуненням недоліків коду при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно пошуку та усунення недоліків коду при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 9. Написання ефективних функцій.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з написанням ефективних функцій при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно написання ефективних функцій при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 10. Використання коментарів та рядків описів.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з використанням коментарів та рядків описів при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно використання коментарів та рядків описів при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 11. Використання підказок типу.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з використанням підказок типів при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно використання підказок типів при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 12. Використання системи керування версіями Git.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 7, 15, 18]	Опис: Ознайомитися з використанням команд інтерфейсу командного рядка системи керування версіями Git при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно використання команд інтерфейсу командного рядка системи керування версіями Git при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 13. Використання розширених можливостей системи керування версіями Git.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 7, 15, 18]	Опис: Ознайомитися з використанням команд інтерфейсу командного рядка розширених можливостей системи керування версіями Git при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно використання команд інтерфейсу командного рядка розширених можливостей системи керування версіями Git при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 14. Використання системи керування версіями Git у редакторі Visual Studio Code.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 7, 15, 18]	Опис: Ознайомитися з використанням системи керування версіями Git при розробці додатків мовою Python безпосередньо у редакторі Visual Studio Code, набути базових практичних умінь та навичок стосовно використання системи керування версіями Git при розробці додатків мовою Python безпосередньо у редакторі Visual Studio Code.
Тема 15. Використання платформи для спільної роботи GitHub.			

Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 7, 15, 18]	Опис: Ознайомитися з використанням платформи для спільної роботи GitHub при розробці додатків мовою Python, набути базових практичних умінь та навичок стосовно використання платформи для спільної роботи GitHub при розробці додатків мовою Python у редакторі Visual Studio Code.
Тема 16. Розробка проєктів додатків.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з етапами розробки проєктів додатків мовою Python, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 17. Розробка проєктів додатків з використанням об'єктно-зорієнтованого підходу.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з етапами розробки проєктів додатків мовою Python з використанням об'єктно-зорієнтованого підходу, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з використанням об'єктно-зорієнтованого підходу, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 18. Розробка багатомодульних проєктів додатків з використанням об'єктно-зорієнтованого підходу.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з етапами розробки багатомодульних проєктів додатків мовою Python з використанням об'єктно-зорієнтованого підходу, набути базових практичних умінь та навичок стосовно розробки багатомодульних проєктів додатків мовою Python з використанням об'єктно-зорієнтованого підходу, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 19. Розробка проєктів додатків з використанням успадкування при об'єктно-зорієнтованому підході.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1. 15]	Опис: Ознайомитися з етапами розробки проєктів додатків мовою Python з використанням успадкування при об'єктно-зорієнтованому підході, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з використанням успадкування при об'єктно-зорієнтованому підході, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 20. Розробка простіших проєктів графічних додатків з використанням модуля tkinter.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [15, 16]	Опис: Ознайомитися з розробкою простіших проєктів графічних додатків мовою Python з використанням модуля tkinter, набути базових практичних умінь та навичок стосовно розробки простіших проєктів графічних додатків мовою Python з використанням модуля tkinter, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 21. Розробка складніших проєктів графічних додатків з використанням модуля tkinter.			

Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [15, 16]	Опис: Ознайомитися з розробкою складніших проєктів графічних додатків мовою Python з використанням модуля tkinter, набути базових практичних умінь та навичок стосовно розробки складніших проєктів графічних додатків мовою Python з використанням модуля tkinter, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 22. Розробка простіших проєктів графічних додатків з використанням модуля PyQt.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [12, 15, 16]	Опис: Ознайомитися з розробкою простіших проєктів графічних додатків мовою Python з використанням модуля PyQt, набути базових практичних умінь та навичок стосовно розробки простіших проєктів графічних додатків мовою Python з використанням модуля PyQt, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 23. Розробка складніших проєктів графічних додатків з використанням модуля PyQt.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [12, 15, 16]	Опис: Ознайомитися з розробкою складніших проєктів графічних додатків мовою Python з використанням модуля PyQt, набути базових практичних умінь та навичок стосовно розробки складніших проєктів графічних додатків мовою Python з використанням модуля PyQt, системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 24. Компіляція проєктів додатків у окремий виконуваний файл за допомогою PyInstaller.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [12, 17]	Опис: Ознайомитися з компіляцією проєктів додатків мовою Python у окремий виконуваний файл (збірку з виконуваним файлом .exe) за допомогою PyInstaller, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з компіляцією проєктів додатків у окремий виконуваний файл за допомогою PyInstaller, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 25. Розробка інсталяційних пакетів проєктів додатків за допомогою InstallForge.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [9, 10, 12]	Опис: Ознайомитися з розробкою інсталяційних пакетів проєктів додатків мовою Python за допомогою InstallForge, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з розробкою інсталяційних пакетів проєктів за допомогою InstallForge, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 26. Розробка проєктів додатків із документуванням та тестуванням коду за допомогою модуля doctest.			

Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [15, 16]	Опис: Ознайомитися з розробкою проєктів додатків мовою Python із документуванням та тестуванням коду за допомогою модуля doctest, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з документуванням та тестуванням коду за допомогою модуля doctest, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 27. Розробка проєктів додатків із тестуванням коду за допомогою модуля pytest.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [11, 13, 15, 16]	Опис: Ознайомитися з розробкою проєктів додатків мовою Python із тестуванням коду за допомогою модуля pytest, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з тестуванням коду за допомогою модуля pytest, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 28. Розробка проєктів додатків із тестуванням коду за допомогою фікстур модуля pytest.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [11, 13, 15, 16]	Опис: Ознайомитися з розробкою проєктів додатків мовою Python із тестуванням коду за допомогою фікстур модуля pytest, набути базових практичних умінь та навичок стосовно розробки проєктів додатків мовою Python з тестуванням коду за допомогою фікстур модуля pytest, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 29. Розробка проєктів додатків із тестуванням коду на основі властивостей за допомогою модулів pytest і Hypothesis.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [5, 8, 15, 16]	Опис: Ознайомитися з розробкою проєктів додатків мовою Python із тестуванням коду на основі властивостей за допомогою модулів pytest і Hypothesis, набути навиків по розробці проєктів додатків мовою Python із тестуванням коду на основі властивостей за допомогою модулів pytest і Hypothesis, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.
Тема 30. Розробка проєктів додатків для надсилання електронних листів.			
Результати навчання: ПРН 9	Кількість годин: практичні – 2 год.	Література: [1, 2, 15, 16]	Опис: Ознайомитися з розробкою проєктів додатків мовою Python для надсилання електронних листів за допомогою модулів smtplib і yagmail, приховування введення паролів за допомогою модуля getpass, набути практичних умінь та навичок стосовно розробки проєктів додатків мовою Python для надсилання електронних листів за допомогою модулів smtplib і yagmail, приховування введення паролів за допомогою модуля getpass, використанням системи керування версіями Git та платформи для спільної роботи GitHub за допомогою команд інтерфейсу командного рядка і редактора Visual Studio Code.

Форми та методи навчання

Демонстрація, навчальна дискусія, дебати, презентації, міні-лекції, ситуаційні дослідження, робота в малих групах та інше.

Інструменти, обладнання, програмне забезпечення

Комп'ютерний клас, навчальна платформа Moodle, Google Meet, Python, Visual Studio Code, Git, InstallForge.

Порядок оцінювання програмних результатів навчання/ результатів навчання

Оцінювання знань студентів відбувається згідно положення «Система оцінювання результатів навчання здобувачів вищої освіти (семестровий поточний та підсумковий контроль) зі змінами та доповненнями» (2021 р.) <http://ep3.nuwm.edu.ua/21123/>, яке передбачає перевірку знань студентів під час захисту практичних робіт та проведення проміжного контролю у вигляді тестування у навчальній системі Moodle.

Розподіл балів:

Змістовий модуль № 1

Практичні роботи – 30 балів

Модульний контроль – 20 балів

Змістовий модуль № 2

Практичні роботи – 30 бали

Модульний контроль – 20 балів

Всього: 100 балів

Таблиця формування білету тестового завдання
проміжного модульного контролю № 1

Рівень складності	Загальна кількість завдань в базі	Кількість завдань в білеті	Макс. оцінка за вірно виконане завдання (балів)	
			за одне	загальна
1	181	15	0,8	12
2	72	4	1,5	6
3	53	1	2	2
	306	20		20

Таблиця формування білету тестового завдання
проміжного модульного контролю № 2

Рівень складності	Загальна кількість завдань в базі	Кількість завдань в білеті	Макс. оцінка за вірно виконане завдання (балів)	
			за одне	загальна
1	178	15	0,8	12
2	68	4	1,5	6
3	56	1	2	2
	302	20		20

Додаткові бали (при умові, що загальна сума поточного оцінювання не перевищує 60 балів) студенти можуть отримати за виконання спеціальних завдань, що узгоджуються з викладачем (не більше, ніж 10 балів), зокрема, за підготовку тез на наукову конференцію за тематикою навчальної дисципліни; за участь з доповіддю на конференції; за наукову статтю, за участь в олімпіадах чи конкурсах.

Загальна оцінка курсу розраховується як арифметична сума набраних балів (не більше, ніж 100) за всі види навчальних та додаткових завдань.

Шкала загальної оцінки курсу

<i>Сума балів за всі види навчальної діяльності</i>	<i>Оцінка за національною шкалою для екзамену</i>
90–100	відмінно
82–89	добре
74–81	
64–73	задовільно
60–63	
0–59	незадовільно

Рекомендована література (основна, допоміжна)

Основна:

1. Al Sweigart. Beyond the Basic Stuff with Python: Best Practices for Writing Clean Code. No Starch Press, 2020. 384 p. Режим доступу: <https://inventwithpython.com/beyond/>.
2. Al Sweigart. Automate the Boring Stuff with Python. 3rd Edition. No Starch Press, 2025. 672 p. Режим доступу: <https://automatetheboringstuff.com/>.
3. Download the latest version for Windows. Python 3.13.1. Режим доступу: <https://www.python.org/downloads/>.
4. Download Visual Studio Code. Режим доступу: <https://code.visualstudio.com/download>.
5. Rodrigo Girão Serrão, Dan Ackerson. Getting Started With Property-Based Testing in Python With Hypothesis and Pytest. Режим доступу: <https://semaphoreci.com/blog/property-based-testing-python-hypothesis-pytest>.
6. Git. Downloads. Режим доступу: <https://git-scm.com/downloads>.
7. Git. Documentation. Режим доступу: <https://git-scm.com/doc>.
8. How to Use Hypothesis and Pytest for Robust Property-Based Testing in Python. Режим доступу: <https://pytest-with-eric.com/pytest-advanced/hypothesis-testing-python/>.
9. InstallForge Documentation. Режим доступу: <https://docs.installforge.net/>.
10. InstallForge. Free Download. Режим доступу: <https://installforge.net/download/>.
11. Kevin Ndung'u Gathuku. Testing Python Applications with Pytest. Режим доступу: <https://semaphoreci.com/community/tutorials/testing-python-applications-with-pytest>.
12. Packaging PyQt6 applications for Windows with PyInstaller & InstallForge. Режим доступу: <https://www.pythonguis.com/tutorials/packaging-pyqt6-applications-windows-pyinstaller/>.
13. Pytest. Documentation. Режим доступу: <https://docs.pytest.org/en/stable/#documentation>.

14. Python Tutorial: Learn Python Programming. Режим доступу: <https://www.tutorialspoint.com/python/index.htm>.
15. Python 3.13.1 documentation. Режим доступу: <https://docs.python.org/3.13/>
16. Real Python Tutorials. Режим доступу: <https://realpython.com/>.
17. Using PyInstaller. Режим доступу: <https://pyinstaller.org/en/stable/usage.html/>.
18. Visual Studio Code documentation. Режим доступу: <https://code.visualstudio.com/docs>.
19. Васильєв Олексій. Програмування мовою Python. Богдан, 2019. 540 с.
20. Зеленьяк Олег. Програмування мовою Python. Алгоритмічні структури і стратегії. Теорія та практика. Ліра-К, 2025. 338 с.

Допоміжна:

21. Hillard Dane. Practices of the Python Pro. Shelter Island : Manning, 2020. 250 p.
22. Lott F. Steven, Phillips Dusty. Python Object-Oriented Programming: Build robust and maintainable object-oriented Python applications and libraries (Fourth Edition). BIRMINGHAM–MUMBAI, 2021. 715 p.
23. Brian Okken. Python Testing with pytest. Second Edition. Simple, Rapid, Effective, and Scalable. Raleigh, North Carolina: The Pragmatic Bookshelf, 2022. 250 p.

Інформаційні ресурси в Інтернет

1. Національна бібліотека ім. В. І. Вернадського. Електронний ресурс: <http://www.nbuv.gov.ua/>
2. Обласна наукова бібліотека (м. Рівне, майдан Короленка, 6). Електронний ресурс]: <http://www.lib.rv.ua/>
3. Наукова бібліотека НУВГП (м. Рівне, вул. Олекси Новака, 75) / Електронний ресурс: <https://lib.nuwm.edu.ua/>

Поєднання навчання та досліджень* (за потреби)

Студенти можуть додатково виконувати індивідуальні завдання у вигляді досліджень; бути долученими до написання та опублікування наукових статей; приймати участь у науково-практичних конференціях, наукових конкурсах.

Здобувачі вищої освіти можуть долучатися до виконання кафедральних науково-дослідних тем, а також тем, що фінансуються з державного бюджету.

ПОЛІТИКИ ВИКЛАДАННЯ ТА НАВЧАННЯ

Перелік соціальних, «м'яких» навичок (soft skills)

Здатність до самонавчання, саморозвитку вдосконалення комунікаційних якостей та знань з іноземної мови, застосовування набутих знань у практичних ситуаціях. Здатність працювати в команді, проявляти лідерські здібності, логічно обґрунтовувати свою позицію та прийняті рішення. Здатність виявляти, ставити та вирішувати проблеми. Здатність генерувати нові ідеї.

Дедлайни та перескладання

Ліквідація академічної заборгованості здійснюється згідно «Порядку ліквідації академічних заборгованостей у НУВГП» <https://ep3.nuwm.edu.ua/25072/>. Студент має право на повторне вивчення дисципліни чи повторне навчання на курсі.

Дата проведення модульних контролів відображається у календарі сторінки дисципліни на платформі Moodle. Перездача модульних контролів, пропущених з поважних причин, здійснюється згідно графіку, розміщеному навчально-науковим центром незалежного оцінювання (ННЦНО) на головній сторінці системи Moodle.

Неформальна та інформальна освіта (за потреби)

Студенти мають право на перезарахування результатів навчання, набутих у неформальній та інформальній освіті, згідно «Положення про неформальну та інформальну освіту НУВГП» <http://ep3.nuwm.edu.ua/18660/>. Також студенти можуть самостійно на платформах Prometheus, Coursera, edEx, edEra, FutureLearn та інших опановувати матеріал для перезарахування результатів навчання. При цьому важливо, щоб знання та навички, що формуються під час проходження певного онлайнкурсу чи його частин, мали зв'язок з очікуваними навчальними результатами даної освітньої компоненти / освітньої програми та перевірялись в підсумковому оцінюванні.

Правила академічної доброчесності

Студент зобов'язаний дотримуватися «Кодексу честі студентів» <http://ep3.nuwm.edu.ua/4917/>, у свою чергу, викладач – «Етичного кодексу викладача НУВГП» <http://ep3.nuwm.edu.ua/4916/>.

Дотримання академічної доброчесності регламентується «Положення про академічну доброчесність» <https://ep3.nuwm.edu.ua/25004/>, «Положенням про виявлення та запобігання академічного плагіату в Національному університеті водного господарства та природокористування» (нова редакція) <http://ep3.nuwm.edu.ua/10325/>.

Додаткова інформація розміщена на головній сторінці НУВГП за посиланням Якість освіти ⇒ Академічна доброчесність <http://nuwm.edu.ua/sp/akademichna-dobrochesnistj>.

Вимоги до відвідування

Студенту не дозволяється пропускати заняття без поважних причин. Студенти можуть використовувати без обмежень на заняттях мобільні телефони, ноутбуки та за потреби інші додаткові прилади. Під час дистанційного навчання заняття проводяться онлайн з використанням додатку Google Meet згідно розкладу.

Студент має право оформити індивідуальний графік навчання згідно «Положення про індивідуальний графік навчання студентів денної форми навчання Національного університету водного господарства та природокористування» <http://ep3.nuwm.edu.ua/6226/>.

У разі пропуску занять студент самостійно опрацьовує теоретичний матеріал, розміщений у навчальній системі Moodle, і виконує практичні роботи. При потребі студент може звернутися за консультацією до викладача відповідно до графіку консультацій або за допомогою корпоративної електронної пошти.

Автор
Доцент

Юрій КЛИМЮК

Затверджено

Проректор з науково-педагогічної та
навчальної роботи

Валерій СОРОКА



документ підписаний КЕП
Номер документа СИЛ №728
Підписувач Сорока Валерій Степанович
Підписувач (дані КЕП):
Сертифікат 3FAA9288358EC003040000009B6C3700C8C2C100