

Київський національний університет
будівництва і архітектури
Кафедра інформаційних технологій

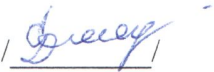
Шифр спеціальності і	Назва спеціальності, освітньої програми	Освітній рівень
122	Комп'ютерні науки, Інформаційні управляючі системи і технології	Бакалавр

«Затверджую»

Завідувачка кафедри
Тетяна ГОНЧАРЕНКО



Розробник силабуса
Сергій ДОЛГОПОЛОВ



СИЛАБУС

OK12 «Програмування мовою Python»

(назва, шифр освітньої компоненти (дисципліни))

1) Статус освітньої компоненти: обов'язкова	
2) Контактні дані викладача: асистент кафедри ІТ, Долгополов Сергій Юрійович, dolhopolov_sy@knuba.edu.ua, +380663035568, https://www.knuba.edu.ua/elementor-192361/ .	
3) Пререквізити (дисципліни-попередники, які необхідно вивчити, щоб слухати цей курс): Програмування та алгоритмічні мови; Офісні інформаційні технології; Інструментальні засоби програмування.	
4) Коротка анотація освітньої компоненти: Освітня компонента «Програмування мовою Python» забезпечує комплексне вивчення розробки програмного забезпечення з використанням екосистеми Python. Курс охоплює фундаментальні концепції мови, включаючи особливості типізації, управління пам'яттю, роботу з генераторами, ітераторами та декораторами, а також принципи ООП з акцентом на метакласи та паттерни проєктування. Програма включає роботу з даними через ORM (SQLAlchemy, Django ORM), обробку даних з використанням pandas та numpy, веб-розробку на Flask та Django. Здобувачі освоюють асинхронне програмування з asyncio, паралельне програмування з threading та multiprocessing, основи машинного навчання з scikit-learn. Особлива увага приділяється практикам розробки: написання чистого коду згідно PEP 8, модульне тестування з pytest, CI/CD з використанням Docker та GitHub Actions. Курс формує практичні навички розробки масштабованих Python-додатків з використанням сучасних інструментів та методологій.	
5) Структура курсу: лекції, лабораторні роботи, самостійні роботи, розрахунково-графічна робота, залік.	
Загальна кількість кредитів ECTS	3,0
Сума годин:	90

Вид індивідуального завдання	РГР
Форма контролю	Залік
<p>б) Зміст курсу:</p> <p><i>Змістовий модуль 1. Основи програмування на Python.</i></p> <p><i>Лекція 1.</i> Вступ до програмування на Python.</p> <p><i>Лабораторна робота 1.</i> Встановлення середовища розробки та написання першої програми.</p> <p><i>Лекція 2.</i> Функції та структури даних.</p> <p><i>Лабораторна робота 2.</i> Реалізація умовних операторів та циклів.</p> <p><i>Лекція 3.</i> Об'єктно-орієнтоване програмування (ООП) в Python.</p> <p><i>Лабораторна робота 3.</i> Створення та використання функцій, реалізація рекурсії.</p> <p><i>Лекція 4.</i> Модульне програмування та організація коду.</p> <p><i>Лабораторна робота 4.</i> Операції зі структурами даних.</p> <p><i>Лабораторна робота 5.</i> Створення класів та об'єктів, реалізація ООП (<i>Командний проєкт</i>).</p> <p><i>Змістовий модуль 2. Робота з даними та веб-розробка.</i></p> <p><i>Лекція 1.</i> Основи роботи з даними та файлами.</p> <p><i>Лабораторна робота 6.</i> Використання та створення модулів та пакетів.</p> <p><i>Лекція 2.</i> Бази даних та веб-розробка.</p> <p><i>Лабораторна робота 7.</i> Читання та запис даних у файли.</p> <p><i>Лекція 3.</i> Тестування, налагодження та оптимізація коду.</p> <p><i>Лабораторна робота 8.</i> Підключення до бази даних та виконання основних операцій.</p> <p><i>Лабораторна робота 9.</i> Створення простого веб-додатку на Flask/Django.</p> <p><i>Лабораторна робота 10.</i> Розробка інтерактивних сторінок на Flask/Django (<i>Командний проєкт</i>).</p> <p><i>Змістовий модуль 3. Розширені можливості та практичне застосування Python.</i></p> <p><i>Лекція 1.</i> Візуалізація та машинне навчання в Python.</p> <p><i>Лабораторна робота 11.</i> Використання pandas для обробки табличних даних.</p> <p><i>Лекція 2.</i> Розширені можливості та безпека Python-додатків.</p> <p><i>Лабораторна робота 12.</i> Візуалізація даних з matplotlib та seaborn.</p> <p><i>Лекція 3.</i> DevOps та CI/CD з Python.</p> <p><i>Лабораторна робота 13.</i> Написання тестів з використанням unittest або pytest.</p> <p><i>Лабораторна робота 14.</i> Розробка початкової версії продукту (<i>Командний проєкт</i>).</p> <p><i>Лабораторна робота 15.</i> Удосконалення та презентація додатку (<i>Командний проєкт</i>).</p> <p>Самостійна робота:</p> <ol style="list-style-type: none"> 1. Аналіз різних парадигм програмування (імперативне, функціональне, об'єктно-орієнтоване, логічне) в контексті Python та їх порівняльна ефективність при вирішенні різних типів задач. 2. Поглиблене вивчення внутрішньої реалізації структур даних Python (списки, словники, множини, кортежі) та методів оптимізації роботи з ними через генератори та ітератори. 	

3. Розширені концепції ООП в Python: метакласи, паттерни проектування (Factory, Singleton, Observer, Strategy) та принципи SOLID в розробці Python-систем.
4. Сучасні підходи до роботи з базами даних через ORM-системи (SQLAlchemy, Django ORM), включаючи оптимізацію запитів, управління транзакціями та міграції.
5. Архітектурні патерни веб-розробки (MVC, MTV) та їх реалізація у Flask і Django, включаючи системи маршрутизації, аутентифікації та авторизації.
6. Методи оптимізації обробки великих об'ємів даних з використанням pandas та numpy, включаючи векторизацію операцій та паралельну обробку.
7. Асинхронне програмування та багатопоточність в Python через asyncio, threading та concurrent.futures, з акцентом на синхронізацію та уникнення race conditions.
8. Основи машинного навчання на Python: алгоритми класифікації, регресії та кластеризації з використанням scikit-learn та TensorFlow.
9. Сучасні практики DevOps для Python-проектів: автоматизація через CI/CD, контейнеризація з Docker та Kubernetes, моніторинг та логування.

Індивідуальне завдання:

Розрахунково-графічна робота (РГР) спрямована на практичне застосування набутих знань з програмування на Python. На виконання РГР відводиться 12 годин самостійної роботи.

Тематика РГР включає розробку прикладних програмних систем різного призначення: управління даними, аналіз інформації, автоматизація процесів, створення чат-ботів, системи моніторингу, веб-додатки тощо. Здобувачі можуть обрати тему з запропонованого переліку або запропонувати власну за погодженням з викладачем.

Приклади можливих тем для виконання РГР:

1. Додаток для управління особистими фінансами.
2. Система рекомендацій для книжкового онлайн-магазину.
3. Автоматизована система управління домашніми задачам.
4. Додаток для аналізу даних соціальних мереж.
5. Чат-бот для підтримки клієнтів.

7) Посилання на сторінку електронного навчально-методичного комплексу освітньої компоненти: <https://org2.knuba.edu.ua/course/view.php?id=3621>.