

УДК 004.41:37.09

Дмитро Трейтяк

доктор філософії (PhD)

Інститут проблем виховання НАПН України, м. Київ, Україна

ORCID ID 0000-0003-3909-5062

stevandthree@gmail.com

МОВА ПРОГРАМУВАННЯ PYTHON ЯК ІНСТРУМЕНТ У НАУКОВО-ПЕДАГОГІЧНІЙ ДІЯЛЬНОСТІ

Анотація. У статті розглянуто можливості застосування мови програмування Python у науково-педагогічній діяльності. Основна мета дослідження – виявити ефективні способи використання Python для автоматизації освітніх і наукових процесів, аналізу великих даних, розробки інтерактивних навчальних матеріалів. Методи дослідження передбачають аналіз наукової літератури, вивчення закордонного досвіду використання Python у навчальному процесі та наукових дослідженнях, а також емпіричне дослідження впровадження Python у педагогічну діяльність.

У дослідженні окреслено основні переваги Python для освітян і дослідників, зокрема: автоматизація обчислювальних процесів та аналізу даних; використання бібліотек для візуалізації та обробки інформації (Matplotlib, Pandas); створення інтерактивних навчальних матеріалів і цифрових інструментів для наукових досліджень (Flask); організація та проведення масових заходів за допомогою Python.

Стаття вирізняється своєю інноваційністю, оскільки не має аналогів у вітчизняних наукових дослідженнях та відкриває новий напрям у педагогічній діяльності. Зокрема в ній досліджуються перспективи використання мови програмування Python як інструменту для підвищення ефективності освітнього процесу, акцентовано увагу на її потенціалі для вчителів і викладачів у гуманітарних галузях. У роботі розкриваються можливості Python для інтеграції цифрових технологій у навчання, реалізації інтерактивних методів викладання, що сприяє підвищенню якості освіти та розвитку цифрової компетентності педагогів. У статті описано приклади застосування Python у викладанні, підготовці дослідницьких проєктів і роботі з великими обсягами інформації. Зокрема наведені приклади використання закордонними педагогами Python для полегшення їх професійної діяльності. Результати дослідження підкреслюють актуальність використання Python для підвищення ефективності навчально-наукової діяльності а також демонструють, у яких аспектах використання Python буде найбільш корисними для педагогів і дослідників.

Практичне значення даного дослідження полягає в тому, що його матеріали можуть бути використані вчителями, викладачами, студентами ЗВО, аспірантами під час проведення наукової або педагогічної роботи.

Ключові слова: Python; науково-педагогічна діяльність; автоматизація; освіта; цифрова компетентність.

1. ВСТУП

Постановка проблеми. У сучасному світі, де технології розвиваються надзвичайно швидко, вміння користуватися програмними засобами є важливою складовою успішної роботи викладача, науковця та студента. Початкові труднощі опанування програмних засобів, які можуть виникнути у представників нетехнічних спеціальностей, не є перешкодою для успішного опанування базових знань. Навпаки, такі знання відкривають широкі можливості для підвищення ефективності майбутньої професійної діяльності. Набуття цих навичок дозволяє розширити професійні компетенції та спростити виконання багатьох завдань.

Дане дослідження присвячене вивченню інноваційних можливостей мови програмування Python у контексті науково-педагогічної діяльності. Ми прагнемо

продемонструвати, як Python може стати каталізатором нових підходів та методів у цій сфері. Також у статті ми дослідимо, як Python може сприяти полегшенню наукових досліджень та вдосконаленню педагогічних практик. Python – це мова програмування, яка відзначається простотою синтаксису, доступністю та універсальністю. Python має низку переваг у педагогічній діяльності порівняно з іншими мовами програмування, такими як C++, Java чи JavaScript. Його синтаксис є максимально наближеним до природної мови, що знижує поріг входу для початківців і дозволяє швидше засвоювати основи програмування. На відміну від C++ або Java, де необхідно оперувати складними конструкціями, такими як оголошення типів змінних чи керування пам'яттю, у Python більшість операцій реалізується значно простіше, що робить його ідеальним для гуманітаріїв і викладачів без глибоких технічних знань.

Крім того, варто зазначити, що на відміну від JavaScript, який використовується переважно для веброзробки, Python має універсальне застосування: від аналізу даних до автоматизації процесів у наукових дослідженнях. Завдяки цим перевагам Python стає незамінним інструментом у сучасній педагогічній діяльності.

Новизна дослідження полягає в комплексному підході до застосування Python у педагогіці: поєднання теоретичних напрацювань із практичними реалізаціями, що дозволяють інтегрувати цю мову в навчальний процес на різних рівнях освіти. Наукова значущість обраної проблематики зумовлена глобальною цифровізацією освіти та зростаючою потребою у використанні гнучких та потужних інструментів для аналізу даних, автоматизації процесів і створення інтерактивних навчальних матеріалів.

Аналіз досліджень. Різноманітним аспектам програмування присвячували свої роботи такі дослідники, як О. Спірін, Т. Вакалюк, Н. Дегтярьова, С. Петренко, О. Удовиченко, К. Bati [1], [2]. До питання вивчення основ програмування мовою Python у ЗВО України зверталася низка науковців: А. Анісімов, Г. Козуб, Н. Семенов, А. Костюченко, С. Матвійчук, С. Погорілий, А. Яковенко [3].

S. Nicolajsen, S. Nielsen та L. Carlsen у своєму дослідженні розглянули підтримку викладачів у впровадженні програмування, оцінили та надали огляд освітніх програм Данії з обов'язковим програмуванням за кількома вимірами. За результатами дослідження науковці виділили п'ять рекомендацій для політиків і педагогів, відповідальних за інтегрування програмування в їхню освіту [4].

Дослідники А. Samala, S. Rawas, S. Criollo у своїй статті проаналізували, як новітні технології, зокрема штучний інтелект, блокчейн, невзаємозамінні токени, доповнена реальність, віртуальна реальність, змішана реальність, робототехніка та 3D-друк суттєво вплинули на освіту. У їх дослідженні було розглянуто експериментальні дослідження з бази даних Scopus за період 2008-2022 років за допомогою інструментів візуалізації даних, зокрема RStudio, VOSviewer, Python і Microsoft Excel. Дослідження визначило прогалини в застосуванні технологій, що виникають через економічний статус, інфраструктуру та проблеми з цифровою грамотністю. Крім того, у дослідженні було розкрито потенціал технологій адаптивного навчання та персоналізованого навчального середовища для трансформації освіти шляхом пристосування досвіду до індивідуальних потреб [5].

F. Stajano у своїй статті «Python in Education: Raising a Generation of Native Speakers» піднімає питання про те, чи варто більш широко навчати програмуванню людей не тільки із професій, дотичних до IT-сфери, але й лікарів, істориків, педагогів, музикантів. Автор відзначає легкість мови для новачків та у висновках підсумовує, що Python все ще недостатньо використовується і вона може бути чудовим ресурсом для широкого навчання програмуванню [6].

До інших сфер використання мови Python звертались у своєму дослідженні науковці V. Dolgopolas, V. Dagienė, S. Minkevičius, L. Sakalauskas [7].

Дослідниця Т. Georgieva-Trifonova у своїй статті вивчала міжпредметні зв'язки Python на прикладі проєктів з навчальних предметів – математики, історії, географії, людина і природа. Автор розглянула проєкти, що могли зацікавити учнів середніх класів за рівнем складності і водночас закріплювали міжпредметні зв'язки, допомагаючи інтегрувати використання Python у процес вивчення звичних предметів у школі [8].

У той же час використання мови Python у науково-педагогічній діяльності є відносно новим напрямом наукового пошуку та має чимало раніше не вирішених частин проблематики, які не досліджувались у світовій та українській науці.

Мета дослідження. Метою даного дослідження є виявлення, обґрунтування та розробка ефективних методів використання мови програмування Python у педагогічній діяльності для підвищення ефективності освітнього процесу, інтеграції цифрових технологій у навчання та автоматизації наукових досліджень. У роботі розглядаються теоретичні аспекти застосування Python у навчальному середовищі, аналізуються міжнародні практики впровадження цієї мови у викладання, а також розробляються практичні інструменти для освітньої діяльності.

2. МЕТОДОЛОГІЯ ТА МЕТОДИ ДОСЛІДЖЕННЯ

Методологія дослідження базується на міждисциплінарному підході, що поєднує педагогічну теорію, інформаційні технології та аналіз даних. У дослідженні використано такі методи:

Аналіз закордонного досвіду – проведено огляд наукових публікацій та освітніх платформ, які використовують Python у навчальному процесі. Це дозволило визначити найкращі практики та розкрити можливості їх адаптації до вітчизняного освітнього контексту.

Теоретичне моделювання – розроблено концептуальні підходи до інтеграції Python у навчальні дисципліни, зокрема у сфері гуманітарних наук, де програмування традиційно має обмежене застосування.

Емпіричне дослідження – створено та протестовано програмні рішення на основі Python, що можуть бути використані в педагогічній діяльності. Реалізовано скрипти для автоматизації обробки освітніх даних, генерації дидактичних матеріалів та аналізу інформації, що дозволило оцінити ефективність запропонованих методик.

Практична частина дослідження містить розробку кодів, які демонструють ефективність Python у педагогічному процесі. Було створено кілька скриптів, які надали можливість значно спростити роботу співробітникам Інституту та автоматизувати організаційні процеси.

Результати дослідження доводять, що Python є ефективним інструментом для розвитку цифрової компетентності педагогів, зокрема їхньої здатності використовувати інформаційно-комунікаційні технології (ІКТ) для розробки інтерактивних освітніх ресурсів, аналізу значних масивів даних та організації заходів. Це сприяє оптимізації навчального процесу та впровадженню інноваційних освітніх технологій.

3. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Мова програмування Python має кілька беззаперечних переваг як зручна та легка у вивченні, вона має багатоцільову спрямованість, призначена для розв'язування широкого спектру задач. Варто зазначити, що програми, створені на Python, можна запускати на різних операційних системах, ще однією перевагою мови є те, що програмний код легко читається, він має мінімальну кількість розділових знаків. Крім

того, у мові наявна строга динамічна типізація даних (при оголошенні змінної її тип не вказується), що також полегшує роботу. У структурі Python існує велика кількість бібліотек та внутрішніх функцій, що також є її перевагою [3].

Саме тому Python більшістю викладачів визначається як найкращий вибір для навчання студентів програмуванню завдяки своїй читабельності, універсальності та широкому використанню в галузі. Її зрозумілий і нескладний синтаксис мінімізує перешкоди для початківців програмістів. Студенти можуть зосередитись на вивченні основних концепцій програмування, не загрузаючи в складних правилах синтаксису. Універсальність Python дозволяє викладачам розробляти захоплюючі проєкти в різних областях, від розробки ігор і аналізу даних до веброзробки [9].

Подібною думки дотримується і J. C. García Monsálvez, стверджуючи у своєму дослідженні, що Python, завдяки своїм особливостям, є ідеальною мовою програмування. Він зазначає, що деякі особливості мови Python (легкість вивчення, інтерпретація, обширна бібліотека та сторонні модулі, безкоштовне програмне забезпечення, безліч відкритих освітніх ресурсів) дають можливість її застосування в навчальних програмах. Дослідник стверджує, що необхідно переосмислити ті цілі, які нині стоять перед предметом інформатики. У якості пропозицій для майбутніх досліджень він пропонує методологію для розробки відкритих освітніх ресурсів. Іншою сферою діяльності може бути створення інструментів для допомоги у навчанні програмуванню, таких як аналіз коду, створеного студентом [10].

Крім того, разом із простотою Python здатний автоматизувати різні завдання, що є одним із факторів його успіху як однієї з провідних і найбільш популярних мов програмування сьогодні. Python також стала популярною у сфері комп'ютерної освіти в цілому, не тільки для спеціальностей інформатики, через відсутність структурування даних [11].

Розглянемо детальніше, як Python можна використати в науковій діяльності. Ця мова програмування надає змогу обробляти, систематизувати, аналізувати та візуалізувати дані. Завдяки бібліотекам «NumPy», «Pandas» та «SciPy» Python дає змогу працювати з великими наборами даних, проводити статистичний аналіз і моделювання. Графіки, діаграми та інтерактивні візуалізації створюються за допомогою бібліотек «Matplotlib», «Seaborn», та «Plotly», що особливо важливо для подання результатів досліджень [6].

Розглянемо декілька напрямків найбільш ефективного використання Python саме в науково-педагогічній діяльності:

1. Автоматизація рутинних завдань. Це одна з найкорисніших та найефективніших сфер застосування Python. Вона передбачає обробку великих обсягів студентських робіт, підготовку навчальних матеріалів або створення тестів. Також може бути надзвичайно зручною при обробці великих масивів даних із таблиць, систематизації, перетворенні їх у зручний формат для подальшого дослідження або презентації.
2. Створення інтерактивних навчальних матеріалів, які можуть у зручній та красивій формі продемонструвати необхідний матеріал. Звісно, нині сервіси створення графіки, такі як Canva, у поєднанні з системами штучного інтелекту здатні допомогти у ряді візуальних рішень, проте за допомогою мови Python учитель або викладач може створити програму з потрібним саме для його методики алгоритмом. Модуль «python-pptx» дозволяє автоматично створювати PowerPoint-презентації. Наприклад, можна згенерувати слайди зі статистикою, таблицями чи графіками для виступу. Python також має зручні інструменти для побудови графіків, гістограм чи теплових карт, у цьому стануть у пригоді бібліотеки «Matplotlib», «Seaborn», «Plotly». Інтерактивні графіки дозволяють

зручніше представити результати досліджень на конференціях чи в публікаціях. Яскравим прикладом інтерактивного засобу навчання на основі Python є Jupyter Notebook, який дозволяє створювати документи, що поєднують код, текст, графіки та візуалізації, і є ідеальним інструментом для навчання програмуванню, аналізу даних, математики тощо. Іншим успішним проектом є Khan Academy, яка використовує Python для створення інтерактивних вправ з математики, хімії та біології.

3. Помічник у навчанні. Python ідеально підходить для викладання основ програмування завдяки її інтуїтивності, а також може стати у пригоді при вирішенні логічних, математичних завдань. Окрім цього, реальне застосування Python можливе при аналізі даних студентських опитувань для оцінки якості викладання, створенні симуляцій фізичних явищ на уроках природничих дисциплін, автоматизації створення графіків успішності студентів.
4. Форматування текстів. Python дозволяє легко обробляти текстову інформацію завдяки різноманітним інструментам. Так за допомогою регулярних виразів (модуль «re») можна швидко знаходити та виправляти потрібні фрагменти в тексті, перевіряти дані на відповідність до створеного формату, переформатовувати текст. Використовуючи шаблонні бібліотеки, такі як «Jinja2», можна створювати автоматизовані звіти або листи, додаючи персоналізовані дані (імена, дати тощо). Корисними у роботі з документами будуть бібліотеки та модулі по роботі з файлами формату docx, які дозволяють змінювати стиль тексту, додавати таблиці, заголовки та інше. Python дозволяє автоматизувати рутинну роботу з текстовими документами та електронними таблицями. За допомогою бібліотеки «python-docx» можна додавати чи редагувати текст, стилізувати заголовки, вставляти таблиці, графіки або зображення, готувати документи для конференцій, наприклад, створювати сертифікати або розклади. Деякі бібліотеки Python дозволяють аналізувати текстові масиви, наприклад, для вивчення тенденцій у наукових статтях [12].
5. Бібліотеки «openpyxl», «pandas», «xlswriter», «NumPy», «SciPy» дозволяють працювати з Excel і можуть допомогти при статистичному аналізі та виконанні складних математичних обчислень. З їх допомогою можна автоматизувати заповнення таблиць даними, провести аналіз великих наборів даних: підсумки, сортування, фільтрацію; генерувати звіти про наукові дослідження з графіками та підсумковими таблицями. Наприклад, створити зведену таблицю для відображення результатів опитувань студентів із графічним представленням даних. Python є незамінним інструментом для роботи з даними.
6. Підготовка до проведення масових заходів. Python може значно спростити підготовку до наукових заходів. За допомогою бібліотеки «smtplib» або платформи «yagmail» можна надсилати персоналізовані запрошення учасникам конференції. Крім того, програму можна налаштувати так, аби вона прикріплювала потрібний файл, інший для кожного адресата, що значно полегшить та прискорить роботу, особливо при значній кількості учасників заходу. Python допоможе також при роботі з реєстраційними даними учасників. Використовуючи бібліотеку «pandas», можна аналізувати список учасників, перевіряти дублікати чи формувати списки для друку. Ще однією зручною функцією стане можливість створення PDF-сертифікатів для учасників конференції та їх розсилка електронною поштою [6].

Підсумовуючи, можна навести конкретний приклад, коли в процесі підготовки до конференції необхідно виконати наступні задачі: зібрати дані про учасників з Excel-файлу, згенерувати сертифікати у форматі PDF, надіслати їх електронною поштою та

побудувати графіки статистики учасників для презентації. Для ілюстрації такого прикладу було створено простий код, який допомагав зібрати дані із таблиці реєстрації учасників заходу до файлу Word для швидкого створення програми заходу. Подібний код уже успішно використовувався співробітниками Інституту проблем виховання НАПН України під час підготовки до планової конференції

```
import openpyxl
from docx import Document

def generate_word_from_excel(excel_file, columns_sequence, output_word_file):
    # Відкриваємо Excel файл
    workbook = openpyxl.load_workbook(excel_file)
    sheet = workbook.active

    # Створюємо новий Word документ
    doc = Document()

    # Проходимо по всіх рядках у Excel
    for row in sheet.iter_rows(min_row=2, values_only=True):
        row_data = [str(row[ord(col.upper()) - ord('A')]) if
row[ord(col.upper()) - ord('A')] is not None else '' for col in columns_sequence]
        # Додаємо рядок до Word документа
        doc.add_paragraph(' '.join(row_data))

    # Зберігаємо Word документ
    doc.save(output_word_file)

# Задаємо файли
excel_file = 'example.xlsx' # Шлях до вашого Excel файлу
columns_sequence = ['C', 'D', 'A', 'B'] # Послідовність колонок
output_word_file = 'output.docx' # Шлях до вихідного Word файлу

generate_word_from_excel(excel_file, columns_sequence, output_word_file)
```

Варто зауважити, що Python також може стати незамінним помічником при організації науково-педагогічних заходів. Збільшення масштабу події (кількості учасників) призводить до експоненціального зростання складності організаційних процесів, що, своєю чергою, створює необхідність у впровадженні автоматизованих систем управління для оптимізації ресурсів, підвищення ефективності та зниження ризику виникнення помилок. При організації таких заходів складність не лінійно зростає з кількістю учасників, а збільшується в геометричній прогресії. Це пов'язано з необхідністю координувати все більшу кількість взаємопов'язаних процесів. Автоматизовані системи дозволяють ефективно обробляти великі обсяги даних, виконувати рутинні завдання, гарантувати швидку реакцію на зміни та мінімізувати людський фактор, забезпечуючи високу якість організації конференції. Також автоматизація мінімізує ризик виникнення помилок, пов'язаних з людським фактором, таких як помилки введення даних, пропуски інформації тощо [13].

Отже, автоматизація організаційних процесів конференції є необхідним кроком для забезпечення ефективного управління великими заходами, оскільки вона дозволяє впоратись зі зростаючою складністю та забезпечити високий рівень якості організації.

Системи штучного інтелекту (ШІ) стали потужним інструментом для новачків у програмуванні, особливо тих, хто опановує Python. Використовуючи такі платформи, як інтерактивні помічники чи спеціалізовані IDE з вбудованими ШІ, новачки можуть значно прискорити своє навчання. ШІ допомагає зрозуміти складні концепції, пропонуючи зрозумілі пояснення коду, синтаксису або логіки програмування. Також можна швидко знайти помилки у коді завдяки автоматичному аналізу та отримати

рекомендації щодо виправлення. Окрім того новачки можуть ставити питання або отримувати готові приклади коду, які можна адаптувати під власні потреби, що спрощує створення перших програм. Наприклад, якщо новачок хоче написати простий калькулятор або вебскрейпер, система ШІ може допомогти розібратися зі структурами даних, модулями чи бібліотеками, такими як «math» або «BeautifulSoup». Впровадження ШІ сприяє зниженню рівня стресу та підвищенню залученості в освітній процес шляхом надання користувачеві можливості експериментального дослідження та опанування нових концепцій в індивідуалізованому темпі з одночасним отриманням оперативного зворотного зв'язку.

Розглянемо можливості використання Python для створення вчителем власного сайту з інтерактивними завданнями, опитуваннями та тестами для учнів. Розпишемо основні кроки.

1. Встановлення середовища розробки. До початку роботи вчитель повинен встановити Python та необхідні бібліотеки. Це можна зробити за допомогою таких інструментів, як `pip` (Python Package Installer). Зручним середовищем може бути PyCharm або Microsoft Visual Studio Code, які містять чимало вбудованих функцій, що полегшують роботу початківцям. Після цього створюють робочу папку, у якій організовують структуру папок та файлів сайту (Рис. 1).

```

flask_quiz_app/
|
├── app.py          # Основний файл для запуску застосунку
├── templates/      # Шаблони сторінок HTML
|   ├── base.html  # Базовий шаблон
|   ├── index.html # Головна сторінка
|   └── quiz.html   # Сторінка з тестом
├── static/         # Статичні файли (CSS, зображення)
|   └── styles.css  # Стили сайту
└── quizzes/        # Файли з тестами (наприклад, JSON)
    └── sample_quiz.json

```

Рис.1. Схема організації файлів при розробці сайту

2. Розробка базового сайту з використанням Flask. Flask – це мікрофреймворк, який дозволяє швидко створювати прості та функціональні вебдодатки. Для більш складного сайту можна використати фреймворк Django, проте в нашому прикладі розглянемо саме Flask. Цей код створює простий сайт із сторінкою для проведення опитувань або тестів, також програма рахує кількість правильних відповідей:

```

from flask import Flask, render_template, request, redirect, url_for, jsonify
import json

app = Flask(__name__)

tests = [
    {
        "title": "Назва тесту",
        "questions": [
            {"question": "Питання_1", "options": ["A", "B", "B"], "correct":
"B"},

```

```

        {"question": "Питання_2", "options": ["A", "B", "B"], "correct":
"A"}
    }
    ]

# Створінка тесту
@app.route('/test/<int:test_id>', methods=['GET', 'POST'])
def take_test(test_id):
    if test_id >= len(tests):
        return "Тест не знайдено", 404

    test = tests[test_id]

    if request.method == 'POST':
        # Обробка результатів
        answers = request.form
        correct_answers = 0
        total_questions = len(test["questions"])

        for i, question in enumerate(test["questions"]):
            user_answer = answers.get(f"q{i}")
            if user_answer == question["correct"]:
                correct_answers += 1

        return render_template('result.html', test=test,
correct=correct_answers, total=total_questions)

    return render_template('test.html', test=test, test_id=test_id)

if __name__ == '__main__':
    app.run(debug=True)

```

3. Використовуючи HTML, CSS та JavaScript, учитель може створювати шаблони сторінок, які будуть використовуватись у додатку. Мова програмування JavaScript дозволяє додавати інтерактивні елементи, проте її використання виходить за межі нашого дослідження, тому не будемо зупинятися на цьому. Наведемо приклад шаблону HTML, мови розмітки, яка використовується для створення структури та вмісту вебсторінок:

```

<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>Головна сторінка</title>
</head>
<body>
    <h1>Виберіть тест</h1>
    <ul>
        {% for test in tests %}
        <li><a href="/test/{{ loop.index0 }}">{{ test.title }}</a></li>
        {% endfor %}
    </ul>
</body>
</html>

<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>{{ test.title }}</title>
</head>
<body>
    <h1>{{ test.title }}</h1>
    <form method="POST">
        {% for question in test.questions %}

```

```

<div>
  <p><b>{{ loop.index }}. {{ question.question }}</b></p>
  {% for option in question.options %}
    <label>
      <input type="radio" name="q{{ loop.parent.index0 }}" value="{{
option }}" required> {{ option }}
    </label><br>
  {% endfor %}
</div>
{% endfor %}
<button type="submit">Завершити</button>
</form>
</body>
</html>

```

Для тестів можна створити форму з питаннями, яку учні заповнюють онлайн. Python оброблятиме введені дані та повертатиме результати учням, підраховуючи різні критерії. Зручність використання Python полягає також у тому, що при створенні такого тесту можна налаштувати зчитування спеціальних показників, таких як час відповіді на кожне питання, що дозволяє проаналізувати, чи користувався учень підказками при проходженні тесту.

4. Інтеграція бази даних. Для збереження та зручного зберігання результатів тестів і прогресу учнів можна підключити базу даних. Наприклад, SQLite є легким і зручним рішенням для невеликих проєктів, за допомогою якого вчитель зможе створити власний функціональний сайт, де він зможе зберігати результати тестів учнів і аналізувати їх. На початковому етапі, якщо вчитель ще не знайомий з використанням баз даних, можна обмежитись використанням Python та створенням звичайного сайту без збереження великих масивів даних. Вчитель може зосередитись на основних концепціях програмування на Python і створенні вебінтерфейсу без необхідності заглиблюватись у складні теми, пов'язані з базами даних. Дані тестів можна зберігати у простих форматах, таких як файли CSV або JSON, що не вимагає використання складних систем управління базами даних [14].

5. Розгортання сайту. Готовий створений сайт учитель може розгорнути на безкоштовних або платних хостингах, таких як Heroku чи PythonAnywhere, що дозволить учням мати доступ до ресурсів у будь-який час. Для розгортання сайту також можна використати GitHub Pages – безкоштовну платформу для розміщення сайтів, проте варто пам'ятати, що GitHub не надає можливості запускати серверні скрипти безпосередньо на GitHub Pages.

Тож, володіючи зовсім невеликими навичками по створенню HTML сторінок та написанню простого коду у Python, можна легко створити власний сайт, адаптований під потреби певного вчителя, який він може сам змінювати, налаштовувати та додавати на нього необхідні матеріали або тести. Використання мови програмування Python надасть можливість автоматизувати збір відповідей учнів, причому отримані результати будуть не тільки кількісні (як це може надати звичайна Google-форма), але й якісні (можна налаштувати відображення додаткових показників, таких як час відповіді на кожне питання чи навіть розширений аналіз: пошук ключових слів у відповідях тощо).

Крім використання мови Python у викладацькій діяльності, значні перспективи має запровадження програмування в навчальний процес. Закордонні педагоги наводять приклади проєктів на Python, які можуть використовуватись для занять безпосередньо з молодшими школярами (інтерактивні історії, прості ігри та основи робототехніки, які дозволять учням побачити відчутні результати їхніх зусиль у кодуванні, сприяючи творчості та критичному мисленню). Для учнів середньої школи Python використовується у більш складних проєктах, таких як веброзробка, візуалізація даних і програми на основі ШІ. Ці проєкти можуть не лише вдосконалювати навички кодування,

вони також готують молодь до міждисциплінарного характеру сучасного ринку робочої сили, де такі навички є цінним надбанням у різних галузях [9].

Розглядаючи приклад закордонних колег, ми можемо побачити, як учитель за допомогою Python поставив собі за мету спростити та автоматизувати оцінювання екзаменаційних робіт великої кількості студентів. Для цього ним було створено програму, яка ставила одне запитання за раз, приймала відповідь як вхідні дані від учня та надсилала результат у вигляді загальних балів, які призначаються учню на основі його відповіді.

Було створено список у Python, який містив усі запитання – «que», та інший список «ans», який містив усі відповіді в тому самому порядку. Якщо відповідь студента на перше запитання в «que» збігається з першою відповіддю в «ans», це правильна відповідь і студент отримує один бал. В іншому випадку відповідь буде неправильною і позначка для цього питання не буде додана. Умовний оператор у Python кодується за допомогою ключових слів «if», «elif» і «else» із належним синтаксисом і відступами. Перевірку умови для кожного набору запитань виконано за допомогою циклу «for». Без нього неможливо було б перевірити кожне питання, особливо коли запитань багато. Цей цикл забезпечує виконання коду від початку до кінця для всіх запитань у «que».

Протестувавши таку систему, педагог вирішив перейти до випадку оцінювання відповідей описового типу. Існує дві різні техніки, які можна застосувати. Перша полягає у використанні ключових слів. Наводиться приклад, коли учень має написати відповідь про певне поняття і в різних викладачів повинні бути свої методики, але для об'єктивної оцінки відповіді наявність певних ключових слів повинна бути для більшості з них слушною. Тобто учень, який згадує всі п'ять ключових слів у своїй відповіді, отримує повний бал, а студент, який згадує три з п'яти ключових слів, отримує три бали з п'яти. Тож кількість ключових слів є важливим критерієм оцінки описових відповідей. Можна написати код, щоб визначити кількість згаданих ключових слів і відповідним чином винагородити відповідь студента.

У Python відповідь, отримана від учня, спочатку перетворюється на нижній регістр, потім розбивається на окремі слова, після чого за допомогою циклу відбувається аналіз на наявність ключових слів у відповіді. Цикл «for» перевіряє кожен елемент у списку «ans», чи вони збігаються з даним списком `key_words`. Для будь-якого слова, яке знаходить збіг, показник `word_count` збільшується на одиницю. Оскільки існує п'ять ключових слів, `word_count` може показувати максимум п'ять. Окрема відповідь, яка призводить до `word_count=5`, оцінюється найвище [15].

Т. Georgieva-Trifonova у своєму дослідженні навела приклад проєктів з математики, історії, географії, під час яких можна використовувати Python, зокрема дослідниця наводить приклади створення коду для відтворення часової шкали подій та для відображення відміток на географічній карті за заданими координатами [8].

Аналіз результатів дослідження, яке проводилось у Туреччині серед учителів природничих дисциплін, дозволяє стверджувати, що інтеграція інформаційно-комунікаційних технологій (ІКТ) у педагогічну практику є ефективним та доступним процесом для вчителів, а, отже, використання мови Python у педагогічній діяльності є цілком перспективним. Дослідження проводилося серед учителів, які попередньо не мали навичок обчислювального мислення чи розв'язування обчислювальних задач. До програми було додано базову інформацію, таку як логіка кодування, алгоритми, використання Google Colab, змінні, цикли та функції на Python. Результати цього дослідження показали суттєве покращення навичок розв'язування комп'ютерних задач у вчителів природничих наук. Отже, результати дослідження підтвердили можливість використання мови Python у педагогічній сфері [2].

Для вчителів, які прагнуть використовувати Python у своїй педагогічній або науковій діяльності є ряд ресурсів, які допоможуть розібратися та розпочати роботу. По-перше – офіційний вебсайт Python (Python.org), який пропонує розширену документацію, навчальні посібники та завантаження для Python. По-друге – платформи Python Software Foundation, edX, Coursera, які пропонують і курси Python для викладачів і студентів. Викладачі також можуть використовувати ці платформи для створення власного контенту курсу. Також корисними в роботі можуть бути Jupyter Notebooks, PythonBooks.org, PyCon, Codecademy, які надають безкоштовні ресурси та плани уроків для викладачів, можуть слугувати довідками та навчальними матеріалами [9].

Отже, використання Python у педагогічній діяльності значно покращує цифрову компетентність педагогів, розширюючи їхні навички роботи з сучасними технологіями. Python дозволяє автоматизувати рутинні процеси, що розвиває вміння працювати з інструментами автоматизації. Використання бібліотек для візуалізації даних покращує вміння створювати та інтерпретувати графіки, діаграми та інші форми візуального представлення інформації. Робота з Python розвиває навички програмування, що є ключовим аспектом цифрової грамотності. Педагоги навчаються писати код, працювати з даними та використовувати алгоритмічне мислення для вирішення задач. Підсумовуючи можна схематично представити основні напрямки використання Python у педагогічній діяльності (Рис. 2).

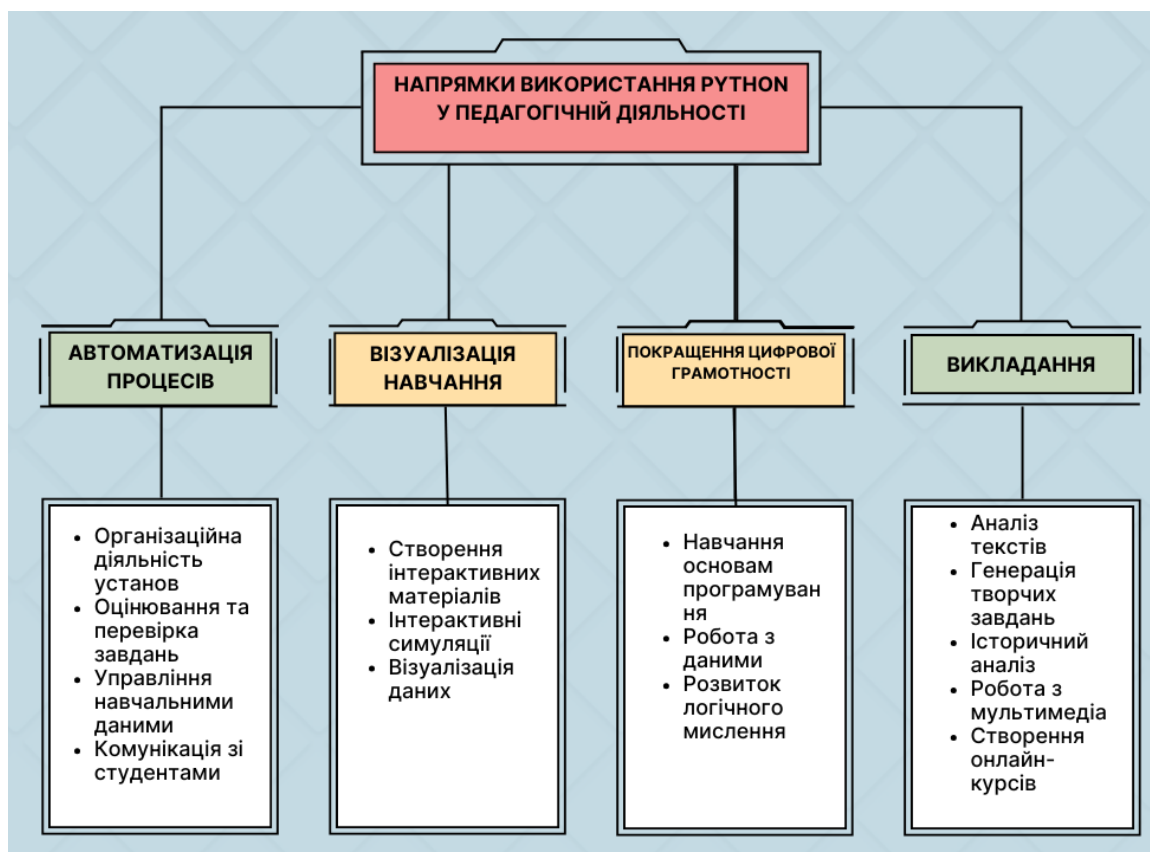


Рис.2. Напрямки використання Python у педагогічній діяльності

4. ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Отже, Python є універсальним інструментом для науково-педагогічної діяльності, що дозволяє автоматизувати освітні процеси, підвищувати точність аналізу даних, створювати інтерактивні навчальні матеріали та оптимізувати виконання завдань. У ході дослідження доведено, що використання Python сприяє підвищенню ефективності

освітньої діяльності та розвитку цифрової компетентності педагогів і студентів. Простий синтаксис, велика кількість бібліотек та доступність навчальних ресурсів роблять цю мову однією з найбільш придатних для впровадження в навчальний процес.

Оригінальність і новизна дослідження полягає у систематизації та аналізі міжнародного досвіду використання Python у педагогіці, а також у розробці практичних підходів до його інтеграції у вітчизняну освітню систему. Проведене дослідження демонструє потенціал використання мови програмування Python як ефективного інструменту для інтеграції цифрових технологій у сфері педагогічних наук. На відміну від існуючих українських досліджень, у статті вперше запропоновано системний підхід до використання Python. Описані практичні приклади застосування Python у викладанні, проведенні масових заходів та обробці великих обсягів інформації підтверджують можливість розширення цифрової компетентності педагогів і студентів. Інноваційність дослідження полягає в розгляді Python як засобу модернізації педагогічних практик та створення умов для підвищення ефективності науково-педагогічної діяльності. Дослідження продемонструвало, що Python може бути ефективно використана для автоматизації рутинних процесів, аналітики навчальних даних, створення інтерактивних платформ та розробки дидактичних матеріалів.

Рекомендації щодо впровадження Python у навчальний процес:

1. Інтегрувати Python у навчальні програми для розвитку алгоритмічного мислення та цифрової грамотності студентів.

2. Використовувати Python для автоматизації аналізу освітніх даних, тестування та оцінювання знань, спрощення організаційних процесів університетів та науково-дослідних установ.

3. Запроваджувати курси підвищення кваліфікації для педагогів, що сприятиме впровадженню нових технологій у навчальний процес.

У подальшому тема використання Python у науково-педагогічній діяльності має широкі перспективи. Python є ефективним інструментом для проведення наукових досліджень, оскільки дозволяє швидко аналізувати великі обсяги даних, будувати математичні моделі та здійснювати симуляції різних процесів. У педагогічній сфері мова програмування може стати основою для створення інтерактивних платформ для опитувань, автоматизованого підрахунку результатів тестування та формування статистичних звітів. Ця мова є універсальним інструментом, який можна адаптувати як для підтримки освітнього процесу, так і для впровадження нових технологій у наукові дослідження, роблячи її важливим елементом сучасної науково-педагогічної діяльності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Т. Вакалюк, В. Болотіна, Є. Байлюк, О. Покотило, "Огляд ігрових онлайн сервісів для вивчення мов програмування", *Інноваційна педагогіка: науковий журнал* №22, с. 192-198, 2020.
- [2] K. Bati, "Integration of Python into Science Teacher Education, Developing Computational Problem Solving and Using Information and Communication Technologies Competencies of Pre-service Science Teachers", *Informatics in Education* 21, no. 2, 235-251, 2022. doi: 10.15388/infedu.2022.12.
- [3] Н. Дегтярьова, С. Петренко, О. Удовиченко, "Робота з графічними віджетами при вивченні мови програмування Python в закладах загальної середньої освіти." *Освіта. Інноватика. Практика.* № 11(4), с. 26-34, 2023. doi: <https://doi.org/10.31110/2616-650X-vol11i4-004>.
- [4] S. Nicolajsen, S. Nielsen, L. Carlsen et al. "Programming education across disciplines: a nationwide study of Danish higher education." *High Educ*, 2024. doi:<https://doi.org/10.1007/s10734-024-01345-4>.
- [5] A. Samala, S. Rawas, S. Criollo-C et al. "Emerging Technologies for Global Education: A Comprehensive Exploration of Trends, Innovations, Challenges, and Future Horizons." *SN Comput. Sci.* 5, 1175, 2024. doi:<https://doi.org/10.1007/s42979-024-03538-1>.
- [6] F. Stajano. Python in Education: Raising a Generation of Native Speakers. 2002. [Електронний ресурс]. Доступно: <https://www.cl.cam.ac.uk/~fms27/papers/python-native-speakers.html> Дата звернення: Грудень 19, 2024.

- [7] V. Dolgopolas, V. Dagienė, S. Minkevičius, L. Sakalauskas, "Python for Scientific Computing Education: Modeling of Queueing Systems", *Scientific Programming*, 22, 164306, 15 pages, 2014. doi: <https://doi.org/10.3233/SPR-140377>.
- [8] T. Georgieva-Trifonova, "Establishing Cross-Curricular Connections Through Python Programming in Computer Modeling Education" *TEM Journal*. Volume 12, Issue 3, Pages 1786-1791, 2023. doi:10.18421/TEM123-61.
- [9] G. Brindha. "Python in Education: Empowering the Next Generation of Coders. " 11.10.2023. [Електронний ресурс]. Доступно: <https://www.linkedin.com/pulse/python-education-empowering-next-generation-coders-brindha-igs> Дата звернення: Грудень 19, 2024.
- [10] J. C. G. Monsálvez, "Python como primer lenguaje de programación textual en la Enseñanza Secundaria". *Education in the Knowledge Society (EKS)*, 18(2), 147–162. 2017. doi:<https://doi.org/10.14201/eks2017182147162>.
- [11] T. Kohn, J. Staub, "The Two Powers: How Pascal and Python Shaped Programming Education", *Informatics in Education* 23, no. 4, 837-868, 2024. doi: 10.15388/infedu.2024.30.
- [12] S. Pradeep. "Python in Education: Teaching and Learning Programming." [Електронний ресурс]. Доступно: <https://medium.com/@pradeep.s.kvch/python-in-education-teaching-and-learning-programming-0fa0071785d6> Дата звернення: Грудень 19, 2024.
- [13] Т. Кобильник, О. Сікора, В. Жидик, О. Шаран, "Python як засіб навчання основ алгоритмізації у закладах загальної середньої освіти" *Інформаційні технології і засоби навчання*, №3 (89), с. 16-32, 2022.
- [14] О. Струк, Т. Зубик, "Написання веб-сайтів з допомогою бібліотеки Flask мови програмування Python" на *Сучасні інформаційні технології та інноваційні методики навчання: досвід, тенденції, перспективи. Матеріали XI Міжнародної науково-практичної інтернет-конференції*, Тернопіль, 2023, с.157-159.
- [15] S. Mukherjee, "How Python helped a teacher in evaluation", Oct 14, 2020. [Електронний ресурс]. Доступно: <https://subrata-mukherjee.medium.com/how-python-helped-a-teacher-in-evaluation-dc4ad2ee34d> Дата звернення: Грудень 19, 2024.

Матеріал надійшов до редакції 06.02.2025 р.

PYTHON PROGRAMMING LANGUAGE AS A TOOL IN SCIENTIFIC AND PEDAGOGICAL ACTIVITIES

Dmytro Treitiak

Doctor of Philosophy (PhD)

The Institute of Problems on Education of the National Academy of Educational Sciences of Ukraine, Kyiv, Ukraine

ORCID ID 0000-0003-3909-5062

stevandthree@gmail.com

Abstract. The article explores the potential applications of the Python programming language in scientific and pedagogical activities. The primary objective of the study is to identify effective approaches to utilizing Python for automating educational and research processes, analyzing large datasets, and developing interactive learning materials. The research methodology includes a literature review, an analysis of international experience regarding the use of Python in education and research, and an empirical examination of Python's integration into pedagogical practice.

The study highlights the key advantages of Python for educators and researchers, including: automation of computational tasks and data analysis; utilization of libraries for data visualization and processing (such as Matplotlib and Pandas); development of interactive educational resources and digital tools for scientific research (e.g., Flask); and facilitation of large-scale educational events using Python.

The article is notable for its innovative approach, as it presents a novel direction in Ukrainian pedagogical research. Specifically, it investigates the prospects of employing Python as a tool to enhance the efficiency of the educational process, with a focus on its potential application by teachers and lecturers in the humanities. The paper outlines opportunities for integrating Python into digital teaching practices, implementing interactive instructional methods, thereby improving educational quality and fostering digital competence among educators.

The article provides concrete examples of how Python can be applied in teaching, preparing research projects, and managing large volumes of information. It also presents international cases where

educators use Python to streamline professional activities. The study's findings underscore the relevance of Python as a means of enhancing educational and research efficiency and delineate specific areas in which its application is most beneficial for educators and scholars. The practical significance of the study lies in its potential use by teachers, university instructors, students, and postgraduate researchers in the context of academic or pedagogical work.

Keywords: Python; scientific and pedagogical activity; automation; education; digital competence.

REFERENCES (TRANSLATED AND TRANSLITERATED)

- [1] T. Vakalyuk, V. Bolotina, E. Baylyuk, O. Pokotilo, "Review of online gaming services for learning programming languages", *Innovative Pedagogy: Scientific Journal* No. 22, pp. 192-198, 2020. (in Ukrainian).
- [2] K. Bati, "Integration of Python into Science Teacher Education, Developing Computational Problem Solving and Using Information and Communication Technologies Competencies of Pre-service Science Teachers", *Informatics in Education* 21, no. 2, 235-251, 2022. doi:10.15388/infedu.2022.12. (in English).
- [3] N. Degtyareva, S. Petrenko, O. Udovichenko, "Working with graphic widgets when learning the Python programming language in secondary education institutions." *Education. Innovation. Practice*. No. 11(4), pp. 26-34, 2023. doi: <https://doi.org/10.31110/2616-650X-vol11i4-004>. (in Ukrainian).
- [4] S. Nicolajsen, S. Nielsen, L. Carlsen et al. "Programming education across disciplines: a nationwide study of Danish higher education. " *High Educ*, 2024. doi: <https://doi.org/10.1007/s10734-024-01345-4>. (in English).
- [5] A. Samala, S. Rawas, S. Criollo-C et al. "Emerging Technologies for Global Education: A Comprehensive Exploration of Trends, Innovations, Challenges, and Future Horizons. " *SN Comput. Sci.* 5, 1175, 2024. doi: <https://doi.org/10.1007/s42979-024-03538-1>. (in English).
- [6] F. Stajano. Python in Education: Raising a Generation of Native Speakers. 2002. [Online]. Available: <https://www.cl.cam.ac.uk/~fms27/papers/python-native-speakers.html> Accessed on: December 19, 2024. (in English).
- [7] V. Dolgopолоvas, V. Dagienė, S. Minkevičius, L. Sakalauskas, "Python for Scientific Computing Education: Modeling of Queueing Systems", *Scientific Programming*, 22, 164306, 15 pages, 2014. doi: <https://doi.org/10.3233/SPR-140377>. (in English).
- [8] T. Georgieva-Trifonova, "Establishing Cross-Curricular Connections Through Python Programming in Computer Modeling Education" *TEM Journal*. Volume 12, Issue 3, Pages 1786-1791, 2023. doi: 10.18421/TEM123-61. (in English).
- [9] G. Brindha. "Python in Education: Empowering the Next Generation of Coders. " 11.10.2023. [Online]. Available: <https://www.linkedin.com/pulse/python-education-empowering-next-generation-coders-brindha-igs> Accessed on: December 23, 2024. (in English).
- [10] J. C. G. Monsálvez, "Python as the first textual programming language in secondary education". *Education in the Knowledge Society (EKS)*, 18(2), 147–162. 2017. doi: <https://doi.org/10.14201/eks2017182147162>. (in Spanish).
- [11] T. Kohn, J. Staub, "The Two Powers: How Pascal and Python Shaped Programming Education", *Informatics in Education* 23, no. 4, 837-868, 2024. doi: 10.15388/infedu.2024.30. (in English).
- [12] S. Pradeep. "Python in Education: Teaching and Learning Programming." [Online]. Available: <https://medium.com/@pradeep.s.kvch/python-in-education-teaching-and-learning-programming-0fa0071785d6> Accessed on: December 23, 2024. (in English).
- [13] T. Kobylnyk, O. Sikora, V. Zhydyk, O. Sharan, "Python as a means of teaching the basics of algorithmization in secondary education institutions", *Information Technologies and Learning Tools*, No. 3 (89), pp. 16-32, 2022. (in Ukrainian).
- [14] O. Struk, T. Zubyk, "Writing websites using the Flask library of the Python programming language" in *Modern information technologies and innovative teaching methods: experience, trends, prospects. Proceedings of the XI International Scientific and Practical Internet Conference*, Ternopil, 2023, pp. 157-159. (in Ukrainian).
- [15] S. Mukherjee. "How Python helped a teacher in evaluation", Oct 14, 2020. [Online]. Available: <https://subrata-mukherjee.medium.com/how-python-helped-a-teacher-in-evaluation-dc4ad2ee34d> Accessed on: December 20, 2024. (in English).

