

Д. І. Угрин, М. Л. Ковальчук, Д. А. Кирстюк, О. В. Олар

FULL-STACK РОЗРОБКА ІНТЕЛЕКТУАЛЬНИХ ДОДАТКІВ

Навчально-методичний посібник

Електронне видання

**ЧЕРНІВЦІ
2023**

УДК 336:004(075.8)

У 72

Друкується за ухвалою вченої ради
Навчально-наукового інституту фізико-технічних та компютерних наук
Чернівецького національного університету
імені Юрія Федьковича
Протокол № 8 від 24.10.2023 р.

Рецензент:

Литвин В.В. – доктор технічних наук, професор, завідувач кафедри інформаційних систем та мереж Національного університету «Львівська політехніка»

Федів В.І. – доктор фізико-математичних наук, професор, завідувач кафедри біологічної фізики та медичної інформатики Буковинського державного медичного університету

Д. І. Угрин, М. Л. Ковальчук, Д. А. Кирстюк, О. В. Олар

Full-stack розробка інтелектуальних додатків. Навчальний посібник. – Чернівці: Чернівецький національний університет ім. Ю. Федьковича, 2023. – 362 с.

Пропонований посібник присвячено теоретичним та практичним аспектам full-stack розробці інтелектуальних додатків. Наведено типові приклади розробки інтелектуальних додатків та їх full-stack розробки. У посібнику розглянуто застосування операцій роботи фреймворками та бібліотеками, які роблять розробку швидшою та ефективнішою, а також з методами машинного навчання та штучного інтелекту.

Посібник адресовано студентам вищих навчальних закладів спеціальності „Комп’ютерні науки” та викладачам.

УДК 336:004(075.8)

© Д. І. Угрин, М. Л. Ковальчук, Д. А. Кирстюк, О. В. Олар

© Чернівецький національний університет імені Юрія Федьковича

ЗМІСТ

ПЕРЕДМОВА	10
РОЗДІЛ 1	11
ТЕМА 1. ОСНОВИ FULL-STACK РОЗРОБКИ ІНТЕЛЕКТУАЛЬНИХ ДОДАТКІВ НА МОВІ PYTHON ..	11
1.1. Типізація	11
1.2. Типи даних в Python	12
1.3. Змінні	12
1.4. Математичні операції	16
1.4.1. Арифметичні операції з цілими і дробовими числами	16
1.4.2. Робота з комплексними числами	17
1.4.3. Бітові операції	17
1.4.4. Представлення чисел в інших системах числення	18
1.5. Бібліотека (модуль) math	19
1.6. Коментарі.....	19
1.7. Порядок обчислень	21
1.8 Неасоціативні оператори	22
1.9. Логічні оператори	23
1.10. Умовний оператор розгалуження if.....	23
1.10.1. Які вирази повертають False?	24
1.10.2. Конструкція if - else	25
1.10.3. Конструкція if - elif – else.....	26
1.11. Оператор циклу while	26
1.12. Оператори break і continue	26
1.13. Оператор циклу for	27
1.14. Списки.....	28
1.14.1. Створення, зміна, видалення списків і робота з його елементами..	29
1.14.2. Методи списків	31
1.14.3. List Comprehensions	32
1.14.4. List Comprehensions як обробник списків	33
1.14.5. Слайси / Зрізи	34
1.14.6. List Comprehensions через генератор	35
1.15. Кортежі	35
1.15.1. Створення, видалення кортежів і робота з його елементами.....	36
1.16. Рядки	37
1.16.1. Методи рядків	38
1.16.2. Методи find() і replace().....	39
1.16.3. f-рядки	39
1.17. Словники.....	40
1.17.1. Додавання і видалення елемента.....	40
1.17.2. Методи словників	41

ТЕМА 2. РОБОТА З ВИНЯТКАМИ. ПОМИЛКИ. ОБРОБКА ВИНЯТКІВ. ОПЕРАТОР WITH. ПРОЕКТУВАННЯ З ВИКОРИСТАННЯМ ВИКЛЮЧЕНЬ	45
2.1. Методи словників	45
2.2. Як читати Traceback.....	46
2.3 Використання винятків	46
2.3.1. Як працює перехоплення винятків	47
2.3.2. Ієрархія винятків.....	50
2.4. Повна форма try except	53
2.5. else - try.. ..	54
2.6. Генерація винятків в Python.....	55
2.7. Власні винятки	57
2.8. Raise без визначеного винятку	57
2.9. Логування (logging).. ..	61
2.10. Основні конфігурації	62
2.10.1. Форматування вихідних даних.....	63
2.10.2. Логування даних	63
2.10.3. Логування винятків.....	64
ТЕМА 3. ФУНКЦІЇ ТА ЇХ ПАРАМЕТРИ. ЛОКАЛЬНІ ЗМІННІ. GLOBAL. NONLOCAL. ЗНАЧЕННЯ ЗА ЗАМОВЧУВАННЯМ. КЛЮЧОВІ АРГУМЕНТИ. ЗМІННЕ ЧИСЛО АРГУМЕНТІВ. RETURN. СТРОКИ ДОКУМЕНТАЦІЇ. ЛЯМБДА-ФУНКЦІЇ	67
3.1. Функції	67
3.2. Не обов'язкові аргументи	69
3.2.1. *args і **kwargs	70
3.3. Перевірка функції за допомогою assert	72
3.4. Змінні незмінних (immutable) та змінних (mutable) типів та функції ...	72
3.5. Функції як об'єкти	73
3.6. Область видимості функції (scope)	74
3.7. Локальні і глобальні змінні	75
3.8. Вкладені (enclosing) області видимості	76
3.9. Анонімні функції	78
3.10. Використання lambda-функцій.....	79
3.11. Замикання (closure).....	80
3.12. Декоратори	81
3.12.1. Ланцюжки з декораторів.....	83
3.12.2. Шаблони декораторів	87
ТЕМА 4. МОДУЛІ І ПАКЕТИ ОСНОВИ НАПИСАННЯ МОДУЛІВ. СТВОРЕННЯ ВЛАСНИХ МОДУЛІВ	90
4.1. Модулі в Python	90
4.1.2. Стандартні модулі.....	91
4.1.3. Функція dir().....	92

4.2. Модулі як об'єкти.....	92
4.3. Пакети	93
4.4. Відносний імпорт.....	94
4.5. <code>__init__.py</code> і шаблон «Фасад»	95
4.6. Виконання пакетів	96
4.7. Встановлення пакетів	96
4.8. Менеджер пакетів Python <code>pip</code>	96
ТЕМА 5. РОБОТА З ФАЙЛАМИ. ФУНКЦІЇ ДЛЯ РОБОТИ З ФАЙЛАМИ. МЕНЕДЖЕР КОНТЕКСТІВ WITH. ФОРМАТИ JSON, XML, CSV.....	
5.1. Відкриття текстового файлу	101
5.2. Запис даних у файл.....	103
5.3. Створення та видалення каталогів.....	104
5.3.1. Отримати список вмісту каталогу.....	104
5.3.2. Отримати метаданні файлу.....	105
5.3.3. Перейменування файлу	105
5.3.4. Видалення файлу	106
5.3.5. Копіювання файлу	106
5.4. Копіювання папок і файлів в них рекурсивно	106
5.5. Існування файлу	107
5.6. Зчитування даних з текстових файлів	107
5.7. Інструкція <code>with</code>	108
5.8. Структуровані текстові файли.....	111
5.9. XML.....	113
5.10. JSON	116
ТЕМА 6. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ НА МОВІ PYTHON. МЕТОДИ ОБ'ЄКТА І МЕТОДИ КЛАСУ	
6.1. Поняття об'єктно-орієнтоване програмування	121
6.2. Абстракція	123
6.3. Інкапсуляція	123
6.4. Спадкування.....	124
6.5. Поліморфізм.....	124
6.6. Створення класу.....	126
6.7. Атрибути.....	127
6.7.1. Користувацькі атрибути.....	128
6.7.2. Перевірка типів	135
6.8. Простір імен в класах і об'єктах	136
6.9. Методи об'єкта і методи.....	137
6.9.1. <code>@staticmethod</code>	138
6.9.2. <code>@classmethod</code>	138
6.10. Інкапсуляція (<code>Public</code> , <code>Protected</code> , <code>Private</code>).....	141
6.10.1. Чим небезпечні <code>Public</code> атрибути і методи	142
6.10.2. <code>Protected</code> . Чи достатньо цього.....	142
6.10.3. Приватні (<code>Private</code>) атрибути і методи.....	143
6.10.4. Властивості (<code>Property</code>)	145

6.11. Спадкування і поліморфізм	150
6.12. Доповнення, воно ж розширення, методу	151
6.13. Параметри зі значеннями за замовчуванням у батьківського класу.	152
6.14. Відношення: «IS-A».....	154
6.15. Відношення: «АКО».....	155
6.16. super()	155
6.17. Параметри super()	157
6.18. Множинне спадкування	158
6.19. Method Resolution Order	158
6.20. Проектування ієрархії класів і клас object	161
6.21. Переваги та недоліки ООП	162
ТЕМА 7. ЗАСТОСУВАННЯ ООП В PYTHON. МІКСИНИ. АТРИБУТ __SLOTS__. ІТЕРАТОРИ. ГЕНЕРАТОРИ. YIELD. КОРУТИНИ. ПАТЕРН SINGLETON В PYTHON. МЕТАКЛАСИ	165
7.1. Міксини в Python	165
7.1.1. Використання Міксинів для «взламування» успадкування	166
7.2. __slots__	168
7.3. Агрегація.....	170
7.4. Ітератори.....	170
7.5. Генератори.....	171
7.6. Що таке yield	172
7.7. Порівняння продуктивності return і yield.....	173
7.8. Корутини.....	173
7.9. __new__	174
7.10. Як влаштований Singleton.....	175
7.10.1. Реалізація класичного Сінглтона в Python	175
7.10.2. Відкладений екземпляр в Singleton.....	176
7.10.3. Singleton на рівні модуля	176
7.10.4. Моностаціональний (Monostate) Сінглтон.....	177
7.10.5. Сінглтон і метакласи	177
7.10.6. Перший приклад використання Singleton	178
7.10.7. Другий приклад використання Singleton.....	179
7.11. Метакласи	180
7.12. Класи в Python це об'єкти.	181
7.12.1. Динамічне створення класів.	182
7.13. Атрибут __metaclass__	182
7.14. Переваги та недоліки ООП.	183
РОЗДІЛ 2.	186
ТЕМА 8. UNIT-ТЕСТУВАННЯ В МОВІ PYTHON. ФРЕЙМВОРК UNITTEST. ПЛАГІН COVERAGE. МЕТОДИ ДЛЯ ЗАПУСКУ ТЕСТІВ. ФРЕЙМВОРК ДЛЯ ТЕСТУВАННЯ PYTEST. ФІКСТУРИ. ПАРАМЕТРИЗАЦІЯ І КОМБІНУВАННЯ ТЕСТІВ.....	186
8.1. Поняття Unit-тест.....	186
8.2. Framework'и для проведення автономного тестування в Python.....	187
8.2.1. Приклад тестування програми без framework'ів.....	187

8.2.2. Приклад тестування програми з використанням unittest.....	188
8.3. Основні структурні елементи unittest	189
8.4. Інтерфейс командного рядка (CLI)	189
8.5. Робота з TestCase	189
8.6. Методи, що використовуються при запуску тестів	190
8.7. Методи, які використовуються при написанні тестів.....	191
8.8. Завантаження і запуск тестів	192
8.9. Умовний пропуск тестів.....	193
8.9.1. Пропуск класів	193
8.10. Управління станами та залежностями.....	195
8.11. Коли слід уникати використання фікстур.....	197
8.12. Позначки: Тестування за категоріями	198
8.13. Параметризація: комбінування тестів	199
8.14. Корисні плагіни pytest	199

ТЕМА 9. ВЕБ-ФРЕЙМВОРК FLASK. СТАТИЧНІ ФАЙЛИ. РОУТИНГИ. КОНТЕКСТИ У FLASK. ВІДПОВІДІ СЕРВЕРА. ПЕРЕХОПЛЕННЯ ЗАПИТІВ.....

9.1. Flask.....	202
9.2. Як Flask обробляє запит?	206
9.3. Контексти у Flask.....	206
9.4. Обробка запиту	207
9.5. Context Locals	210
9.6. Створення відповідей за допомогою кортежів.....	213
9.7. Перехоплення запитів.....	214
9.8. Скасування запиту за допомогою abort()	215
9.9. Шаблони	215
9.9.1. Рендеринг шаблонів в консолі	217

ТЕМА 10. ШАБЛОНІЗАЦІЯ. ШАБЛОНІЗАТОР JINJA2. ЦИКЛИ, УМОВНІ ВИРАЗИ, ФІЛЬТРИ І МАКРОСИ В JINJA2. ЕКРАНУВАННЯ. ВКЛАДЕНІ ШАБЛони. СПАДКУВАННЯ ШАБЛОНІВ.....

10.1. Шаблони	220
10.2. Рендеринг шаблонів в консолі.....	222
10.3. Основи шаблонізатора Jinja	222
10.3.1. Змінні, вирази і виклики функцій	222
10.3.2. Виклик функції.....	223
10.3.3. Атрибути і методи	223
10.3.4. Коментарі.....	224
10.3.5. Оголошення змінних	224
10.3.6. Цикл і умовні вирази	224
10.4. Фільтри.....	228
10.5. Макроси	229
10.6. Екранування	230
10.7. Вкладені шаблони.....	231
10.8. Спадкування шаблонів	232

ТЕМА 11. РОБОТА З URL У FLASK. СТАТИЧНІ ФАЙЛИ. КОРИСНІ РОЗШИРЕННЯ ДЛЯ FLASK.....	237
11.1. Створення URL у Flask	237
11.2. Робота зі статичними файлами у Flask	238
11.3. Розширення Flask.....	239
11.4. Розширення Flask-Script.....	240
11.5. Команда shell	242
11.6. Створення команд.....	242
11.7. Автоматичний імпорт об'єктів	244
ТЕМА 12. РОБОТА З ФОРМАМИ. FLASK-WTF. ТЕСТУВАННЯ І ВІЗУАЛІЗАЦІЯ ФОРМ. ПІДТВЕРДЖЕННЯ ФОРМ. РОБОТА З COOKIES. СЕСІЇ У FLASK. РОБОТА З БАЗАМИ ДАНИХ. SQLALCHEMY. СТВОРЕННЯ МОДЕЛЕЙ. СТВОРЕННЯ ТАБЛИЦЬ. РОЗГОРТАННЯ FLASK-ДОДАТКУ НА ПЛАТФОРМІ HEROKU.....	247
12.1. Робота з формами у Flask.....	247
12.2. WTForms	250
12.3. Встановлення SECRET_KEY	251
12.4. Форми в консолі.....	251
12.5. Візуалізація форми	252
12.6. Візуалізація полів за допомогою циклу	255
12.7. Впливаючі повідомлення	260
12.8. Cookies	262
12.8.1. Налаштування cookies у Flask	263
12.8.2. Доступ до cookies.....	264
12.8.3. Недоліки cookies	266
12.9. Сесії у Flask	267
12.9.1. Як читати, записувати і видаляти дані сесії.....	267
12.9.2. Зміна даних сесії	268
12.10. Створення баз даних у Flask.....	269
12.11. Створення моделей.....	271
12.12. Визначення відносин (зв'язків).....	281
12.13. Відношення один-до-багатьох.....	282
12.14. Відношення один-до-одного.....	283
12.15. Відношення багато-до-багатьох.....	284
12.16. Розгортання Flask- додатку на платформі Heroku.....	285
ТЕМА 13. СТВОРЕННЯ БОТІВ ДЛЯ МЕСЕНДЖЕРА TELEGRAM. TELEGRAM-API. ТОКЕНИ. WEBHOOK. BOTGRAM. AIOGRAM.....	289
13.1. Про Telegram API і Telegram Bot API.....	289
13.2. Видимість повідомлень у групах	292
13.3. Кнопки	292
13.4. Як отримувати повідомлення користувача.....	295
13.5. Як створити і зареєструвати бота.....	296
13.6. Обробник команди /start.....	299
13.7. Обробник команди /help.....	300

13.8. Додавання обробника команди /exchange	301
13.9. Обробник для кнопок вбудованої клавіатури.....	302
13.10. Реалізація обробника кнопки оновлення (Update)	304
13.11. Створення telegram-бота для отримання інформації про погоду за допомогою Aiogram.....	306
ТЕМА 14. РОЗРОБКА ДОДАТКІВ З GUI. ОГЛЯД І ОСНОВИ РОБОТИ З TKINTER. БІБЛІОТЕКА WXPYTHON. NUMPY І SCIPY.....	317
14.1. Що таке NumPy	317
14.2. Створення масиву	317
14.3. Доступ до елементів, зрізи.....	321
14.4. Форма масиву і її зміна	323
14.5. Перестановка осей і транспонування	327
14.6. Клонування даних.....	331
14.7. Математичні операції над елементами масиву	333
14.8. Матричне множення.....	335
14.9. Агрегатори.....	336
14.10. Побудуємо алгоритм лінійної низькочастотної фільтрації зображення	337
14.11. Інтерполяції для imshow/matshow	339
14.12. SciPy	340
14.13. Інтегрування	342
14.14. Інтерполяція	342
14.15. Сплайни	344
ТЕМА 15. РОЗРОБКА ПАРСЕРІВ НА МОВІ PYTHON. СИНТАКСИЧНИЙ АНАЛІЗ HTML-ФАЙЛІВ ЗА ДОПОМОГОЮ МОВИ PYTHON. ПАРСИНГ І РОБОТА З ВЕБ-ДАНИМИ. БІБЛІОТЕКА BEAUTIFUL SOUP 4.....	348
15.1. Парсинг	348
15.2. Створення скрипту скрапінгу.....	349
15.3. Проходження структурою HTML	350
15.4. Парсинг HTML-розмітки	351
15.5. LXML	351
15.6. Аутентифікація	353
15.7. Властивість text бібліотеки BeautifulSoup.....	354
15.8. Логіка отримання всіх точок даних із однієї сторінки	356
15.9. Скрапінг з урахуванням пагінації	357
ЛІТЕРАТУРА.....	361