Структури даних

№ уроку: 6 **Курс:** Python Базовий

Засоби навчання: Персональний комп'ютер/ноутбук стандартної продуктивності

Огляд, мета та призначення уроку

Познайомитись з ускладненими варіаціями комбінацій структур даних, розглянути зміст їх застосування та тонкощі роботи з ними. Також буде розглянуто так звані comprehensions. Після цього уроку ви зможете працювати зі складними структурами даних і використовувати comprehensions у ваших задачах.

Вивчивши матеріал даного заняття, учень зможе:

- Розуміти складні структури даних та працювати з ними.
- Знати, що таке comprehensions і вміти їх застосовувати на практиці.

Зміст уроку

1. Приклади комбінацій структур даних
2. Що таке comprehensions в Python
3. Вирішення задач

Резюме

- Комбінувати можна майже будь-які структури даних між собою, створюючи тим самим складні та комплексні об'єкти. Наприклад, ви можете зробити список словників або зробити словник зі списками словників, де будуть списки рядків і т.д. Фантазія тут може бути безмежною.
- Однак для кожного завдання існує своя ідеальна структура даних. Наприклад, якщо вам потрібно зберігати дані про користувачів, то буде зручно зробити список словників. Таким чином, вам буде легко порахувати їх кількість (len), взяти, видалити, додати новий елемент. А найголовніше це те, що ви зможете дуже легко працювати з такою структурою навіть без коду. У кожному словнику буде наочно видно пари ключ-значення, де ви зможете записати будь-які дані про користувачів.
- Але якщо ви вирішите зробити це завдання через список списків, створіть собі зайві складності. Наприклад, ви банально можете забути яка позиція у внутрішніх списках за що відповідає (25 це вік чи зарплата чи що?). У випадку зі словниками - у вас буде {'age': 25} і все зрозуміло.
- Буває таке, що структури ще складніші. Це залежить від складності даних, з якими ви працюєте. Якщо у користувачів є дані про придбану ними нерухомість, то у кожного користувача всередині його словника за ключом 'real_estate' буде свій список словників з даними про нерухомість. А якщо кожен куплений будинок ще матиме дані про його меблі, то це ще нові списки словників. Таким чином структури даних можуть бути дуже величезними та глибокими.

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- Comprehensions – це один з one-liners мови Python. Спеціальний синтаксис, який дозволить вам працювати з структурами, що ітеруються за допомогою циклу for одним рядком. Крім корисної властивості скорочення простору коду, ще варто відзначити, що comprehensions працюють швидше за звичайні цикли.
- Синтаксис comprehensions наступний [object for object in iterable]. Цей запис просто продублює об'єкт, що ітерується. Якщо ви хочете зробити додаткові перетворення з ним, наприклад звести в квадрат кожен елемент списку, то це буде так: sqaures_list = [element**2 for element in my_list]

Закріплення матеріалу

- Які структури даних можна поєднувати між собою?
- Чи можна використовувати вкладені comprehensions?
- Чи є завжди розумним використовувати comprehensions замість циклів так як вони швидше?

Додаткове завдання

Пофантазуйте зі структурами даних. Спробуйте об'єднувати різні структури та подивитися, як з ними працювати (списки словників списків, словники словників, списки списків зі словниками списків тощо) Напишіть цикли для ітерації таких структур.

Самостійна діяльність учня

1. Напишіть комбінацію циклів, щоб проітерувати всі елементи складної структури даних виду d = {"a": {"1": [...], "2": [...], ...}, ...} (словник словників у яких лежать списки)
2. Напишіть comprehension, який ітерує рядок "1_2,40_5,5_32" (розбити по комам) і кожен елемент розбиває за символом "_" і отримані елементи призводить до типу int, і складає їх, утворюючи список цілих чисел. (Ви можете використовувати функцію lambda або написати звичайну, яка приймає рядок виду "1_2" на вхід і повертає число як суму значень з рядка; використовуйте метод split()).

Рекомендовані ресурси

- Nested dicts in Python
- Рекурсивна робота із вкладеними словниками
- Вкладені структури
- Comprehensions
- List comprehensions
- Double comprehension

CyberBionic systematics

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com