

# Python Базовий

Модулі Python

# Python Базовий

## Introduction



Вікторія Бойчук

Python Developer, тренер CBS

 [Вікторія Бойчук](#)



# Python Базовий

Тема уроку

Модулі Python

# Python Базовий

## План уроку

1. Що таке модулі та які з них будуть розглянуті в рамках цього уроку
2. Розбір модулів та практичні приклади
3. Як написати свій модуль
4. Розв'язання задач по пройденим модулям

# Python Базовий

## Які модулі Python будуть в уроці

- **collections** - модуль із розширеними структурами даних
- **math** - модуль готових математичних функцій
- **random** - модуль для генерації випадкових значень
- **itertools** - модуль із готовими функціями комбінаторики
- **re** - модуль підтримки регулярних виразів
- **datetime** - модуль дати та часу

Щоб використовувати модуль у коді, потрібно зробити імпорт :

```
import module_name
```

# Python Базовий

## Модуль collections

```
1  import collections
2
3  counts = collections.Counter([1, 1, 2, 3, 3, 3])
4  print(counts)  # Counter({3: 3, 1: 2, 2: 1})
5
6  dict_of_lists = collections.defaultdict(list)
7  for value in [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]:
8      if value % 2 == 0:
9          dict_of_lists['even'].append(value)
10     else:
11         dict_of_lists['odd'].append(value)
12 print(dict_of_lists)
13 # {'odd': [1, 3, 5, 7, 9], 'even': [2, 4, 6, 8, 10]}
14
15 d = collections.deque([1, 2, 3])
16 d.popleft()  # [2, 3]
17 d.append(4)  # [2, 3, 4]
18 d.appendleft(6)  # [6, 2, 3, 4]
19 d.pop()  # [6, 2, 3]
20
21
```

# Python Базовий

## Модуль math

```
1  import math
2
3  print(math.sqrt(16)) # 4
4  print(math.pi) # 3.141592653589793
5  print(math.e) # 2.718281828459045
6  print(math.ceil(2.1)) # 3
7  print(math.floor(2.1)) # 2
8  print(math.dist([1, 0, 0], [0, 1, 0])) # 1.4142
9  print(math.isnan(math.nan)) # True
10 print(math.exp(2)) # ~math.e ** 2 ~ 7.38905609893065
11 print(math.log(8, 2)) # 3.0 (if no base, it's natural log ie. base~2.7)
12
13
```

# Python Базовий

## Модуль random

```
1  import random
2
3  print(random.random()) # random value from 0 to 1
4  print(random.randint(0, 10)) # random value in giver arguments range
5  print(random.choice([1, 2, 3, 4, 5])) # random element from iterable object
6  random.seed(42) # random generator start number
7  seq = [1, 2, 3, 4, 6]
8  print(random.shuffle(seq)) # returns nothing, shuffles the sequence
9  print(seq)
10
11
```



# Python Базовий

## Модуль itertools

```
1  import itertools
2
3  combs = itertools.combinations([1, 2, 3, 4], 2)
4  print(list(combs)) # [(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)]
5  permutations = itertools.permutations([1, 2, 3, 4], 2)
6  print(list(permutations))
7  # [(1, 2), (1, 3), (1, 4), (2, 1), (2, 3), (2, 4), (3, 1), (3, 2), (3, 4), (4, 1), (4, 2), (4, 3)]
8  print(list(itertools.combinations_with_replacement([1, 2, 3], 2)))
9  # [(1, 1), (1, 2), (1, 3), (2, 2), (2, 3), (3, 3)]
10 print(list(itertools.accumulate([1, 2, 3, 4, 5]))) # [1, 3, 6, 10, 15]
11 print(list(itertools.accumulate([1, 2, 3, 4, 5], lambda a, b: a - b))) # [1, -1, -4, -8, -13]
12 print(list(itertools.filterfalse(lambda x: x % 2, range(10))))
13 # [0, 2, 4, 6, 8]
14
15
```

# Python Базовий

## Модуль re

```
1 import re
2
3 text = "Hi @mr_alex ! My name is John and I love coffee. #coffeelover #2021"
4
5 print(re.sub(r"[.!?\\"-]", '', text))
6 print([element for element in re.finditer(r"#[A-Za-z0-9]*", text)])
7 print(re.split(r"[.!?\\"-]", text))
8
9
```

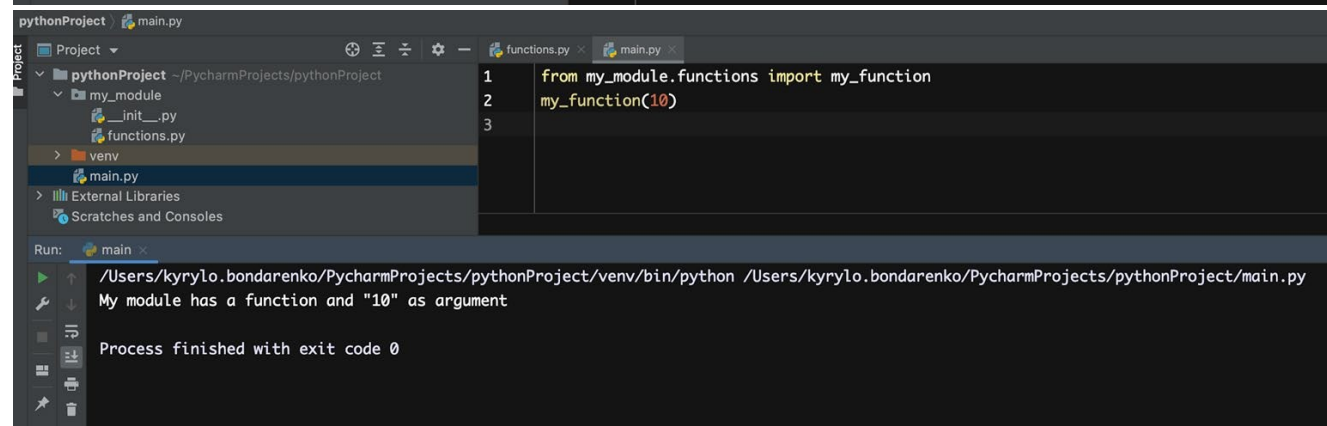
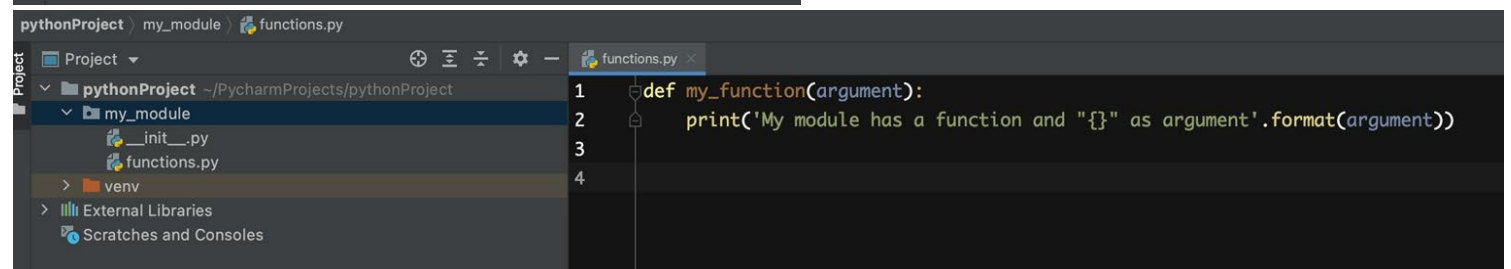
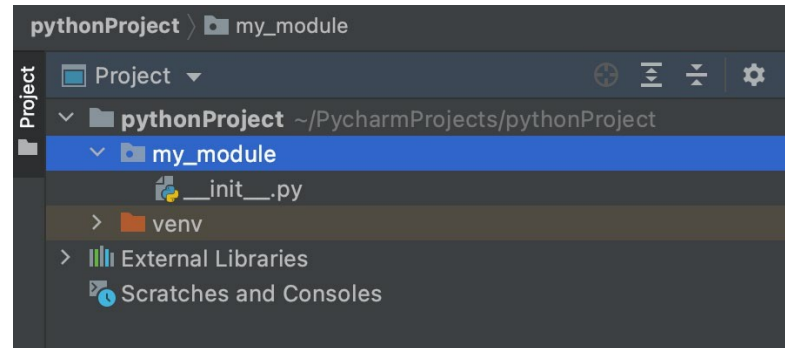
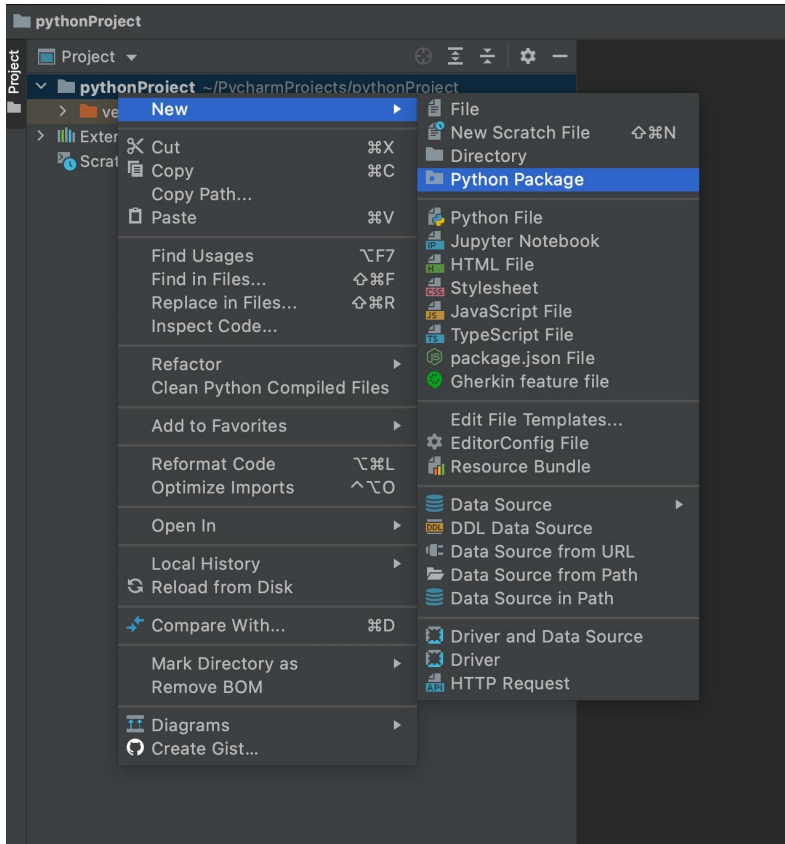
# Python Базовий

## Модуль datetime

```
1  from datetime import datetime
2
3  now = datetime.now()
4  print(now)
5  print(datetime.time(now))
6  print(now.year, now.month, now.day)
7
8  date_string = "21 June, 2018"
9  date_object = datetime.strptime(date_string, "%d %B, %Y")
10 print(date_object)
11
```

# Python Базовий

## Пишемо свій модуль



# Python Базовий

## Задачі: частина 1

Є рядок : `"blue car is parked right behind the blue building with blue walls and red door"`

1. Написати функцію `process`, яка на вхід приймає рядок. Насамперед усередині функції знайдіть найчастіше слово (модуль `combinations`) і винесіть код в окремий метод `get_max_count_word`.
2. Знайдіть в тексті всі комбінації (пари по 2 слова) із словом, що найчастіше зустрічається. Винесіть код окремим методом `get_max_count_word_combinations`, який поверне список кортежів.
3. Виберіть випадкову комбінацію зі списку, отриманого в пункті 2, та поверніть її у методі `process`.

# Python Базовий

## Відповідь

```
1 import collections
2     import itertools
3 import random
4
5
6 def get_max_count_word(counts_dict):
7     max_count = 0
8     max_count_word = None
9     for word, count in counts_dict.items():
10         if count > max_count:
11             max_count = count
12             max_count_word = word
13     return max_count_word
14
15
16 def get_max_count_word_combinations(words, max_count_word):
17     combinations = itertools.combinations(words, 2)
18     max_count_word_combinations = [comb for comb in combinations if max_count_word in comb]
19     return max_count_word_combinations
20
21
22 def process(text):
23     words = text.split(' ')
24     counts = collections.Counter(words)
25     max_count_word = get_max_count_word(counts)
26     max_count_word_combinations = get_max_count_word_combinations(words, max_count_word)
27     random_combination = max_count_word_combinations[random.randint(0, len(max_count_word_combinations) - 1)]
28     return random_combination
29
30
31 process(text="blue car is parked right behind the blue building with blue walls and red door")
32
33
```

# Python Базовий

## Задачі: частина 2

Є рядок: `"I want to be a professor, but only one thing I can solve is x=exp(8)+sqrt(17)-log(5)"`

1. Оголосіть функцію `process`, яка прийматиме на вхід рядок. Насамперед знайдіть саме математичне вираз у переданому тексті.
2. Зробіть окремий словник, до якого запишіть ключами імена всіх операцій як у тексті, а їх значеннями - відповідні функції у пакеті `math`, тобто `{ 'exp': math.exp, ... }`.
3. Розділіть підрядок із виразом за знаком `"+"`. Ітеруючи кожен елемент у отриманому списку виділіть у ньому окремо назву функції та окремо значення. З раніше зробленого словника візьміть за отриманим ім'ям функцію та передайте в неї отримане значення. По черзі додавайте отримані значення до нової змінної `result = 0`.
4. Зробіть змінну `"Result = {response}; Date {date}"`
5. Використовуючи метод рядка `format`, заповніть значення `response = result` та значення `date =` поточної дати у форматі день/місяць/рік без часу. (Використовуйте `strftime` та формат `"%d/%m/%y"`)
6. Отриманий рядок виведіть у консоль.

# Python Базовий

## Відповідь

```
1 import re
2 import math
3 import datetime
4
5 t = "I want to be a professor, but only one thing I can solve is  $x=\exp(8)+\sqrt{17}+\log(5)$ "
6
7
8 def process(text):
9     re_match = re.search('x=', text)
10    equation = text[re_match.span()[1]:]
11    ops = {'exp': math.exp, 'sqrt': math.sqrt, 'log': math.log}
12    actions = equation.split('+')
13    result = 0
14    for action in actions:
15        action = action.replace('(', ' ').replace(')', '')
16        op, value = action.split(' ')
17        result += ops[op](int(value))
18    report = "Result = {response}; Date {date}".format(
19        response=result,
20        date=datetime.datetime.now().strftime('%d/%m/%y')
21    )
22    print(report)
23
24
25 process(t)
26
27
```



# Інформаційний відеосервіс для розробників програмного забезпечення



# Перевірка знань

TestProvider.com



Перевірте як Ви засвоїли даний матеріал на [TestProvider.com](https://testprovider.com)

TestProvider – це online сервіс перевірки знань з інформаційних технологій. За його допомогою Ви можете оцінити Ваш рівень та виявити слабкі місця. Він буде корисним як у процесі вивчення технології, так і для загальної оцінки знань IT спеціаліста.

Успішне проходження фінального тестування дозволить Вам отримати відповідний Сертифікат.

# Python Базовий

Дякую за увагу! До нових зустрічей!



Вікторія Бойчук  
Python Developer, тренер CBS