

Python для решения научных задач

Знакомство с Python

Литвинов Владимир Геннадьевич (litvinov.vg@ssau.ru)

Факультет информатики

1 Лекционные.

2 Практические (лабораторные).

- ❶ Знакомство с Python.
- ❷ Основы синтаксиса.
- ❸ Коллекции и строки.
- ❹ Списки и строки. Основные функции обработки. Срезы.
- ❺ Структурирование программы.
- ❻ Работа с файловой системой.

- 1 Библиотека math
- 2 Библиотека random.
- 3 Библиотека numpy.
- 4 Библиотека pandas.
- 5 Библиотека matplotlib.

```
self.debug = settings.getbool('debug')
self.logger = logging.getLogger('log')
if path:
    self.file = open(path, 'w')
    self.file.seek(0)
    self.fingerprints.update({path: self.file})
```

```
classmethod
def from_settings(cls, settings):
    debug = settings.getbool('debug')
    return cls(job_dir(settings), debug)
```

```
def request_seen(self, request):
    fp = self.request_fingerprints(request)
    if fp in self.fingerprints:
        return True
    self.fingerprints.add(fp)
    if self.file:
        self.file.write(fp + os.linesep)
```

```
def __call__(self, request):
    return self(request)
```

Что? Где? Когда? Кто?

Python, Нидерланды, 1991 г., Гвидо Ван Россум



Технические:

- 1 Мультипарадигменность.
- 2 Кроссплатформенность.
- 3 Динамическая строгая типизация.
- 4 Автоматическое управление памятью.
- 5 Отсутствие операторных скобок (вложенность формируется отступами).

С точки зрения специалиста:

- 1 Ясность, простота и лаконичность синтаксиса.
- 2 Наличие огромного количества библиотек и фреймворков.
- 3 Обширная поддержка.
- 4 Популярность (самый популярный по версии [github](#)).

- 1 Web-разработка.
- 2 Анализ данных.
- 3 Машинное обучение и искусственный интеллект.
- 4 Автоматизация решения различных задач.
- 5 И т.д.

```
self.debug = settings.DEBUG
self.logger = logging.getLogger(__name__)
if path:
    self.file = os.path.join(settings.LOGGING_PATH,
                             settings.LOGGING_FILE)
    self.file.write(f)
    self.fingerprints.add(fp)
```

```
@classmethod
def from_settings(cls, settings):
    """Returns a new instance of the class,
    using the settings object to configure it.
    """
    obj = cls()
    obj.configure(settings)
    return obj

def request_seen(self, request):
    fp = self.request_fingerprint(request)
    if fp in self.fingerprints:
        return True
    self.fingerprints.add(fp)
    if self.file:
        self.file.write(fp + os.linesep)
    return True
```


- 1 Dropbox.
- 2 BitTorrent.
- 3 Mercurial.
- 4 GIMP.
- 5 Instagram backend.
- 6 И т.д.

```
self.debug = settings.getbool('debug', False)
self.logger = logging.getLogger(__name__)
if path:
    self.file = os.path.join(settings.get('path', ''),
                              settings.get('filename', 'fingerprint.log'))
    self.file.seek(0)
    self.fingerprints.update({path: self.file})

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool('debug', False)
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def request_fingerprint(self, request):
        return hashlib.md5(
            request.get_full_url().encode('utf-8')).hexdigest()
```

- 1 Нативный.
- 2 Anaconda.

```
self.debug = settings.getbool('debug')
self.logger = logging.getLogger(__name__)
if path:
    self.file = open(path, 'a')
    self.file.seek(0)
    self.fingerprints.update(self.get_fingerprints(request))

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool('debug')
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def __call__(self, request):
        pass
```

- 1 Текстовый редактор (**Notepad++**, **Sublime Text**, ...).
- 2 IDE: Интегрированная среда разработки (**Visual Studio**, **PyCharm**).
- 3 **Jupyter Notebook**.
- 4 **Google Colab**.

Найдите ответ

```
print("".join(chr(a) for a in (2**6+8, 101, int("0x6C 16), 108, int("1"*3),  
30+-8%11)))
```

```
self.debug = debug
self.logger = logging.getLogger(__name__)
if path:
    self.file = open(path, 'a')
    self.file.seek(0)
    self.fingerprints.update(self.get_fingerprints())
```

@Спасибо за внимание.

```
def from_settings(cls, settings):
    debug = settings.getbool('debug', False)
    return cls(job_dir(settings), debug)

def request_seen(self, request):
    fp = self.request_fingerprint(request)
    if fp in self.fingerprints:
        return True
    self.fingerprints.add(fp)
    if self.file:
        self.file.write(fp + os.linesep)
    return True
```