

Um pouco sobre lambdas

Na hora de converter o objeto para JSON, utilizamos uma função e passamos como parâmetro `default` da função `dump`.

O parâmetro `default` recebe um tipo de objeto no Python chamado de `callable`, ou invocável. Isto é, um tipo de objeto, como uma função ou método que pode ser invocado. Internamente, a função `dump` invocará o objeto que estamos passando como parâmetro, passando um contato para ele.

No vídeo, passamos uma função chamada de `contato_para_dict`; contudo, poderíamos passar uma função anônima para o parâmetro, ou seja, uma função **sem nome**.

Para isso, basta falar que o parâmetro `default` recebe uma função (`lambda`), que tem um argumento (`contato`) e que seu retorno (`:`) é o dicionário deste contato. Ao final, a chamada da função poderia ficar assim:

```
# código omitido
json.dump(contatos, arquivo, default=lambda contato: contato.__dict__)
```

O conceito do `lambda` está fortemente ligado ao paradigma de programação funcional. Se deseja conhecer um pouco mais sobre esse estilo de programação, leia [este post do blog da Caelum \(https://blog.caelum.com.br/programacao-funcional-no-python/\)](https://blog.caelum.com.br/programacao-funcional-no-python/), que aborda a programação funcional e Python.