

Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования «Московский
государственный технический университет им. Н. Э. Баумана»



Тема

Отчет по лабораторной работе №2 «Python. Функциональные
ВОЗМОЖНОСТИ»

Выполнил: студент группы СГНЗ-71
Корчагин А. С.

Москва 2015

Генератор — функция, которая в данном задании принимает на вход список словарей с профессиями, доставет оттуда значение по ключу «job-name» и возвращает его в вызывающую функцию.

Итератор — класс, получающий на вход итерируемый объект, и последовательно выполняющий указанные в итераторе методы с каждым объектом из переданного объекта.

В 4-й ЛР итератор принимает в конструкторе функцию-генератор и аргументы `kwargs`, в которые включена переменная `ignore_case` и ее булево значение. Данная переменная по-умолчанию равна `False`, что говорит о том, что слова с разными регистрами букв будут считаться итератором разными. Также в классе присутствует переменная класса «множество» `result`. Каждую итерацию (метод `__next__()`) итератор вызывает функцию-генератор, получает название работы, и в зависимости от `ignore_case` проверяет, есть ли значение в `result`. Если `ignore_case == False` то проверяется строка, в которой буквы приведены к нижнему регистру. Если название работы есть в `result`, то вызывается следующий `__next__()`. Если названия в переменной `result` отсутствует, то оно туда добавляется, а итератор возвращает значение в вызывающую функцию.

Менеджер контекста — конструкция, которая позволяет обернуть блок кода и выполнить различные действия на этапе перед выполнением действий с обернутым кодом, так и по завершении, в том числе помогает подавлять исключения, не останавливая выполнение программы. В данном случае, перед выполнением кода менеджер контекста запоминает текущее время в локальную переменную, а по завершении работы блока опять берет текущее время и вычитает из него время, записанное в локальной переменной, получая тем самым время выполнения блока.

Декоратор — функция, которая в своем теле вызывает декорируемую функцию(аргументы декорируемой функции передаются в декоратор). В нашем случае сначала декоратор выводит название декорируемой функции, затем передает ей управление, получает результат ее выполнения, производит с ним действия(в данном случае — вывод на экран значений) и отдает декорируемой функции результат, который та вернет вызывающей функции(тот самый результат выполнения декорируемой функции).