

## Лабораторная работа №7.

### Основные требования:

- каждая функция должна иметь docstring - множ-й комментарий (по типу что делает данная функция)

Пример грамотного оформления функции с docstring + typehints:

```
def add(a: int, b: int) -> int:
    """
    функция для вычисления суммы двух чисел
    :param a: первое целое число
    :param b: второе целое число
    :return: int (целое число)
    """
    return a + b
```

- написание функций должно быть компактным ( не в 100 строчек кода ) - именованя функций должны быть нормальными и четко отражать смысл самой функции.
- код должен соответствовать стандарту языка Python (PEP). - классы и объекты должны соответствовать стандартам языка программирования Python.
- запрещено использование доп. библиотек (os, time, random... )

### Общее задание к Л/Р:

Создать класс "Сотрудник" с атрибутами "имя", "фамилия", "возраст" и методом "расчет заработной платы".

```
class Employee:
    def __init__(self, ..):
        pass
```

**НЕОБХОДИМО ВЫПОЛНИТЬ (РЕАЛИЗОВАТЬ/СОЗДАТЬ) НЕКОТОРЫЕ КОНЦЕПЦИИ ООП ДЛЯ КЛАССА СОТРУДНИК;**

❖ Реализовать концепцию **инкапсуляция**:

- атрибуты класса сделать приватными
- написать ГЕТТЕРЫ/СЕТТЕРЫ

```
def set_firstname(self, new_firstname: str) -> None:  
    self.__firstname = new_firstname
```

```
def get_firstname(self) -> str:  
    return self.__firstname
```

- \*метод расчет зарплаты сделать пустым (pass)

❖ Реализация **наследования**:

- унаследовать классы "Менеджер", "Директор", "Зам.Директора" от базового (родительского) класса "Сотрудник".
- переопределить в каждом классе метод расчета зарплаты.

❖ Реализация **полиморфизма**:

- продемонстрируйте понимание полиморфизма (любым способом - придумайте сами)

например вне класса можно задать функцию **info\_fl** - которая принимает на вход объект класса и выводит красиво имя и фамилию.

```
def info_fl(object):  
    pass
```

### Дополнительное задание (+3 балл):

1) Реализовать консольное меню ( где

- [1] - добавить сотрудника,
- [2] - вывод сотрудника,
- [3] - удалить сотрудника,
- [4] - поиск сотрудника,
- [5] - выход из программы (exit)

)

(возможно потребуется видоизменить класс) **(+2 балла)**

[https://docs.python.org/3/reference/compound\\_stmts.html#match](https://docs.python.org/3/reference/compound_stmts.html#match)

2) Добавить возможность сохранения и загрузки данных о сотрудниках в файл. При запуске программы сначала должно проверяться наличие файла с данными, и если он есть - загружаться последний сохраненный список сотрудников. **(+1 балл).**