## Теоретическая часть

* Понятие о классах в **Python** – объявление классов, свойств классов
* Понятие о ссылке на текущий объект - **self**
* Методы классов
* Некоторые специальные методы классов: конструктор (\_\_init\_\_()), деструктор (\_\_del\_\_()), получение строкового представления объекта класса (\_\_str\_\_())
* Создание объектов классов
* Вызов методов классов
* Доступ к свойствам классов по чтению и записи
* Удаление объектов классов
* Понятие об инкапсуляции в классах Python
* Ограничение области видимости атрибутов класса
* Создание геттеров (аксессоров) и сеттеров (мутаторов) для атрибутов класса
* Создание свойств для атрибутов класса
* Введение в наследование в **Python**
* Синтаксис задания наследования в объявлении класса **Python**
* Понятие о функции **super**() – доступ к базовому классу
* Доступ к полям и свойствам базового класса из производного класса
* Понятие о полиморфизме, переопределение методов базового класса
* Статические методы класса, аннотация @staticmethod, синтаксис вызова метода
* Статический атрибут класса, синтаксис обращения к атрибуту
* Сортировка списка объектов, методы sort(), sorted() – использование параметра key для задания поля сортировки
* Понятие о лямбда-функциях для задания полей сортировки

## Практическая часть

Разработайте консольное приложение Python в составе главного модуля **main.py**, модуля **utils.py** со вспомогательными функциями, модуля **app.py** и дополнительных модулей для функций решения задач, классов.

Используйте исключения. Реализуйте простое меню для выбора решаемой задачи, пункта в решаемой задачи.

**Task1.** Разработайте класс **Plane** со следующими свойствами (созданными при помощи декораторов):

* тип самолета (Ил-76, Boeing-747, …)
* количество пассажирских мест (целое число, от 0 и выше)
* текущее количество пассажиров
* расход горючего за час полета (вещественное число, от 0 и выше)
* количество двигателей (целое число, от 1 до 12)
* название авиакомпании владельца самолета (непустая строка)

В свойствах-сеттерах выбрасывайте исключение при некорректных значениях. Разработайте конструктор **\_\_init\_\_()** и метод формирования строкового представления **\_\_str\_\_()** в виде строки таблицы.

Создайте список самолетов (не менее 10 элементов). Разработайте функции для обработки списка:

* Вывод списка самолетов в виде таблицы
* Увеличение количества пассажиров на введенное с клавиатуры значение
* Удаление выбранного по номеру в списке самолета
* Реализуйте сортировки списка самолетов:
  + По типу самолета
  + По убыванию количества двигателей
  + По названию авиакомпании владельца самолета
  + По убыванию расхода горючего за час полета

**Task2.** Создайте класс **Animal** со свойствами (созданными при помощи декораторов) для хранения:

* клички животного,
* веса (в кг),
* возраста (в полных годах),
* цвета (масть) животного,
* фамилии и инициалов владельца (Иванов И.И., …).

Реализуйте конструктор **\_\_init\_\_()**, свойства, метод **\_\_str\_\_()**, метод вывода данных животного в виде строки таблицы.

Создайте при помощи конструкторов список из 10 животных, выведите список в консоль.

Разработайте функцию, при помощи цикла находящую всех животных, возраст которых больше заданного с клавиатуры. Поместить найденных животных в дополнительный список, удалив животных из исходного списка. Выводите в консоль списки до и после вызова этой функции.

Реализуйте сортировки списка животных:

* По убыванию возраста
* По кличке
* По возрасту и по цвету (сортировка по двум полям)
* По фамилии и инициалам владельца

**Task 3.** Создать иерархию классов:

* Базовый класс Фигура3D со свойством радиус, пустыми методами для вычисления площади и объема (имеются в виду объявления вида **def area(): pass** и **def volume(): pass**)
* Класс Цилиндр, наследует от Фигура3D с методами для вычисления площади и объема
* Класс Конус, наследует от Фигура3D с методами для вычисления площади и объема
* Класс ТрехграннаяПирамида, наследует от Фигура3D с методами для вычисления площади и объема (правильная трехгранная пирамида для упрощения вычислений)
* реализовать по два объекта каждого типа в списке наследников класса Фигура3D, вычислить суммарную площадь фигур, суммарный объем фигур

## Дополнительно

Материалы задания и задачник – в архиве, запись занятия можно скачать[**по этой ссылке**](https://cloud.mail.ru/public/oAha/YsWCPFMEN).