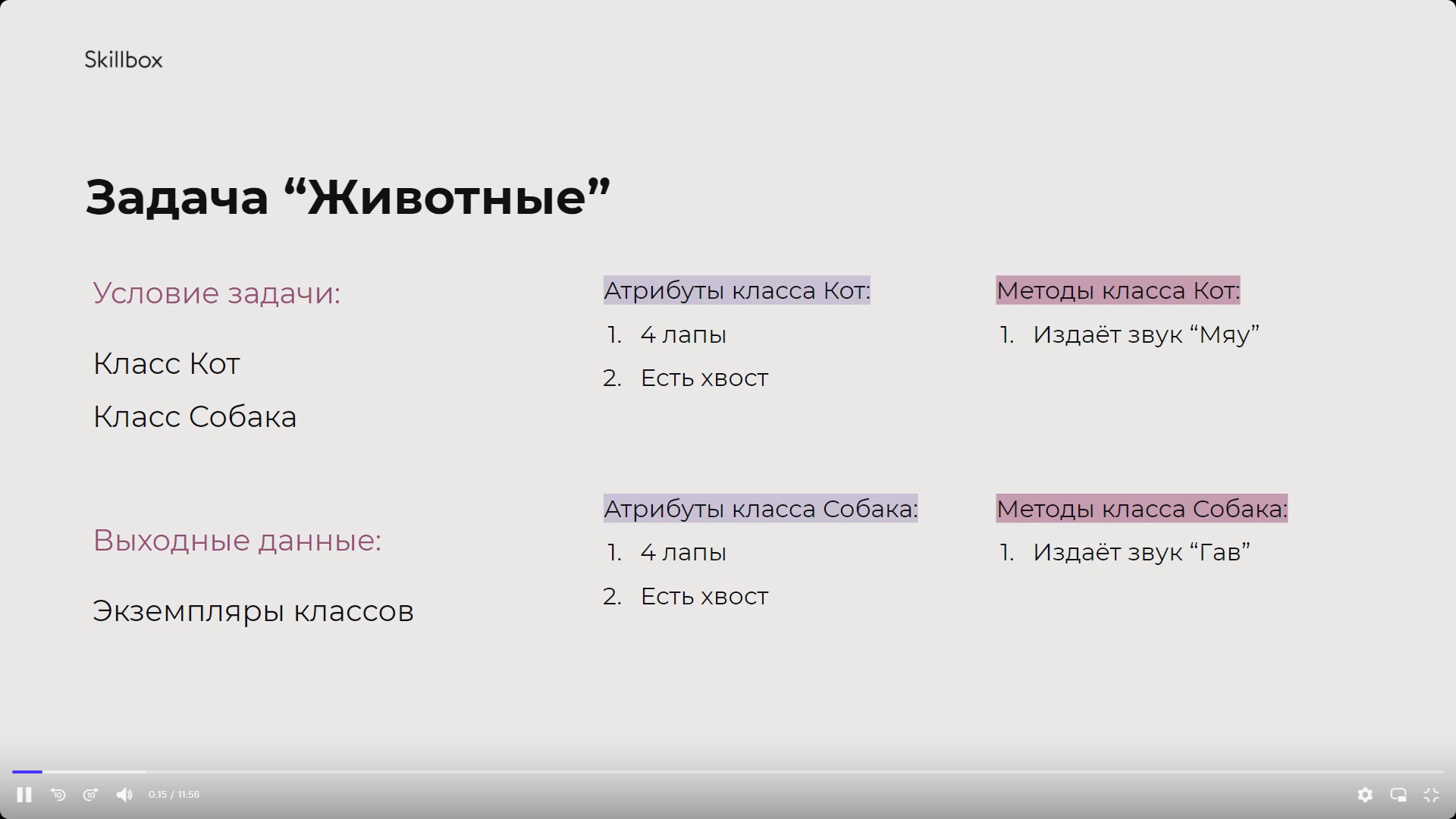
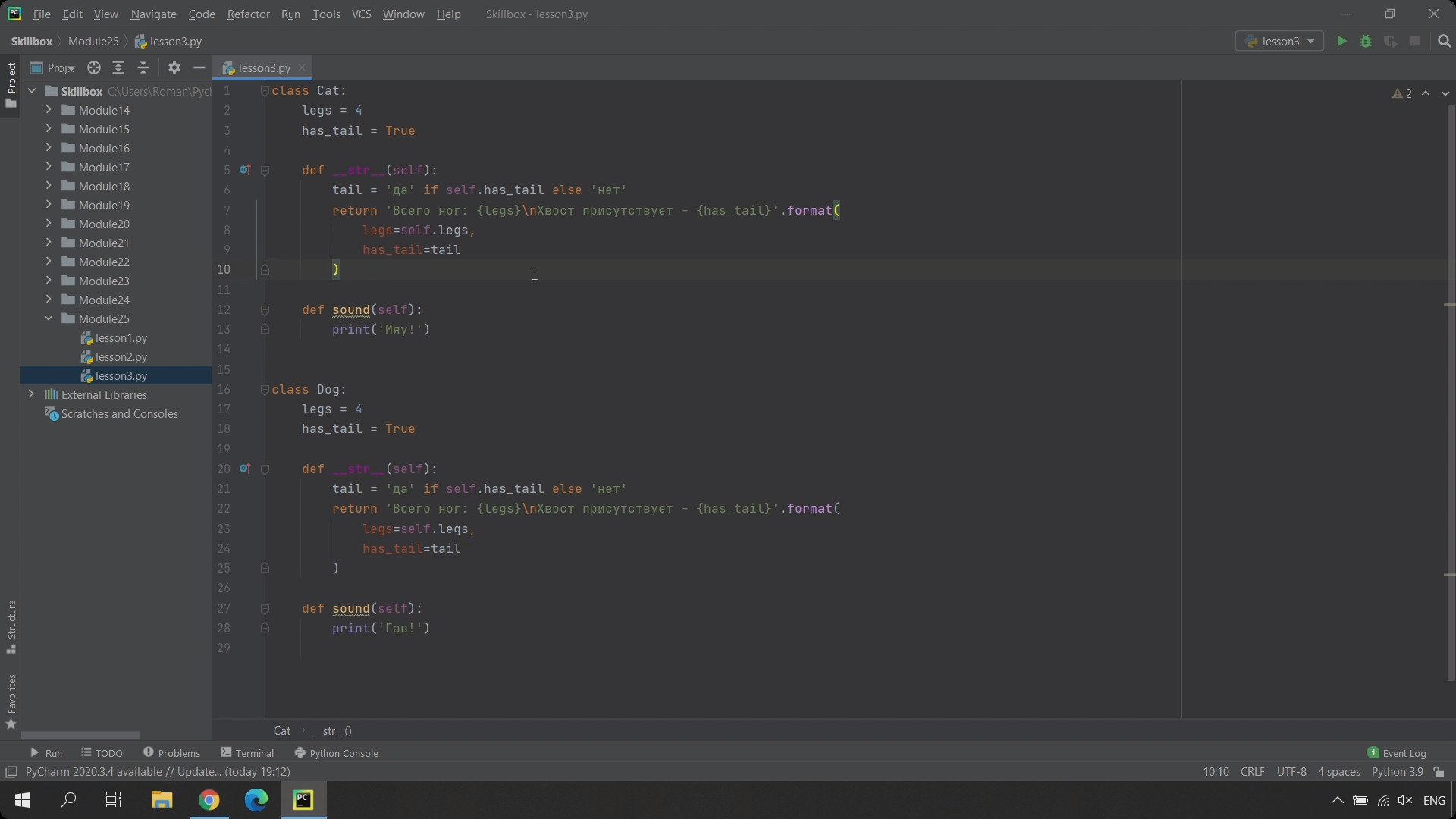
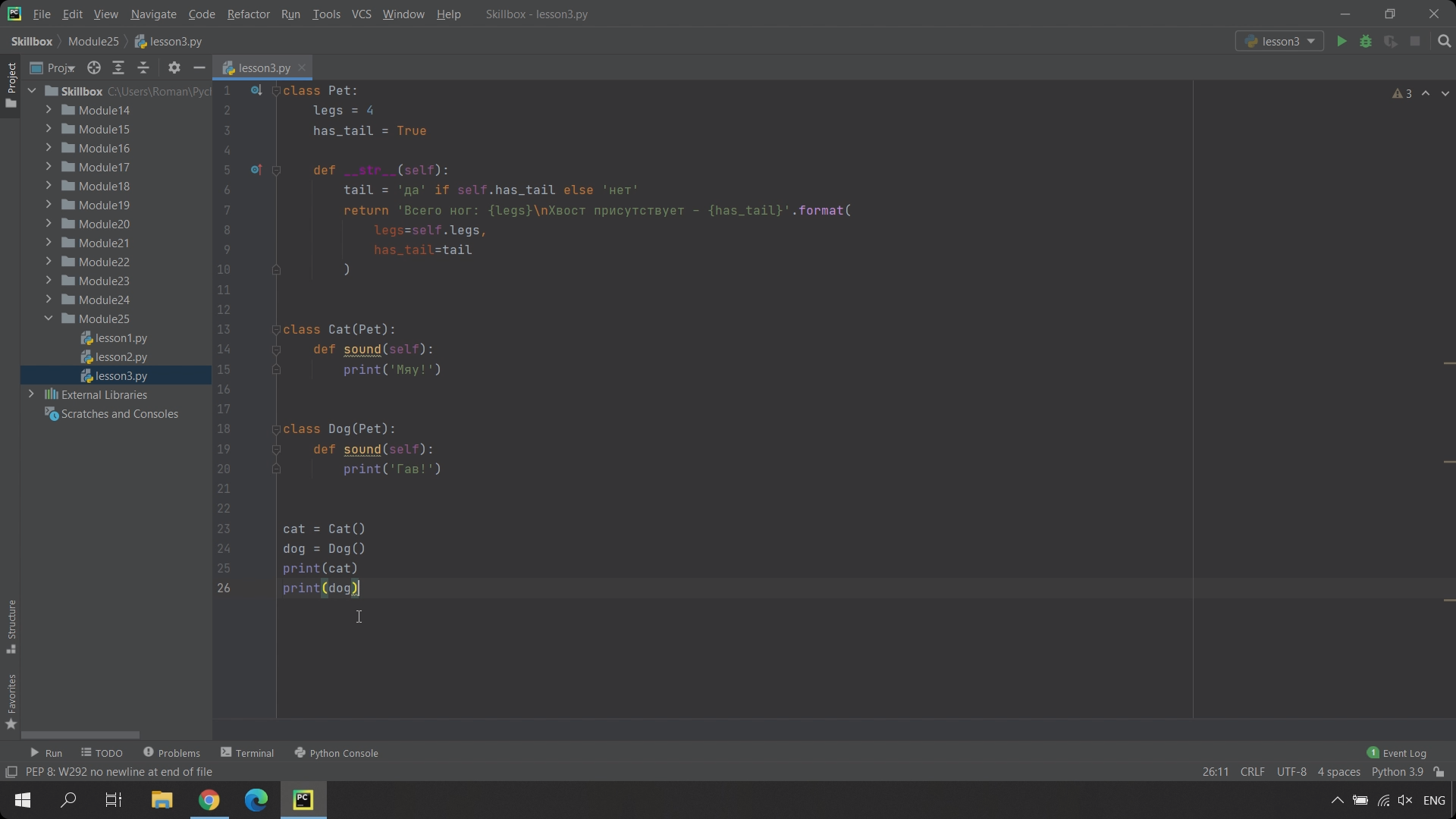
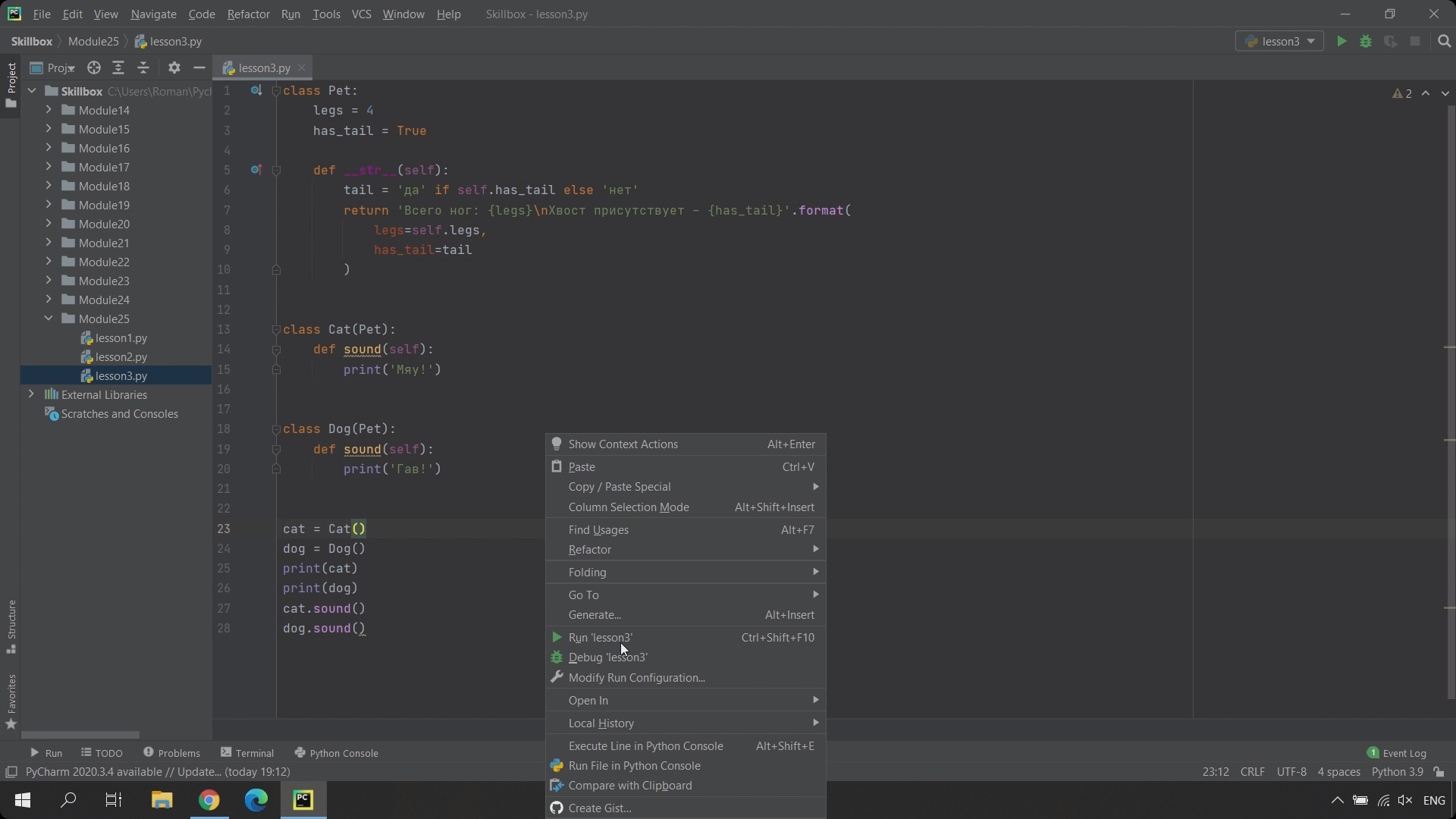
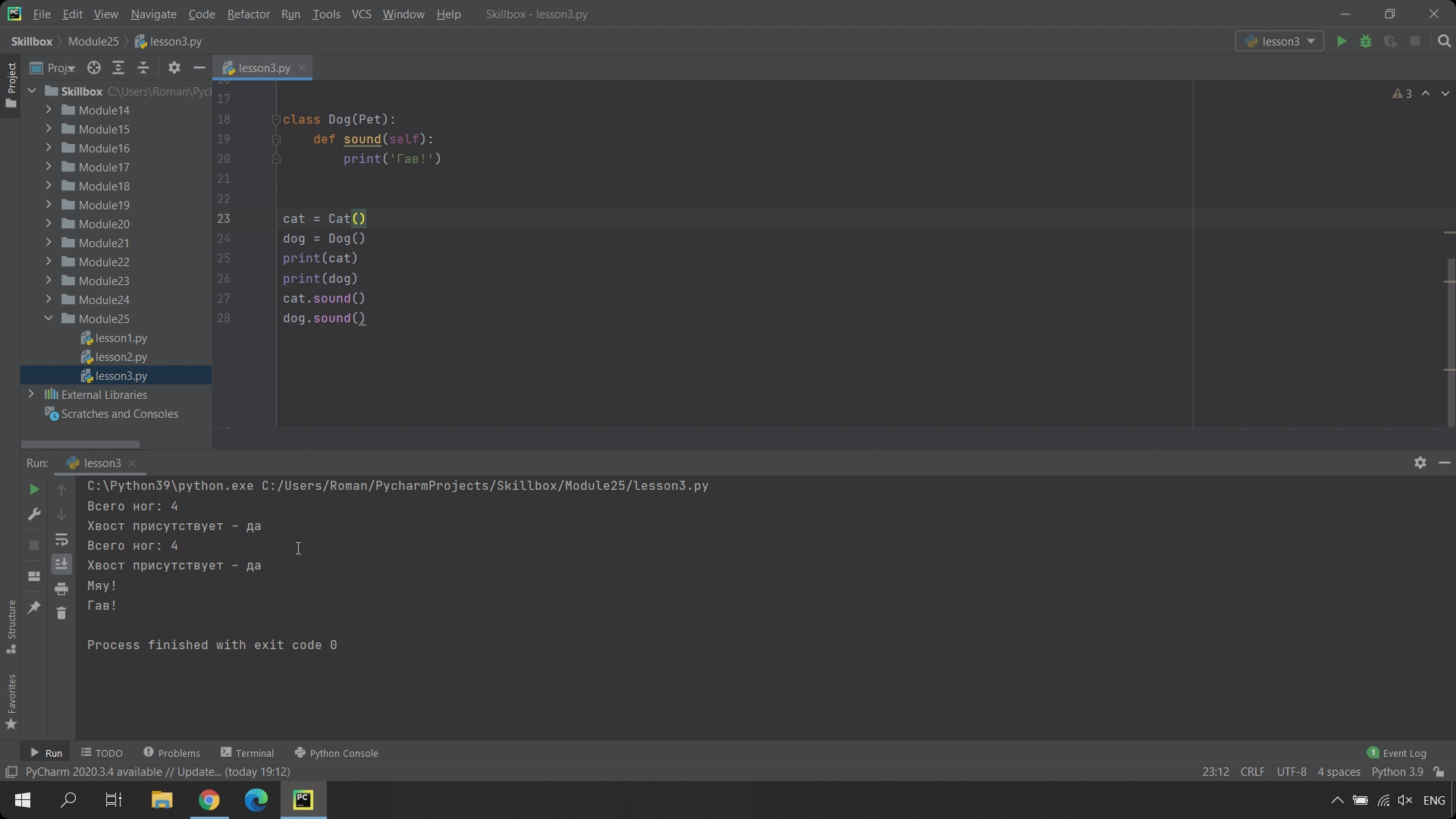
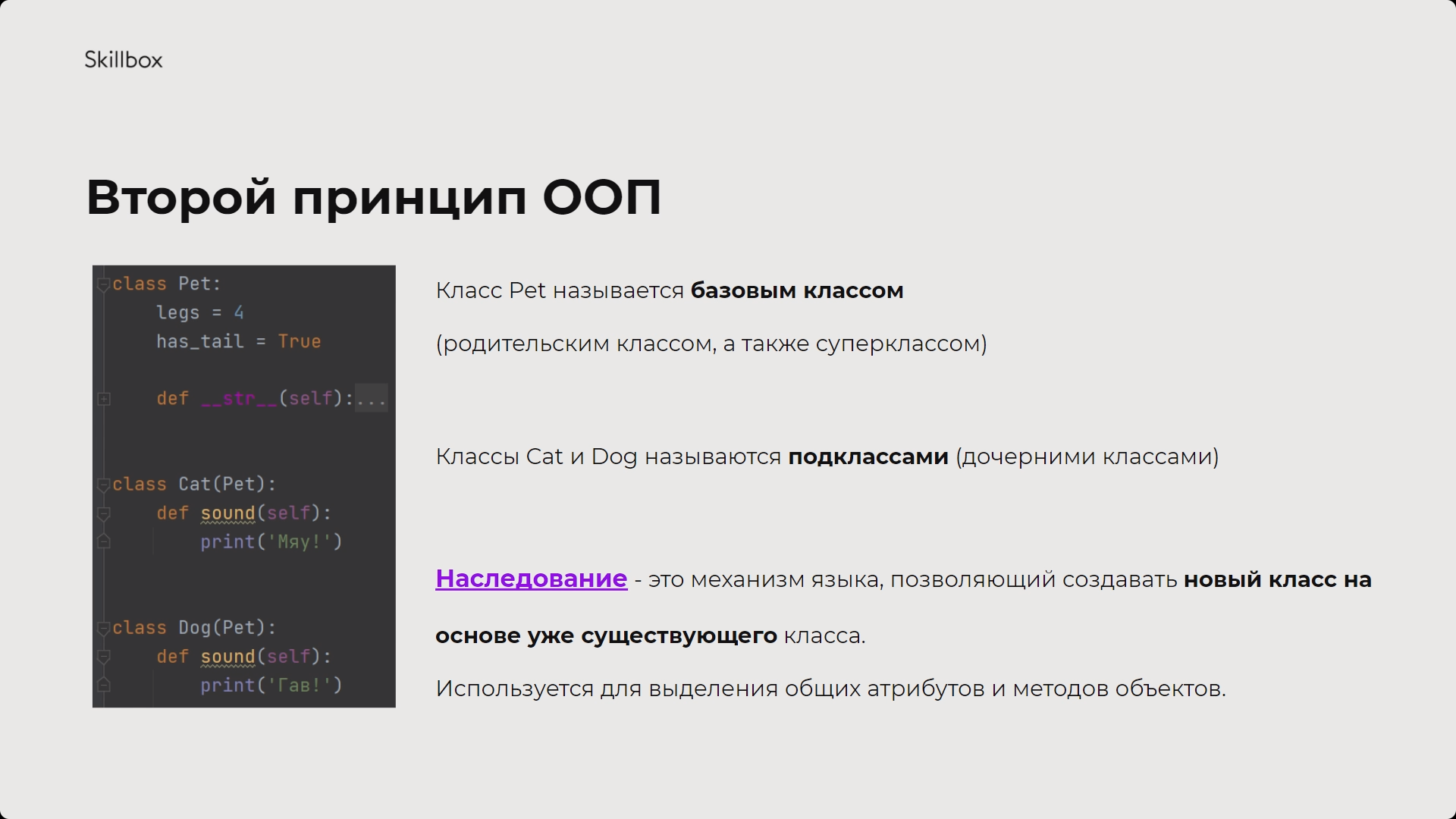
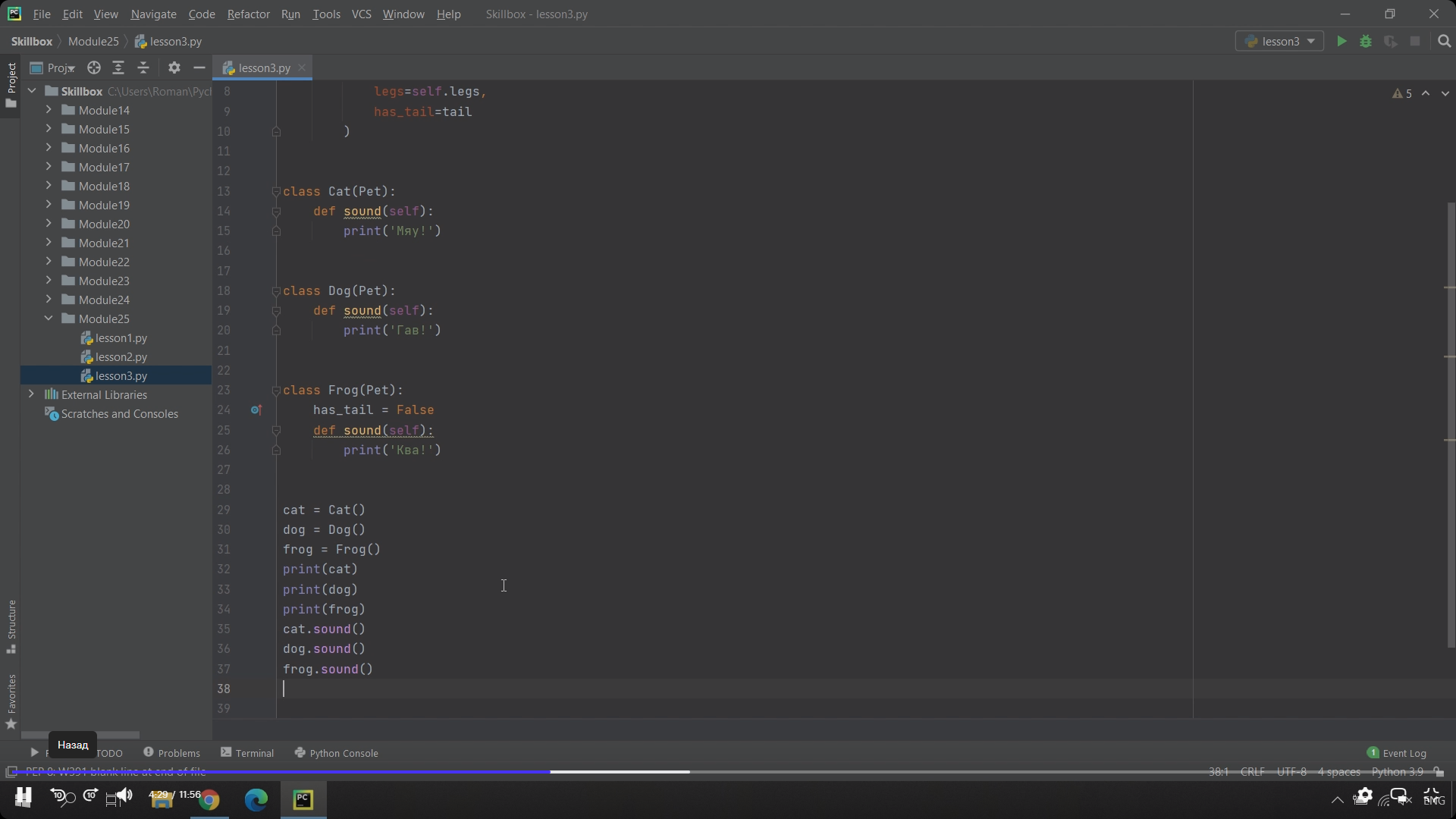
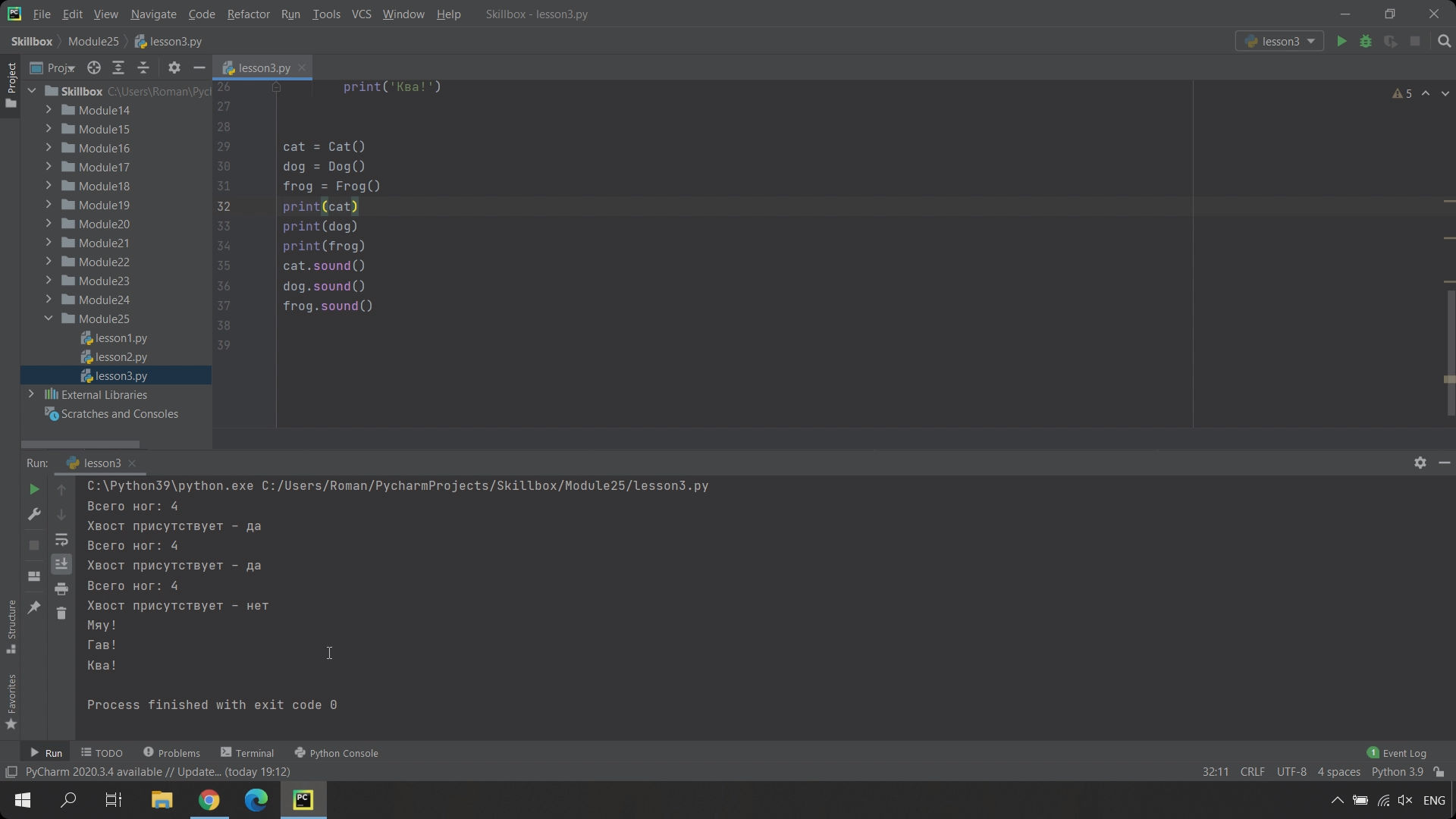
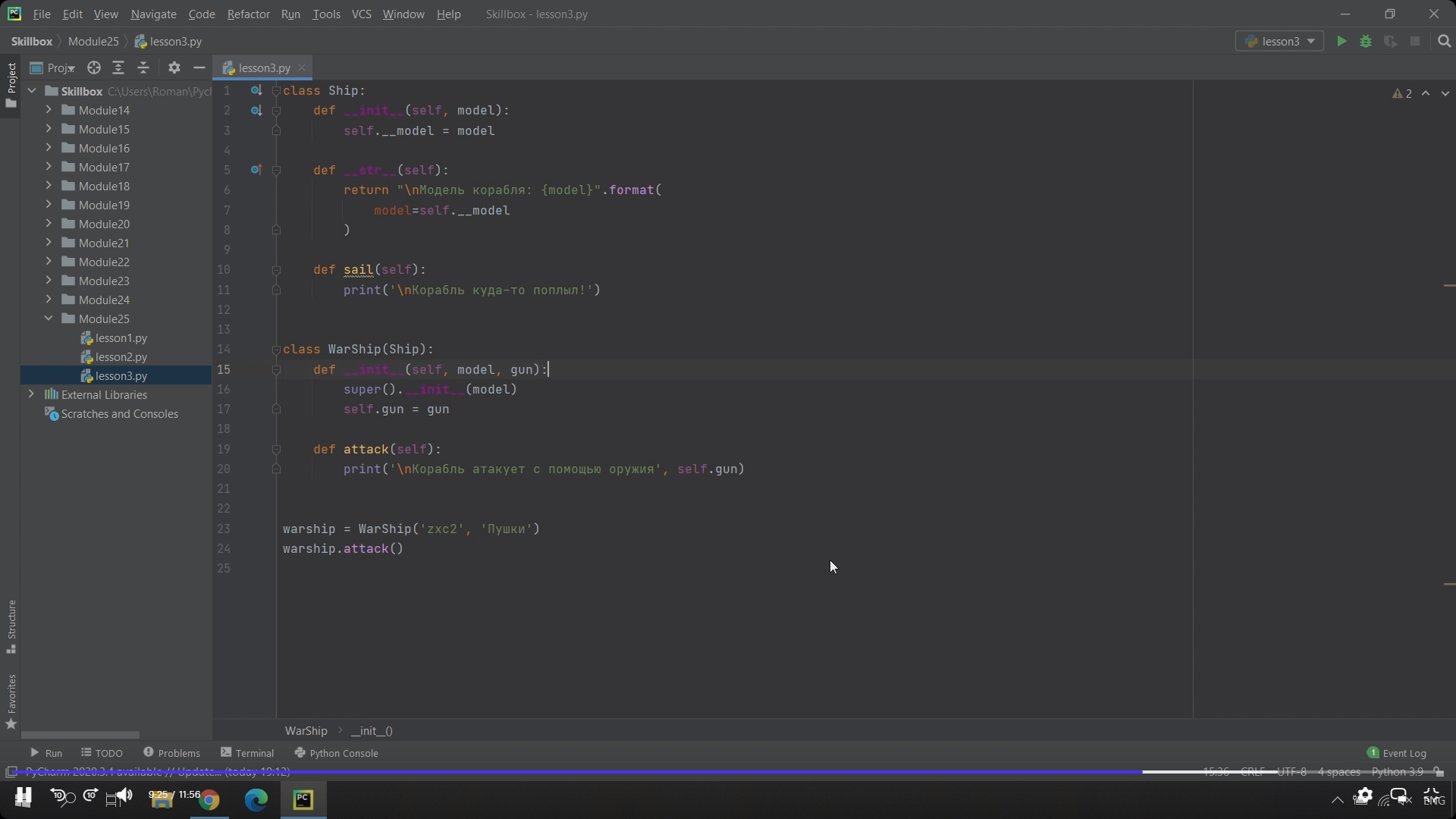
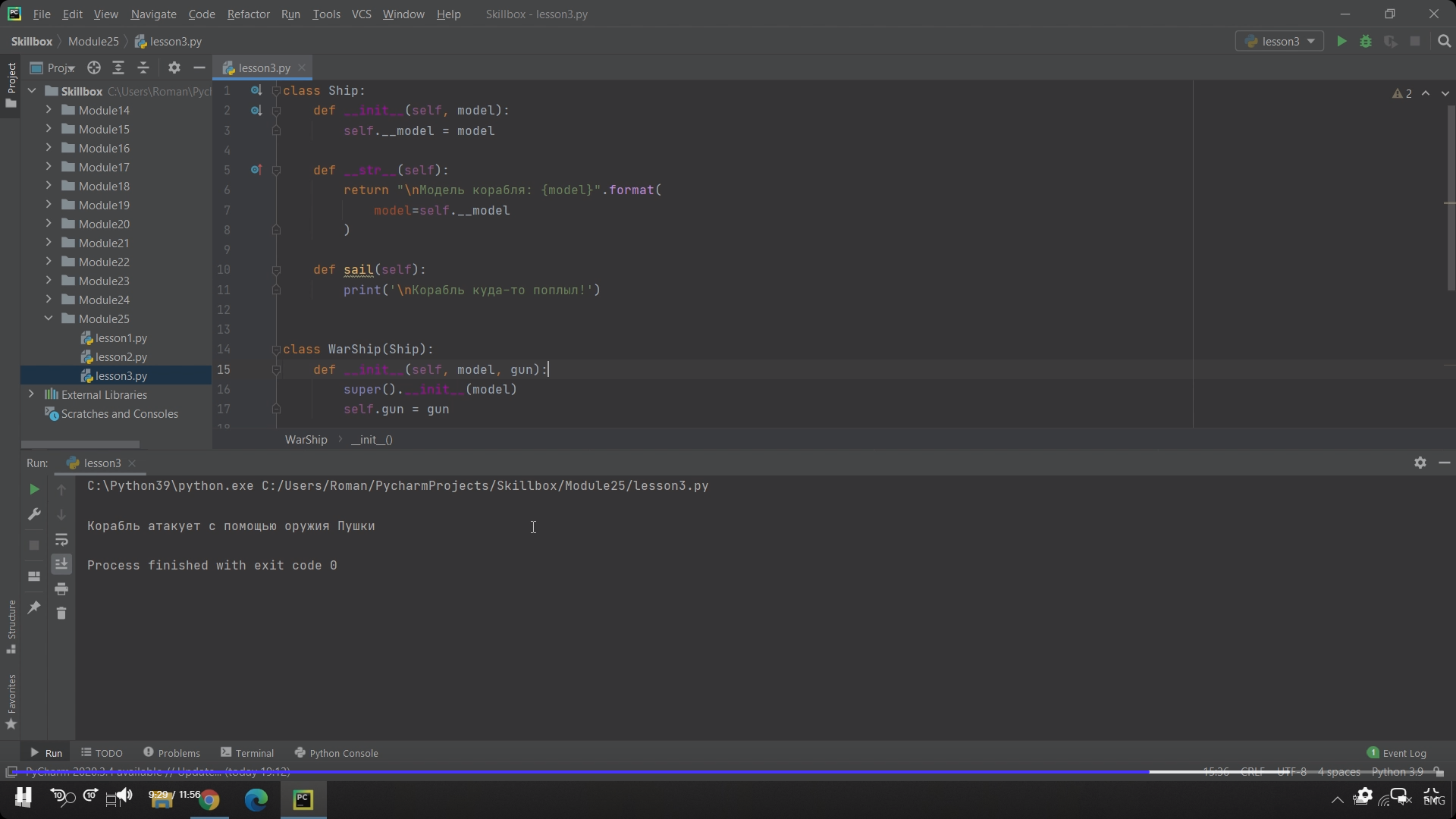
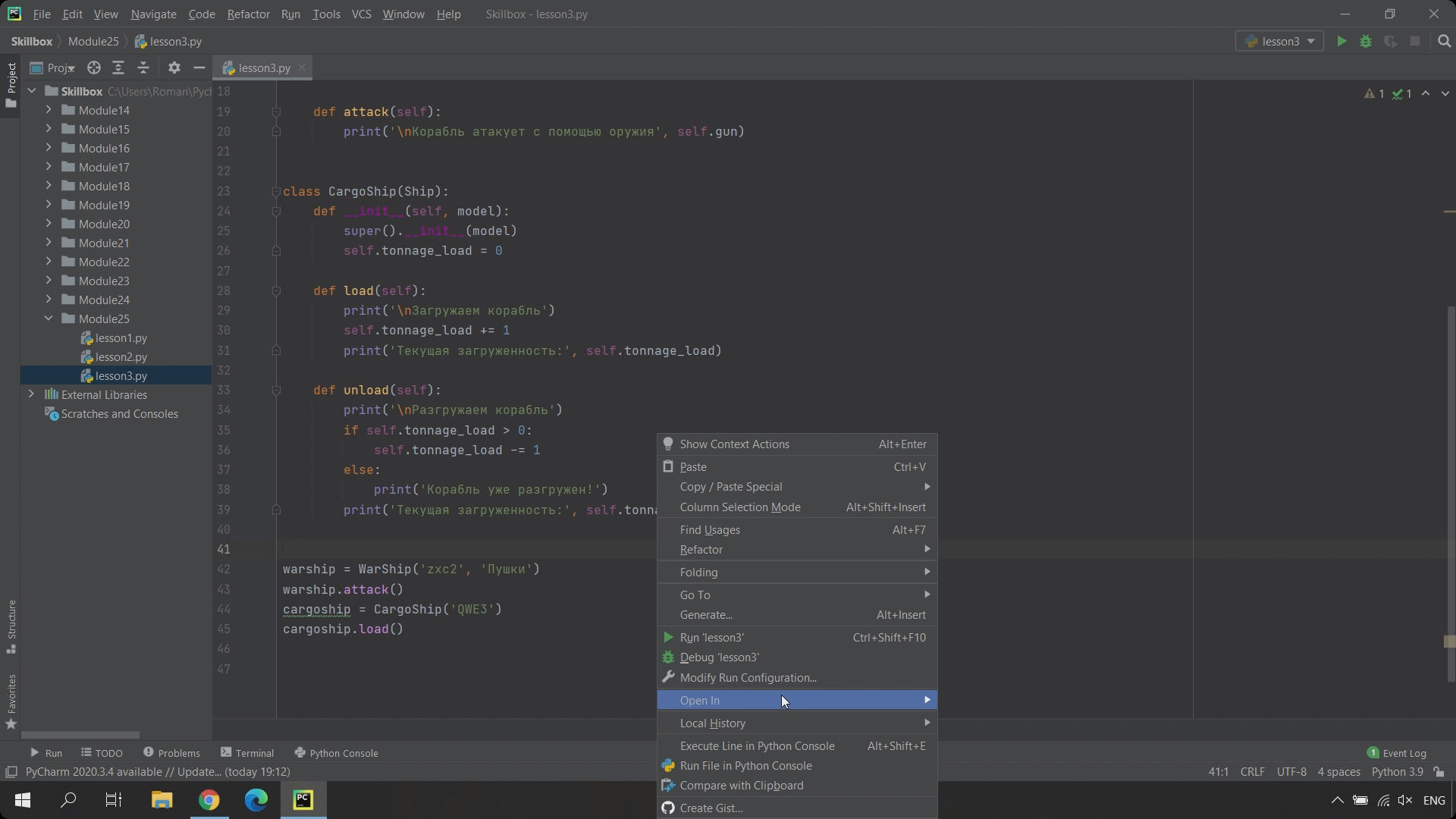
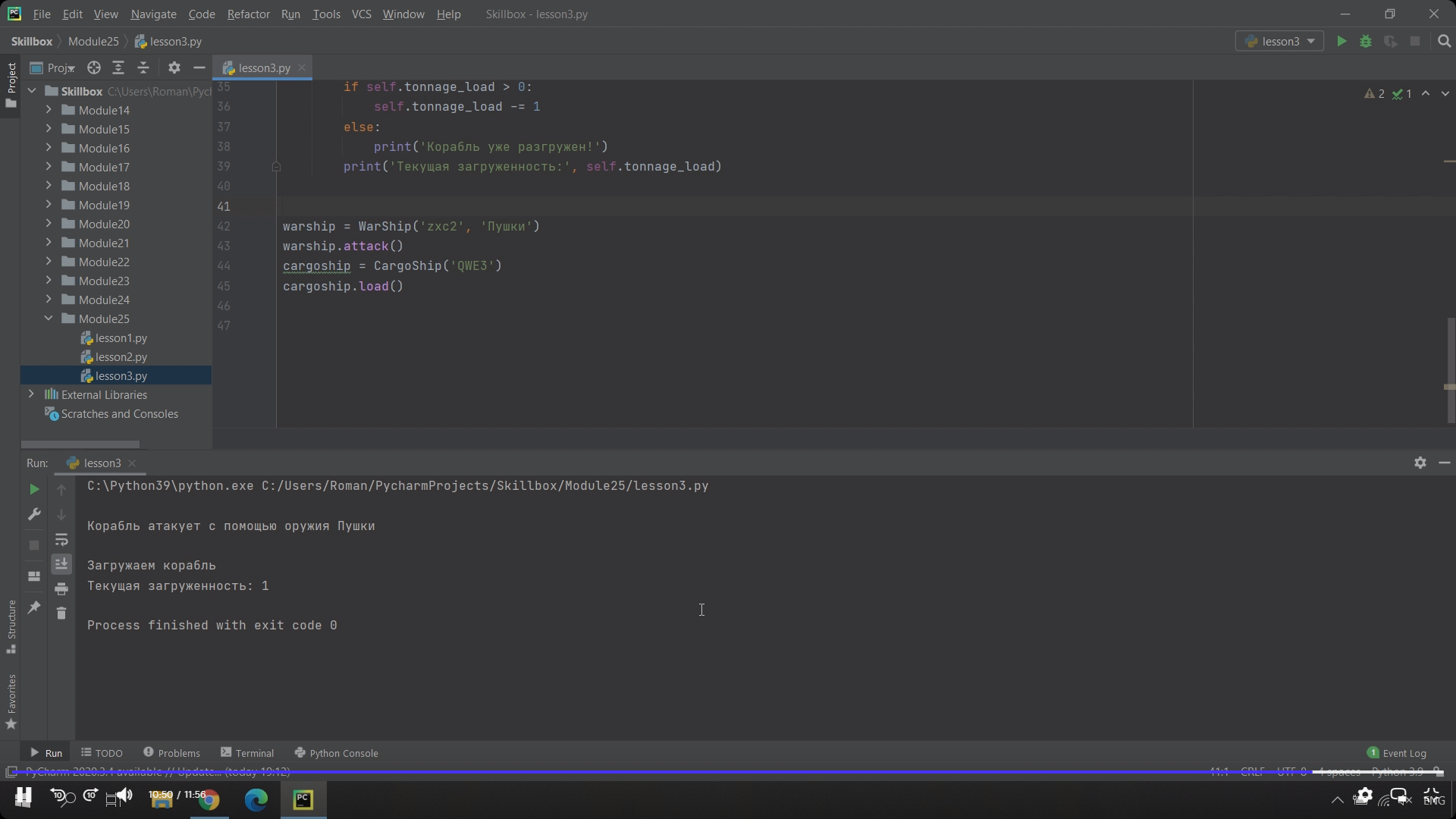
**25.3 Наследование**

#### Практика

##### **Задача 1. Корабли**

Даны два класса кораблей — грузовой и военный. У каждого из этих кораблей есть своя модель, и каждый может сделать два действия: сообщить свою модель и идти по воде.

Грузовой корабль имеет такой атрибут, как заполненность, изначально он равен нулю. У него есть ещё два действия: погрузить и выгрузить груз с корабля.

У военного же корабля нет никаких грузов, есть только оружие, которое передаётся вместе с моделью. Также, вместо погрузки и выгрузки, у него есть другое действие — атаковать.

Реализуйте классы грузового и военного кораблей. Для этого выделите общие атрибуты и методы в отдельный класс «Корабль» и используйте наследование. Не забудьте про функцию super в дочерних классах.

##### **Задача 2. Роботы**

На военную базу завезли несколько интересных моделей роботов, которые похожи между собой, но имеют немного разные функции. У каждого робота есть номер модели и действие operate, которое означает, что делает робот. Особенности роботов следующие:

* У робота-пылесоса есть мешок для мусора, изначально он пустой (0). При команде operate робот сообщает, что он пылесосит пол, и выводит текущую заполняемость мешка.
* У военного робота есть оружие, и при команде operate он выводит сообщение о защите военного объекта с помощью этого оружия.

Ещё есть робот — подводная лодка, который также является военным. У этого робота есть значение глубины, и при команде operate он делает то же, что и военный робот, плюс сообщает, что охрана ведётся под водой.

Напишите программу, которая реализует все необходимые классы роботов.

##### **Задача 3. Кастомные исключения**

Исключения в Python также являются классами, и все они берут свои истоки от самого главного класса — Exception. И для создания своего собственного класса ошибки достаточно написать его дочерний класс. Например:

class MyOwnException(Exception):

    pass

raise MyOwnException('Это моя ошибка!')

Причём содержимое объекта исключения чаще всего так и оставляют (просто pass), чтобы не сломать автоматические обработчики исключений.

Напишите программу, которая считывает из файла numbers.txt пары чисел, делит первое число на второе и выводит ответ на экран. Если первое число меньше второго, то программа выдаёт исключение DivisionError (нельзя делить меньшее на большее).

Дополнительно, с помощью try except, можно обработать исключение на своё усмотрение.