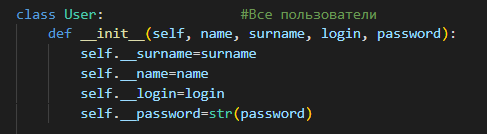
CRM мед.организации, предоставляющей услуги по выполнению анализа биоматериала.

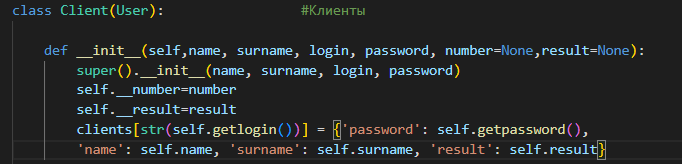
**ОСНОВНЫЕ ТРЕБОВАНИЯ**

В **программе 6 классов**, описанных в документе *classes.py*, 4 из них **наследуются** от оставшихся:

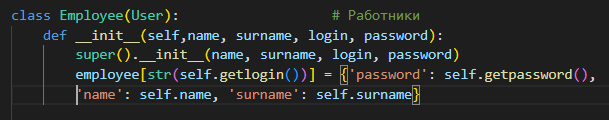
1. User. Описывает всех пользователей программы.



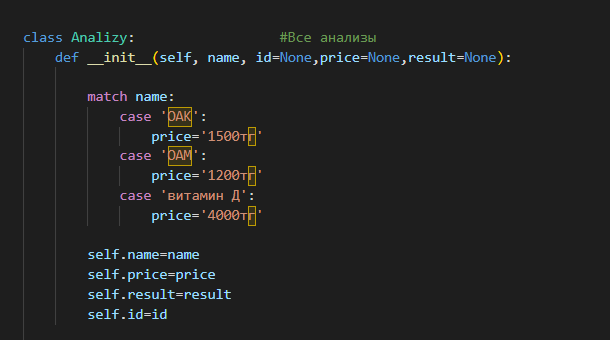
1. Client. Наследуется от класса User, описывает профили клиентов организации.



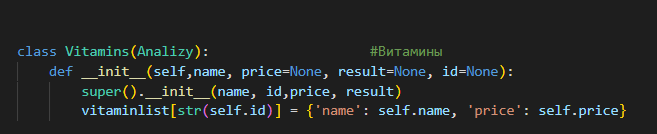
1. Employee. Наследуется от класса User, описывает профили работников организации.



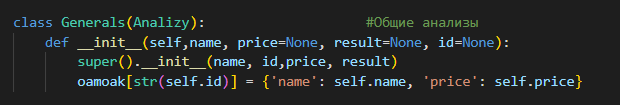
1. Analizy. Описывает виды доступных анализов.



1. Vitamins. Наследуется от класса Analizy, описывает анализы на витамины.

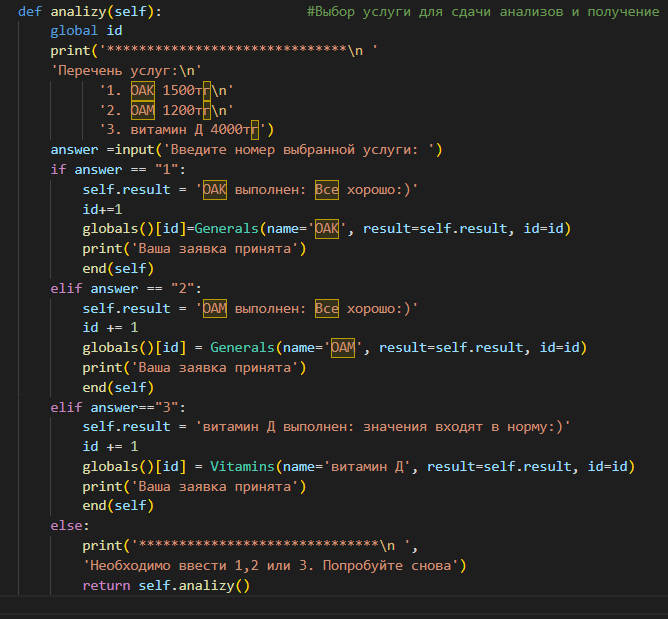


1. Generals. Наследуется от класса Analizy. Описывает общие анализы.

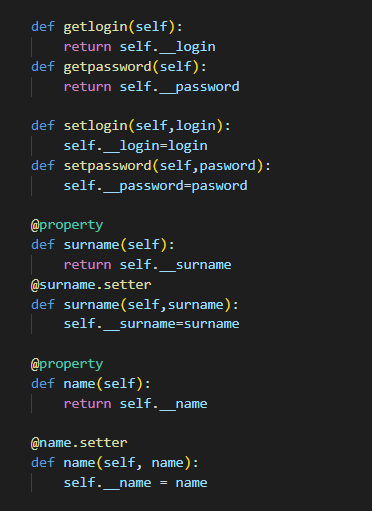


Пример **использования композиции** в коде (*classes.py*):

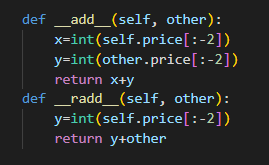
Метод класса Client

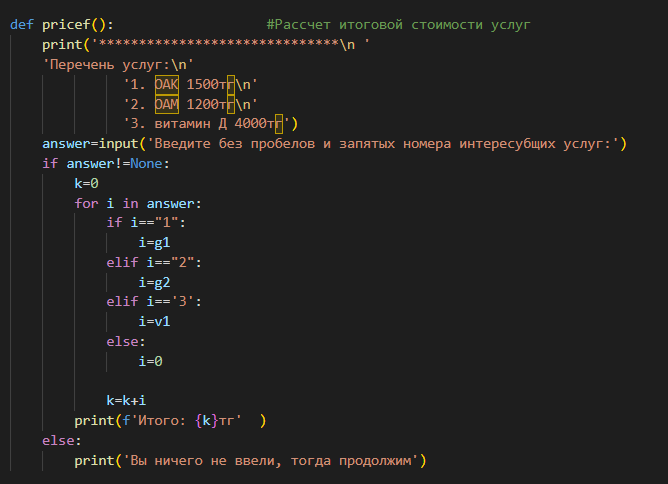


Пример использования **инкапсуляции** и **Computed** свойств в классе User (*classes.py*):



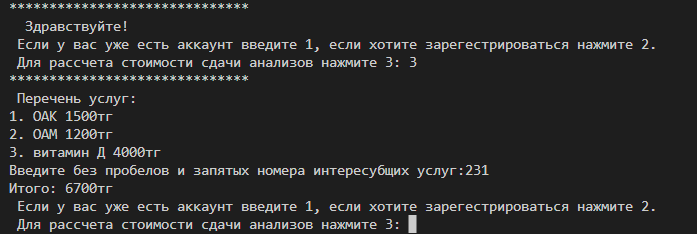
Описание перегрузки операторов в классе Analizy (*classes.py*) и их применение в функции pricef (*main.py*):





**РАБОТА ПРОГРАММЫ**

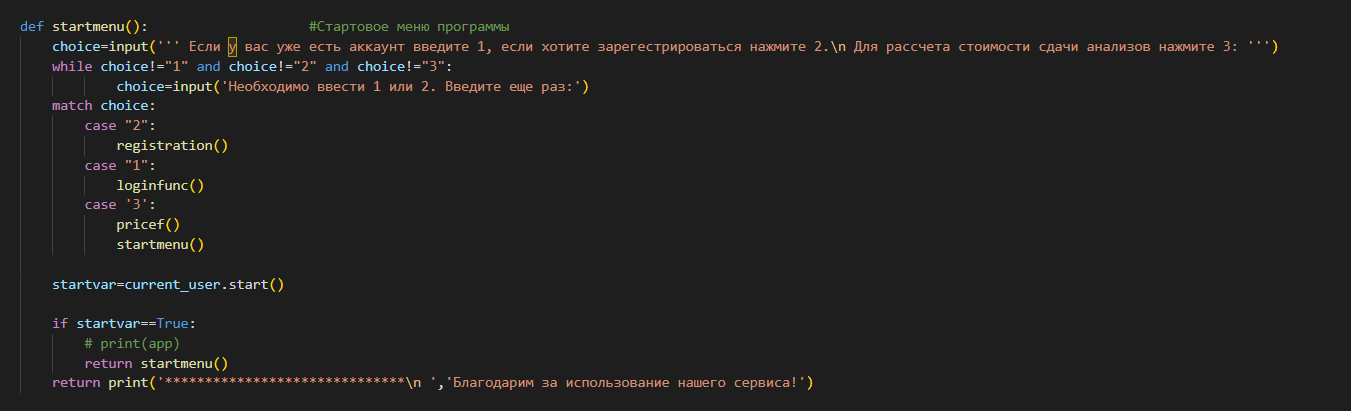
Программа начинается с приветствия и выбора дальнейших действий:



Всего 3 варианта:

1. Совершить вход в свой аккаунт, если он существует
2. Зарегистрировать новый аккаунт
3. **Без входа рассчитать стоимость планируемых к сдаче анализов**

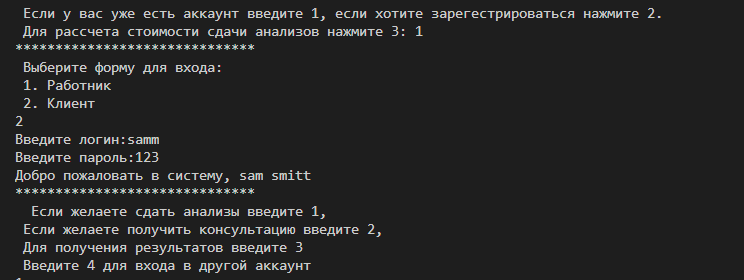
Тело программы описано в функции startmenu (main.py):



current\_user – хранит объект пользователя, чтобы запустить метод класса для дальнейших действий. В startvar записывается результат выполнения метода start текущего пользователя (True or Faulse), влияет на продолжение работы в сервисе или выход из него.

В результате выбора номера 3, выведены доступные услуги и предложена возможность рассчитать стоимость введенных услуг с помощью функции pricef (main.py). После программа возвращает результат и первоначальное меню.

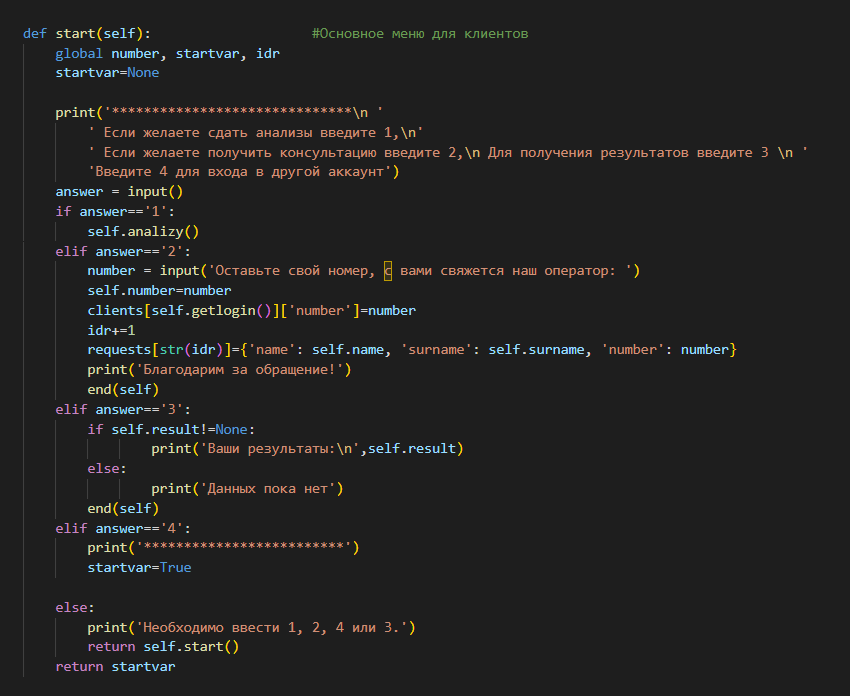
**При вводе числа 1** вызывается функция loginfunc (main.py).



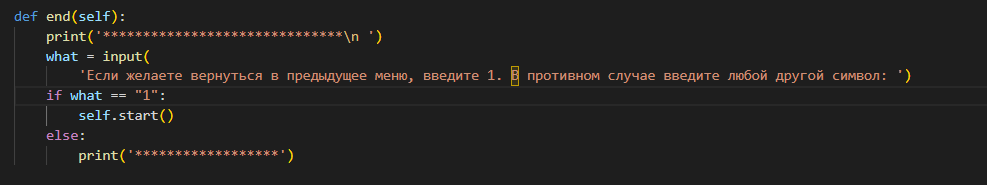


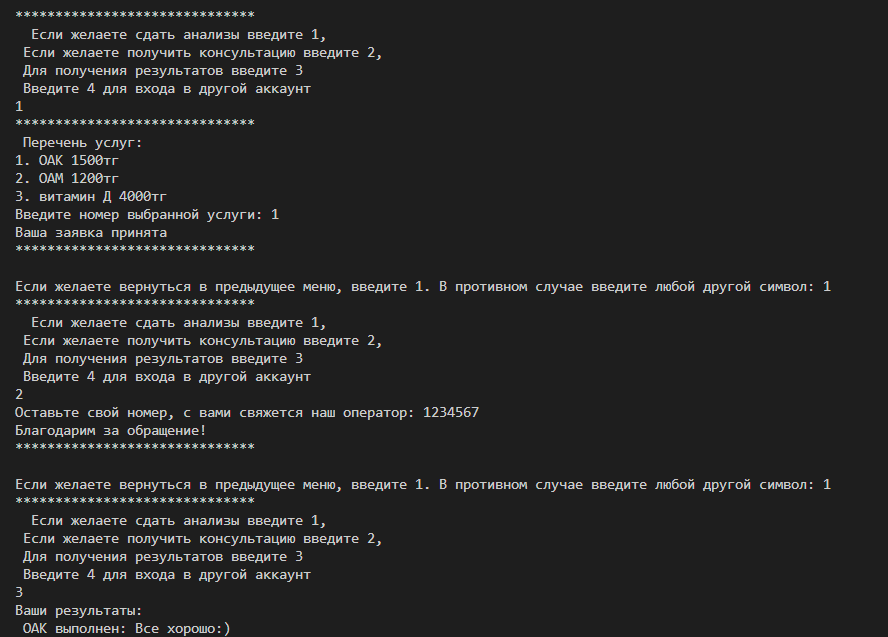
Функция входа принимает значение who, определяющее в каком списке будет вестись поиск пользователя: в работниках или в клиентах (при создании объекта его данные сразу записываются в один из этих списков). Если логин пользователя существует, переменная login, хранящая его текстовое значение, преобразуется в объект класса lgn.

Далее мы сверяем пароль, введенный с клавиатуры, с паролем в аргументе объекта и выходим из функции, переходя в методу класса start, описывающему индивидуальное меню пользователя, в данном случае клиента.

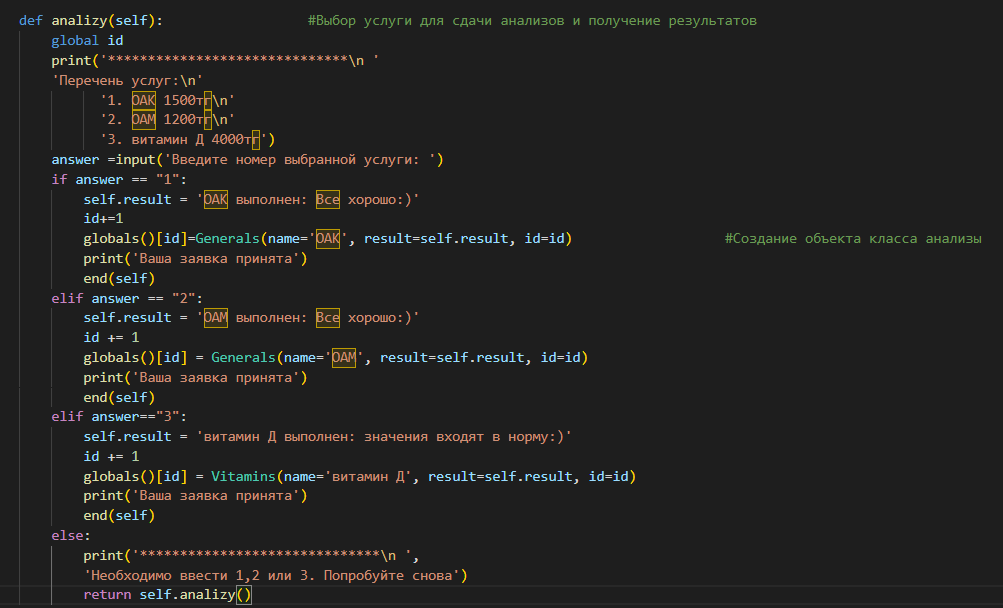


После выполнения действия всегда можно вернуться в это стартовое меню с помощью функции end, либо завершить работу с сервисом.



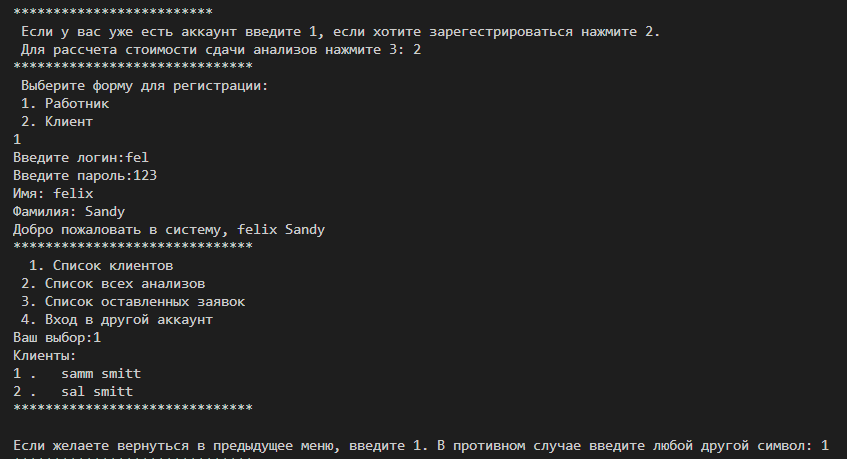


Выбранная услуга автоматически создает объект класса Analizy с помощью одноименного метода, записывает данные в соответствующий список для хранения.

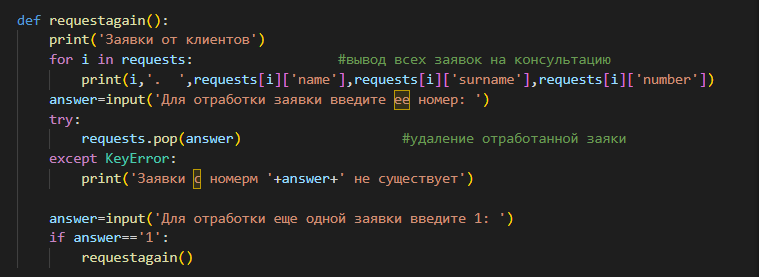


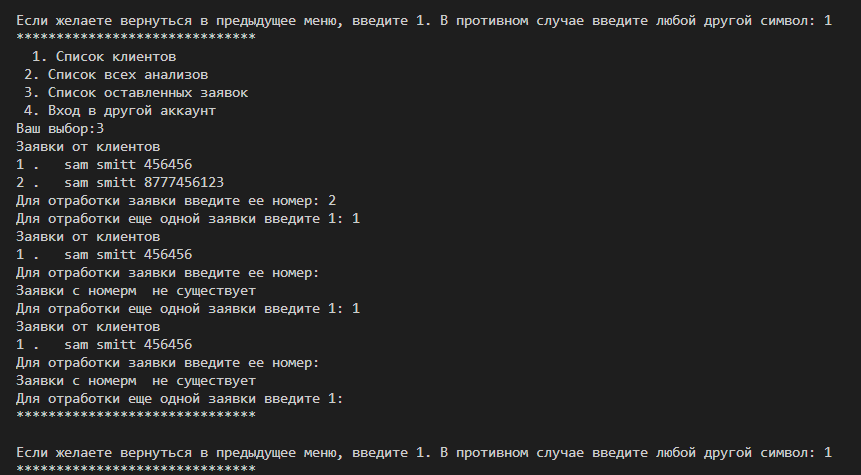
При вводе числа 4, программа возвращает стартовое меню.

При **выборе в стартовом меню второй опции**, запускается функция registration.



Функция позволяет создать объект нужного класса и продолжить работать под новым профилем. В меню для работников можно выбрать вывод всех клиентов, список всех заказанных анализов, отработать заявки с помощью функции requestagain, которая позволяет убирать из списка определенные строки. Вернуть в меня можно ответив на полученные опции нажатием enter.





При вводе любого символа, кроме 1, программа завершается.

