

Lab_6_1. Необходимо создать класс **Building**. Важно, что в нашем классе за одним этажом может быть закреплена только одна компания

Для этого в классе **Building** должно быть реализованы

1. метод `__init__`, который принимает количество этажей в здании
2. метод `__setitem__`, который закрепляет за определенным этажом компанию. Если этаж был занят другой компанией, нужно заменить название другой компанией
3. метод `__getitem__`, который возвращает название компании с этого этажа. В случае, если этаж пустует, следует вернуть `None`
4. метод `__delitem__`, который высвобождает этаж

В этом задании вы сами решаете какие атрибуты создавать внутри класса, главное реализовать магические методы из списка выше

```
iron_building = Building(12) # Создаем здание с 12 этажами
iron_building[0] = 'Morgan Stanley'
iron_building[2] = 'IBM'
iron_building[4] = 'Boeing'
iron_building[6] = 'Amazon.com'
iron_building[10] = 'Intel'
print(iron_building[4])
del iron_building[4]
print(iron_building[4])
```

В конце программы написать код, который будет выводить:

1. Непустой этаж состоящий из: Этаж - Название;
2. Свободные этажи.

Lab_6_2. Создайте класс **Addition**, у которого необходимо:

- переопределить метод `__call__`. Он должен принимать произвольное количество аргументов и среди этих аргументов находить числа и их суммировать. Все остальные типы данных необходимо пропускать. В результате метод `__call__` должен распечатать ответ в следующем виде: "Сумма переданных значений = {сумма}"

```
add = Addition()
add(10, 20) # печатает "Сумма переданных значений = 30"
add(1, 2, 3.4) # печатает "Сумма переданных значений = 6.4"
add(1, 2, 'hello', [1, 2], 3) # печатает "Сумма переданных значений = 6"
```

Lab_6_3. Для выражения относительной силы шахматистов используется система рейтингов. Наиболее популярная система рейтингов, которая используется Международной шахматной федерацией (ФИДЕ), большинством других шахматных федераций и игровых шахматных сайтов, является система рейтингов Эло.

В зависимости от выступлений на различных соревнованиях каждому шахматисту

начисляются баллы в его рейтинг. Давайте с вами реализуем класс **ChessPlayer** и научимся сравнивать рейтинги шахматистов между собой.

И так, ваша задача реализовать класс **ChessPlayer**, который состоит из:

- метода инициализации, принимающего аргументы name, surname, rating;
- магического метода `__eq__`, который будет позволять сравнивать экземпляры класса **ChessPlayer** с числами и другими экземплярами этого класса. Если сравнение происходит с целым числом и атрибут **rating** с ним совпадает, то необходимо вернуть **True**, в противном случае - **False**. Если же сравнение идет с другим шахматистом(экземпляром класса **ChessPlayer**) и значения атрибутов **rating** равны, то возвращается **True**, в противном случае - **False**. А если же сравнивается с другим типом данных, верните 'Невозможно выполнить сравнение';
- магического метода `__gt__`. Если сравнение происходит с целым числом и атрибут **rating** больше его, необходимо вернуть значение **True**, в противном же случае - **False**. Если сравнение происходит с другим шахматистом(экземпляром класса **ChessPlayer**) и атрибут **rating** у нашего экземпляра больше, то верните **True**, в противном случае - **False**. В случае если сравнение идет с остальными типами данных, верните 'Невозможно выполнить сравнение'
- магического метода `__lt__`. Если сравнение происходит с целым числом и атрибут **rating** меньше его, необходимо вернуть значение **True**, в противном же случае - **False**. Если сравнение происходит с другим шахматистом(экземпляром класса **ChessPlayer**) и атрибут **rating** у нашего экземпляра меньше, то верните **True**, в противном случае - **False**. В случае если сравнение идет с остальными типами данных, верните 'Невозможно выполнить сравнение'.

```
magnus = ChessPlayer('Carlsen', 'Magnus', 2847)
ian = ChessPlayer('Ian', 'Nepomniachtchi', 2789)
print(magnus == 4000) # False
print(ian == 2789) # True
print(magnus == ian) # False
print(magnus > ian) # True
print(magnus < ian) # False
print(magnus < [1, 2]) # печатает "Невозможно выполнить сравнение"
```