

## Практическое задание.

1. Создайте класс «Отель»
2. Атрибут класса отель – его название.
3. В конструкторе класса, в виде словаря укажите его комнаты.

```
rooms = {100: True, 101: True, 102: True, 103: True,  
         200: True, 201: True, 202: True, 203: True}
```

Не забудьте self. Ключ словаря – номер комнаты, значение – True (в будущем, False будет означать занятость номера).

4. Метод для вывода названия отеля и всех номеров. Используйте цикл for в виде:

```
for переменная_1, переменная_2 в словаре:  
    если переменная_2 == True:  
        вывод "Номер переменная_1 свободен"  
    иначе  
        вывод "Номер переменная_1 занят"
```

Переменная\_1 – ключи словаря, переменная\_2 – его значение.

### Тест метода:

Входные данные:

```
hotel = Hotel("Отель: У Олега")  
hotel.print_rooms()
```

Выходные данные:

```
Отель: У Олега  
Номер № 100 свободен  
Номер № 101 свободен  
Номер № 102 свободен  
Номер № 103 свободен  
Номер № 200 свободен  
Номер № 201 свободен  
Номер № 202 свободен  
Номер № 203 свободен
```

5. Метод для брони номера. Метод принимает значение номера отеля и меняет значение ключа на False. Если номер уже занят или был введен некорректно, необходимо вывести советующее сообщение.

Изменить значение ключа:

```
словарь[ключ] = новое_значение
```

Также можно проверить его значение:

```
if словарь[ключ] == значение ...
```

## Тест метода:

Входные данные 1. Некорректный ввод номера:

```
hotel = Hotel("Отель: У Олега")
hotel.book_room_by_number(111)
```

Выходные данные:

```
Ошибка! Такого номера нет.
```

Входные данные 2. Успешная бронь:

```
hotel = Hotel("Отель: У Олега")
hotel.book_room_by_number(200)
hotel.book_room_by_number(103)
hotel.print_rooms()
```

Выходные данные:

```
Отель: У Олега
Номер № 100 свободен
Номер № 101 свободен
Номер № 102 свободен
Номер № 103 занят
Номер № 200 занят
Номер № 201 свободен
Номер № 202 свободен
Номер № 203 свободен
```

Входные данные 2. Попытка забронировать номер дважды:

```
hotel = Hotel("Отель: У Олега")
hotel.book_room_by_number(200)
hotel.book_room_by_number(103)
hotel.book_room_by_number(103)
hotel.print_rooms()
```

Выходные данные:

```
Номер 103 уже забронирован.
Отель: У Олега
Номер № 100 свободен
Номер № 101 свободен
Номер № 102 свободен
Номер № 103 занят
Номер № 200 занят
Номер № 201 свободен
Номер № 202 свободен
Номер № 203 свободен
```

**6. Метод для завершения брони номера.** Код практически такой же как и у метода брони, только значение ключа меняется на True. Ошибки обрабатываются точно такие же.

## Тест метода:

Входные данные 1. Успешная отмена:

```
hotel = Hotel("Отель: У Олега")
hotel.book_room_by_number(200)
hotel.book_room_by_number(103)
hotel.print_rooms()
print() # Для отступа
hotel.finish_number_by_number(200)
hotel.print_rooms()
```

Выходные данные:

```
Отель: У Олега
Номер № 100 свободен
Номер № 101 свободен
Номер № 102 свободен
Номер № 103 занят
Номер № 200 занят
Номер № 201 свободен
Номер № 202 свободен
Номер № 203 свободен
```

```
Отель: У Олега
Номер № 100 свободен
Номер № 101 свободен
Номер № 102 свободен
Номер № 103 занят
Номер № 200 свободен
Номер № 201 свободен
Номер № 202 свободен
Номер № 203 свободен
```

Входные данные 2. Ошибки:

```
hotel = Hotel("Отель: У Олега")
hotel.book_room_by_number(200)
hotel.finish_number_by_number(123)
hotel.finish_number_by_number(203)
```

Выходные данные:

```
Ошибка! Такого номера нет.
Номер 203 не забронирован.
```

7. Протестируйте программу. Создайте экземпляр класса, убедитесь, что все методы работают исправно и обрабатывают все ошибки. Напишите документацию для класса и для каждого метода:

```
"""Класс для бронирования и снятия брони номеров отеля 🏠"""
```

Напишите её максимально полной и вызовите (метод `__doc__`).

8. Реализуйте меню. Под классом создайте циклическую конструкцию вида:

```
"[1] - Вывести занятость всех номеров\n"[2] - Занять номер\n"[3] - Снять бронь с номера\n"[4] - Выход"
```

Структура:

```
бесконечный while:
    переменная_ввода (input)
    если переменная_ввода == '1':
        вызвать метод для показа комнат
    если переменная_ввода == '2':
        переменная_ввода_2 (input)
        проверка введённого значения на число (try, except)
            если не число - ошибка
        иначе
            вызвать метод для брони комнат со значением переменная_ввода_2
    если переменная_ввода == '3':
        переменная_ввода_2 (input)
        проверка введённого значения на число (try, except)
            если не число - ошибка
        иначе
            вызвать метод для отмены брони со значением переменная_ввода_2
    если переменная_ввода == '4':
        остановить всё
    иначе:
        Такого варианта нет
```

Используйте дополнительные переменные для управления классом и циклической конструкцией (меню). Обработайте все возможные ошибки (смотрите входные/выходные данные).

**Тест:**

Входные/выходные данные 1. Исправная работа:

```
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 2
Введите номер комнаты для брони.
> 101
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 2
Введите номер комнаты для брони.
> 102
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
```

```
[4] - Выход
> 3
Введите номер комнаты для отмены брони.
> 101
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 1
Отель: У Олега
Номер № 100 свободен
Номер № 101 свободен
Номер № 102 занят
Номер № 103 свободен
Номер № 200 свободен
Номер № 201 свободен
Номер № 202 свободен
Номер № 203 свободен
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 4
```

## Входные/выходные данные 2. Тест ошибок:

```
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> абв
Такого варианта нет.
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 5
Такого варианта нет.
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 2
Введите номер комнаты для брони.
> абв
Ошибка. Введите номер комнаты.
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 2
Введите номер комнаты для брони.
```

```
> 1000000000
Ошибка! Такого номера нет.
[1] - Вывести занятость всех номеров
[2] - Занять номер
[3] - Снять бронь с номера
[4] - Выход
> 4
```

**9. Протестируйте программу.**