

## Pro Java №4

Перегрузка методов (method overloading) позволяет в Java создавать несколько методов с одним и тем же именем, но с различными параметрами (разный тип, количество или порядок параметров).

В Java **enum** (перечисление) – это особый тип класса, который используется для определения набора именованных констант. Он может быть полезен, когда тебе нужно работать с фиксированным набором значений, таких как дни недели, сезоны, цвета и т.д. Константы, определенные в **enum**, доступны везде, где доступен этот **enum**.

Пример enum с конструктором и методами:

```
public enum Day {  
    SUNDAY("Первый день недели"),  
    MONDAY("Второй день недели"),  
    TUESDAY("Третий день недели"),  
    WEDNESDAY("Четвертый день недели"),  
    THURSDAY("Пятый день недели"),  
    FRIDAY("Шестой день недели"),  
    SATURDAY("Седьмой день недели");  
  
    private final String description;  
  
    // Конструктор enum  
    Day(String description) {  
        this.description = description;  
    }  
  
    // Метод для получения описания  
    public String getDescription() {  
        return description;  
    }  
}
```

В этом примере:

- Перечисление `Day` содержит константы для каждого дня недели.
- Конструктор перечисления принимает строку, которая описывает день недели.
- Метод `getDescription()` возвращает описание дня.

В Java для метода можно использовать многоточия (...), чтобы передавать переменное количество аргументов в метод без необходимости явного создания массива. Это называется `varargs` (variable arguments).

В перечислениях (`enum`) все элементы являются статическими и финальными по умолчанию.

Ты абсолютно прав! В Java метод `values()` в перечислениях (`enum`) возвращает массив, содержащий все значения этого перечисления в том порядке, в котором они объявлены.

Метод `valueOf(String name)` сравнивает переданную строку с именами всех констант в перечислении и возвращает ту константу, которая точно совпадает с переданной строкой.

Перечисления (`enum`) в Java фактически представляют собой набор фиксированных констант, которые могут использоваться как массив значений.

В Java метод `ordinal()` возвращает порядковый номер (индекс) константы перечисления, соответствующий ее позиции в списке значений `enum`, начиная с 0.

В Java метод `name()` у перечислений (`enum`) возвращает строковое представление имени константы.

В Java все аргументы методов передаются по значению:

- **\*\*Примитивные типы (int, double, char и т.д.)\*\***: Передается копия самого значения переменной. Изменения, сделанные с этим значением внутри метода, не влияют на оригинальную переменную.
- **\*\*Ссылочные типы (объекты)\*\***: Передается копия ссылки на объект. Изменения, сделанные с объектом внутри метода, влияют на оригинальный объект, так как обе ссылки указывают на один и тот же объект в памяти.

В Java строки представляют собой ссылочные типы, но они неизменяемы (`immutable`). Это означает, что после создания строковый объект не может быть изменен. Если ты пытаешься изменить строку, то создается

новый объект строки с измененным значением, а исходный объект остается неизменным.