ЗАНЯТИЕ 12

МЕТОДЫ

- Если переменные и константы хранят некоторые значения, то методы содержат собой **набор операторов**, которые выполняют определенные действия

```
[модификаторы] тип_возвращаемого_значения название_метода ([параметры]){
    // тело метода
}
```

МЕТОДЫ

- По умолчанию главный класс любой программы на Java содержит метод main, Ключевые слова public и static являются модификаторами.

```
public static void main(String[] args) {
    System.out.println("привет мир!");
}
```

- Далее идет тип возвращаемого значения. Ключевое слово void указывает на то, что метод ничего не возвращает.
- Затем идут название метода - main и в скобках параметры метода - String[] args. И в фигурные скобки заключено тело метода - все действия, которые он выполняет.
- Methods_1.java

ПАРАМЕТРЫ МЕТОДОВ

- С помощью параметров мы можем передать в методы различные данные, которые будут использоваться для вычислений.
- Methods_2.java

ПАРАМЕТРЫ МЕТОДОВ

- Метод может принимать параметры переменной длины одного типа. Например, нам надо передать в метод набор числел и вычислить их сумму, но мы точно не знаем, сколько именно чисел будет передано - 3, 4, 5 или больше. Параметры переменной длины позволяют решить эту задачу:
- Methods_3.java
- Троеточие перед названием параметра int ...nums указывает на то, что он будет необязательным и будет представлять массив. Мы можем передать в метод sum одно число, несколько чисел, а можем вообще не передавать никаких параметров. Причем, если мы хотим передать несколько параметров, то необязательный параметр должен указываться в конце.

RETURN

- Методы могут возвращать некоторое значение. Для этого применяется оператор **return**.
- После оператора return указывается возвращаемое значение, которое является результатом метода. Это может быть литеральное значение, значение переменной или какого-то сложного выражения.
- Methods_4.java
- если в качестве возвращаемого типа для метода определен любой другой, отличный от void, то метод обязательно должен использовать оператор **return** для возвращения значения.
- При этом возвращаемое значение всегда должно иметь тот же тип, что значится в определении метода. Нельзя вернуть больше, можно только меньше (Downcasting)

RETURN

- Метод может использовать несколько вызовов оператора return для возвращения разных значений в зависимости от некоторых условий:
- Methods_5.java
- Оператор return применяется не только для возвращаения значения из метода, но и для выхода из метода:
- Methods_6.java

ПЕРЕГРУЗКА МЕТОДОВ

- В программе мы можем использовать методы с одним и тем же именем, но с разными типами и/или количеством параметров. Такой механизм называется **перегрузкой методов** (method overloading).
- Methods_7.java

ОБЛАСТЬ ВИДИМОСТИ

- Переменные доступны только в той области, где они созданы
- 1) Область видимости метода: переменные, объявленные непосредственно внутри метода, доступны в любом месте метода после строки кода, в которой они были объявлены.

```
public class Main {
  public static void main(String[] args) {

    // Code here CANNOT use x

    int x = 100;

    // Code here can use x
    System.out.println(x);
  }
}
```

ОБЛАСТЬ ВИДИМОСТИ

- 2) Блок кода: блок кода относится ко всему коду, заключенному в фигурные скобки {}. Переменные, объявленные внутри блоков кода, доступны только коду между фигурными скобками, который следует за строкой, в которой была объявлена переменная:

```
public class Main {
  public static void main(String[] args) {

    // Code here CANNOT use x

    { // This is a block

      // Code here CANNOT use x

      int x = 100;

      // Code here CAN use x
      System.out.println(x);

   } // The block ends here

  // Code here CANNOT use x

  }
}
```

РЕКУРСИЯ

- Рекурсия – это когда метод вызывает сам себя. Это позволяет разбить задачу на подзадачи, которые, возможно, проще решить.
- Пример: Сложить два числа легко, но сложить диапазон чисел сложнее.
- Например, посчитать сумму чисел до 10
- Recursion_1.java

РЕКУРСИЯ

- Объяснение примера:

```
10 + sum(9)
10 + ( 9 + sum(8) )
10 + ( 9 + ( 8 + sum(7) ) )
...
10 + 9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 + sum(0)
10 + 9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 + 0
```

РЕКУРСИЯ

- Задача: найти сумму в определенном диапазоне (задаем start и end)

ПРАКТИКА

- 1) Из прошлой домашки переписать все используя методы. Максимально сфокусировать, систематизировать логику, т.е вывод, ввод, обработка (максимальный, минимальный и тп) в отдельные методы.