

Возвращаемые значения

Привет!

В этом материале поговорим о возвращаемых значениях.

Как вы уже знаете, методы — это фрагменты кода, которые выполняют какие-то действия. И часто в программах требуется не просто выполнить определённые действия, но и получить результат их работы.

Для этого в методах существует возможность возвращать значение — то есть, по сути, возвращать переменную, которую этот метод сформировал в процессе своего выполнения и которая содержит результат его работы.

Ранее мы создали корзину с тремя методами. Допишем в нашу корзину пару методов, которые будут что-то возвращать.

Можно узнать общую стоимость товаров. Для этого допишем переменную:

```
private int totalPrice = 0;
```

Допишем строку в метод `add`, которая будет прибавлять цены к этой переменной:

```
public static void add(String name, int price) {  
    items = items + "\n" + name + " - " + price;  
    totalPrice = totalPrice + price;  
}
```

В метод `clear()` не забудем дописать её обнуление:

```
totalPrice = 0;
```

Ключевое слово `return`

01:40–06:26

Теперь напишем метод, который нам будет возвращать эту общую цену:

```
public int getTotalPrice(){  
    return totalPrice;  
}
```

Обратите внимание: с помощью ключевого слова `return` наш метод возвращает значение переменной `totalPrice`. Эта переменная имеет тип `int` (целое число), и поэтому мы прописали слово `int` перед именем метода.

То есть для того чтобы метод вернул какую-то переменную, мы должны, во-первых, написать слово `return`, которое обозначает операцию возврата, а во-вторых — прописать перед именем метода тип этой переменной.

Важно, что слово `return` приводит к завершению работы метода. Если после слова `return` попытаться написать какой-то код, он не будет выполняться.

```
public static int getTotalPrice() {  
    return totalPrice;  
    System.out.println("gdhsgkj");  
}
```

Создадим ещё один метод, который будет проверять, есть ли в нашей корзине определённый товар. То есть это будет метод, который не только возвращает значение, но и принимает один параметр:

```
boolean contains(String name){  
    if(items.contains(name)) {  
        return true;  
    }  
    return false;  
}
```

Обратите внимание, что здесь логично было бы написать слово `else`, но поскольку выполнится оператор `return`, то на этом метод завершит свою работу. Если же он не выполнится, то есть не выполнится условие в операторе `if`, то код продолжит выполняться, и сработает следующий `return`.

В данном случае возможно сразу вернуть результат сравнения, так как каждая проверка нам возвращает boolean-значение, и вообще не писать операторы if и else:

```
boolean contains(String name){  
    return items.contains(name);  
}
```

То есть код необходимо сокращать по мере возможности.

Не всегда код можно так сильно упростить, но о его упрощении нужно всегда думать. Если вам кажется, что написанный вами код выглядит слишком сложно и непонятно, — это повод подумать, не стоит ли его переписать попроще.

Ключевое слово void

06:26–09:04

Существуют методы, которые не возвращают никакие значения. В нашем случае это методы add, clear и print. И для обозначения того, что у них нет возвращаемых значений, используется ключевое слово void.

Есть ещё один важный момент, о котором хотелось бы рассказать. Вы уже видели, что методы могут вызывать друг друга.

Напишем ещё один пример того, как это может происходить. Улучшим программу и не будем добавлять дублирующиеся товары. То есть в метод add пропишем условие, которое будет проверять, содержится ли в корзине добавляемый товар или нет:

```
public static void add(String name, int price) {  
    if(contains(name)) {  
        return;  
    }  
    items = items + "\n" + name + " - " + price;  
    totalPrice = totalPrice + price;  
}
```

Посмотрим, как выполняется программа и в каком порядке вызываются методы.

Выполнение программы начинается с метода `main`. Метод `main` вызывает метод `add`. В нём, в свою очередь, вызывается метод `contains`. И перечень методов, которые для выполнения той или иной операции вызываются последовательно друг из друга, называется **стеком вызовов** методов.

Итоги

09:04 — до конца видео

Вы подробно познакомились с несколькими понятиями. Во-первых, с тем, что метод может возвращать переменную, а может её не возвращать. Если он ничего не возвращает, то перед именем метода пишется ключевое слово `void`. Если возвращает, то перед именем метода пишется тип возвращаемого значения.

Во-вторых, если метод что-то возвращает, то делает это с помощью ключевого слова `return`, на котором выполнение метода всегда заканчивается. Поэтому, если в методе `return` выполняется только при выполнении условия `if`, писать дальше `else` не нужно.

Ну и, наконец, вы увидели, что методы могут вызывать друг друга.

Глоссарий

Возвращаемый тип метода, ключевое слово `return`, ключевое слово `void`, стек вызовов