

ПРИМЕР-4: Простая программа на Java – выполнить математический расчёт с переменной.

ЗАДАЧА:

- Вывести на экран пояснение и значение целочисленной переменной, а потом то же, но в 2 раза больше.

РЕШЕНИЕ:

- Пишется программа на Java:

```
1  /*
2  Это еще один короткий пример программы.
3  Присвоить исходному файлу имя "Example4.java"
4  */
5  class VarCalculation {
6      public static void main(String args []) {
7          int num; // в этой строке кода объявляется переменная с именем num
8          num = 100; // в этой строке кода переменной num присваивается значение 100
9          System.out.println("Это переменная num: " + num);
10         num = num * 2;
11         System.out.print("Значение переменной num * 2 равно ");
12         System.out.println(num);
13     }
14 }
```

- Исходному Файлу программы следует присвоить Имя «**VarCalculation.java**». По принятому соглашению Имя Главного Класса должно совпадать с Именем Файла с исходным кодом.

- Строки ##1–4:

```
/*
Это еще один короткий пример программы.
Присвоить исходному файлу имя "Example4.java"
*/
```

- «/* ... */» – Многострочный Комментарий – Весь текст, расположенный между «/*» и «*/» игнорируется Компилятором. В данном случае описывает программу и напоминает, что исходному файлу должно быть присвоено имя «**VarCalculation.java**».

- Строка #5:

```
class VarCalculation {
```

- «**class**» – Ключевое слово, которое служит для объявления вновь определяемого Класса.
- «**VarCalculation**» – служит в качестве идентификатора, обозначающего Имя Класса.
- «{ ... }» – Всё определение Класса и его членов, должно располагаться между фигурными скобками.

- Строка #6:

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

- «**public**» – модификатор доступа – даёт возможность управлять видимостью Членов Класса – этот Член доступен из кода за пределами Класса, где он определён. Метод «**main()**» должен быть определён как «**public**», поскольку при запуске он должен вызываться из кода за пределами его Класса.
- «**static**» – позволяет вызывать метод «**main()**» без получения экземпляра Класса – необходимо потому, что метод «**main()**» вызывается виртуальной машиной JVM перед созданием любых объектов.
- «**void**» – сообщает Компилятору, что метод «**main()**» не возвращает никаких значений.
- «**main()**» – Метод – служит всего лишь началом программы.
- «(...)» – за Именем Метода в круглых скобках указываются Параметры – переменные, служащие для передачи любой информации, требующейся Методу.
- «()» – пустые круглые скобки указываются, если Методу не требуются Параметры.
- «**String args []**» – объявляется Параметр «**args**», обозначающий Массив экземпляров Класса «**String**».
- В объектах типа «**String**» хранятся символьные строки.
- В выражении «**String args []**» объявляется Параметр «**args[]**».
- У метода «**main()**» это единственный, хотя и довольно сложный Параметр.
- «**args[]**» – Параметр – обозначает Массив экземпляров Класса «**String**».
- «**[...]**» – Массивы берутся в квадратные скобки.
- В данном случае «**args[]**» принимает любые аргументы ком строки, во время выполнения программы.
- «{» – Символ открывающей фигурной скобки обозначает начало тела метода «**main()**». Весь код, составляющий тело Метода, должен располагаться между «{...}» в определении этого Метода.

• Строка #7:

```
int num; // в этой строке кода объявляется переменная с именем num
```

- «**int**» – Ключевое слово – Целочисленный Тип Данных.
- «**num**» – объявляется Целочисленная Переменная с именем «**num**».
- «**;**» – в языке Java все операторы обычно должны оканчиваться точкой с запятой.
- «**// ... ↵**» – Строчный Комментарий, приводимый в середине программы для коротких заметок.

• Строка #8:

```
num = 100; // в этой строке кода переменной num присваивается значение 100
```

- «**num**» – объявленная ранее Целочисленная Переменная с именем «**num**».
- «**=**» – в Java операция присваивания обозначается одиночным знаком равенства.
- «**100**» – целочисленное значение, присваиваемое Переменной с именем «**num**».
- «**;**» – в языке Java все операторы обычно должны оканчиваться точкой с запятой.
- «**// ... ↵**» – Строчный Комментарий, приводимый в середине программы для коротких заметок.

• Строка #9:

```
System.out.println("Это переменная num: " + num);
```

- «**System**» – обозначает predetermined Класc, предоставляющий доступ к Системе.
- «**out**» – поток вывода, связанный с Консолью.
- «**println()**» – Встроенный Метод, которым выполняется вывод данных на экран.
- «**"Это ... "**» – текстовая строка, выводимая на экран, должна быть заключена в двойные кавычки.
- «**+**» – присоединяет значения Переменной «**num**»
- «**num**» – значение Переменной с именем «**num**»
- «**;**» – Оператор, в котором вызывается Метод **println()**, завершается точкой с запятой «**;**».

• Строка #10:

```
num = num * 2;
```

- «**num**» – новое значение, для объявленной ранее Целочисленной Переменной с именем «**num**».
- «**=**» – в Java операция присваивания обозначается одиночным знаком равенства.
- «*****» – арифметическая операция умножения.
- «**2**» – Число 2.
- «**;**» – Оператор, в котором вызывается Метод **println()**, завершается точкой с запятой «**;**».

• Строки ##11–12:

```
System.out.print("Значение переменной num * 2 равно ");  
System.out.println(num);
```

- «**print()**» – Метод – вызывается для вывода текстовой строки «**Значение переменной num * 2 равно**». После этой строки не следует «**↵**» символ новой строки. Таким образом, следующий результат будет выводиться в той же самой строке. Метод «**print()**» действует аналогично методу «**println()**», за исключением того, что после каждого вызова он не выводит символ новой строки.
- «**println()**» – Встроенный Метод, которым выполняется вывод данных на экран.
- «**println(num)**» – Вызов Метода – имя Переменной «**num**» указывается буквально.

• Строки ##13–14:

```
}  
}
```

- Первый символ «**}**» – завершает Метод **main()**.
- Последний символ «**}**» – завершает определение Класcа **VarCalculation**.

- Чтобы выполнить программу – воспользоваться загрузчиком приложений Java, который называется Java.
- Ему нужно передать имя класса «**VarCalculation**» в качестве аргумента командной строки.
- Выполнение данной программы приведёт к выводу на экран следующего результата:

```
Это переменная num: 100  
Значение переменной num * 2 равно 200
```