

## № 2 Введение в JAVA

### Задание

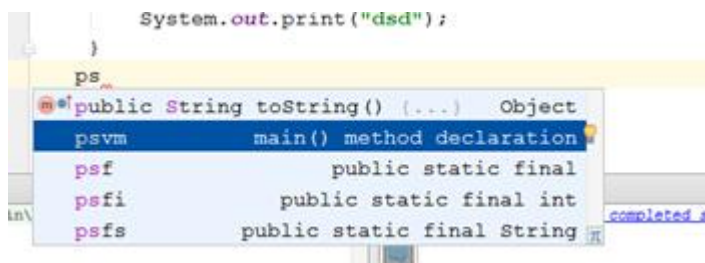
1) Хорошо изучите материал по темам

- ✓ Зарезервированные слова в Java
- ✓ Зарезервированные имена методов Java
- ✓ Идентификаторы
- ✓ Литералы (целые, с плавающей точкой, логические, символьные, строчные) и константы
- ✓ Управляющие символы
- ✓ Операции и примитивные типы данных (целые типы, числа с плавающей точкой, символы и кодировки)
- ✓ Разделители
- ✓ Переменные
- ✓ Операторы
- ✓ Классы оболочки (ссылочные типы данных)
- ✓ Массивы (многомерные массивы)
- ✓ Тип String
- ✓ Консольный ввод-вывод
- ✓ Code convention

2) Напишите текст программы на языке Java с выводом в консоль (по желанию на TextView )

a) Использовать пакеты `by.bstu.фio.basejava`

Создайте в нем класс `JavaTestTepу` с методом `main` (наберите просто `psvm` ). В дальнейшем запускайте `main`.



b) Определить переменные типа char, int, short, byte, long, boolean.

Выполнить над ними следующие операции:

- ✓ `String + int`
- ✓ `String + char`
- ✓ `String + double`
- ✓ `byte = byte + int`
- ✓ `int = double + long`

- ✓ `long = int + 2147483647;` // при выводе значение должно быть положительное
- ✓ `static int sint;` // выведите значение без инициализации
- ✓ `boolean = boolean && boolean`
- ✓ `boolean = boolean ^ boolean`
- ✓ проверьте можно ли выполнить `boolean + boolean`
- ✓ подберите типы для чисел 9223372036854775807 и 0x7fff\_ffff\_fff
- ✓ проинициализируйте и выведите `char - 'a' ; \u0061'; 97;` после чего сложите все `char`.
- ✓ Проверьте результат операции `3.45 % 2.4`
- ✓ Проверьте результат операции `1.0/0.0;`
- ✓ Проверьте результат операции `0.0/0.0;`
- ✓ Проверьте результат операции `log(-345);`
- ✓ Проверьте результат `Float.intBitsToFloat(0x7F800000);`
- ✓ Проверьте результат `Float.intBitsToFloat(0xFF800000);`

c) Задайте две **константы** в классе

```
final int
public final int
public static final int
```

Объясните разницу между ними.

d) Выполните статический импорт

```
import static java.lang.Math.*;
```

- ✓ выведите значения **Math.PI**; `Math.E`; округлите их (`Math.round()`); найдите минимальное среди них `Math.min(p,e)`;
- ✓ сгенерируйте случайное число из диапазона [0,1)

e) Создать объекты разных **классов оболочек** (`Boolean`, `Character`, `Integer`, `Byte`, `Short`, `Long`, `Double`)

- ✓ выполните на них арифметические, логические и битовые операторы (`,` `>>>`, `>>`, `~`, `&`, `*`, `-`, `+`) – выборочно
- ✓ введите `MIN_VALUE` и `MAX_VALUE` для `Long` и `Double`
- ✓ выполнить упакровку и распаковку для типов `Integer` и `Byte`
- ✓ вызовите для `Integer` методы : `parseInt` ; `toHexString` ; `compare` ; `toString` ; `bitCount` ; `isNaN` ...

f) Выполните преобразование числа типа **String** (`String s34 = "2345";`) к `int` , используя: конструктор, `valueOf`, `parse`....

- ✓ переводите строку в массив байтов и обратно из массива байтов в строку
- ✓ преобразуйте строку в логический тип 2-мя способами.
- ✓ определите два строки (`String`) с одинаковыми инициализаторами. Выполните `==`, `equals`, `compareTo`. В чем разница, поясните

результат. Одной из строк присвойте null. Повторите все тип варианта сравнения.

- ✓ для произвольной строки выполните функции split, contains, hashCode, indexOf, length, replace.
- ✓

g) Проверьте какая из форм объявлений многомерного массива допустима:

```
char[][] c1;  
char[] c2[];  
char c3[][];
```

- ✓ для c1

```
c1 = new char[3][];
```

сделайте так, чтобы каждая последующая строка содержала на один элемент больше чем предыдущая. Выведите c1.length; c1[0].length и т.д.

- ✓ проинициализируйте c2 и c3 и выполните :

```
boolean comRez = c2==c3;  
c2 = c3;
```

поясните результат

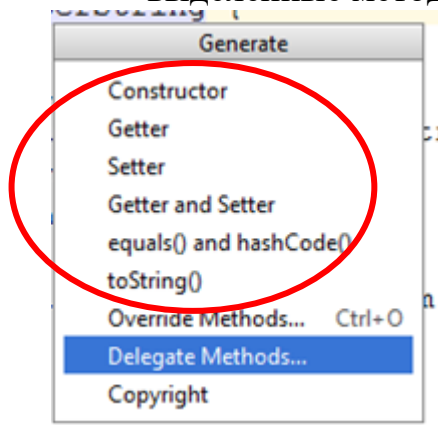
- ✓ выведите один из массивов вращенным циклом (foreach)

```
for(тип итер_пер : коллекция)
```

- ✓ выйдите преднамеренно за границы массива. Что будет получено?

h) Создайте класс WrapperString с приватным полем типа String.

- ✓ Используя Code->Generate (Alt+Insert) сгенерируйте выделенные методы



Определите метод

```
public void replace (char oldchar, char newchar)
```

- ✓ Создайте объект, выполните метод
- ✓ Определите анонимный класс с переопределенной реализацией replace (char oldchar, char newchar) и дополнительным методом delete (char newchar)

- i) Документируйте код. Добавить комментарии в программы в виде `/** комментарий */`,

✓ Добавьте в комментарии дескрипторы

Для класса – @author, @version

Для метода main - @return @throws @param

Для произвольной переменной - @value @see

✓ сгенерируйте html-файл документации.

Можно через утилиту javadoc.exe – извлекает информацию из классов

Или Tools --> Javadoc Настройте Generate Javadoc диалог и выберите Custom scope

