



Типове данни. Променливи и константи

Разработил: инж.А.Анчев



1.Променливи

- 🚀 **Променливата** е идентификатор свързан с област от паметта, където може да бъде съхранена стойност;
- 🚀 **Локална променлива** – променлива декларирана в метод;
- 🚀 Тип на променливата;
- 🚀 Име на променливата;





2.Базови типове на JAVA

Тип	Описание	Размер от паметта	Клас обвивка
Byte	Целочислен тип, диапазон $-128 \div 128$	8 бита	Byte
Short	Целочислен тип, диапазон $-32768 \div 32768$	16 бита	Short
Int	Целочислен тип, диапазон $-2147483648 \div 2147483648$	32 бита	Integer
Long	Целочислен тип, диапазон $-9223372036854775808 \div 9223372036854775808$	64 бита	Long
Float	Числов тип с плаваща запетая, диапазон $3,4 \times 10^{38} \div -3,4 \times 10^{38}$	32 бита	Float
Double	Числов тип с плаваща запетая, диапазон $1,7 \times 10^{308} \div -1,7 \times 10^{308}$	64 бита	Double
Char	Символен тип – приема стойност отделен символ	16 бита	Character
boolean	Логически тип. Приема стойност true (истина) или false (лъжа)	Зависи от вирт. машина	Boolean





3. Деклариране на променливи

- Променливите могат да бъдат деклариране на всяко място в програмата, стига това да бъде преди мястото на използването им;
- Ако се декларират няколко променливи от един тип тогава може да се използва един идентификатор а променливите да се разделят със запетая;





```
package area;
```

```
/**  
 * Програма за определяне площ и обиколка на окръжност  
 * @author eng.Anchev  
 */  
public class Area {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        double radius, circum, area, PI=3.14159265359;  
        //---- Initialize variables  
        radius=1.;  
        circum=2.*PI*radius;  
        area=radius*radius*PI;  
        //---- screen output  
        System.out.println("Обиколка на окръжност = "+circum);  
        System.out.println("Радиус = "+radius +", Площ= "+area);  
    }  
}
```

Red line and arrow pointing to the PI value in the code.



SOCIETY
ROBOTIC



```
/*  
 * Пример за деклариране на променливи  
 */  
package variables;  
/**  
 * @author eng.Anchev  
 */  
public class Variables {  
  
    public static void main(String[] args) {  
        // Целочислена променлива:  
        int number=123;  
        // Реална променлива:  
        double x=32.1;  
        // Символна променлива:  
        char symb='A';  
        // Логическа променлива:  
        boolean state=true;  
        // Текстова променлива за съставяне съдържанието на съобщението:  
        String text="Използвани променливи:\n";  
        // Дописва се стойността на целочислената променлива:  
        text=text+"Цяло число: "+number+"\n";  
        // Дописва се стойността на числовата променлива:  
        text=text+"Реално число: "+x+"\n";  
        // Дописва се стойността на символната променлива:  
        text=text+"Символ: "+symb+"\n";  
        // Дописва се стойността на логическата променлива:  
        text=text+"Логическа стойност: "+state;  
        // Показване на съобщението:  
        System.out.println(text);  
    }  
}
```



```
/*  
 * Деклариране на променливи и изход на конзола  
 */  
package usingvarconsole;  
  
/**  
 *  
 * @author eng.Anchev  
 */  
public class UsingVarConsole {  
  
    public static void main(String[] args)  
    {  
        // Целочислена променлива:  
        int number=123;  
        // Реална променлива:  
        double x=32.1;  
        // Символна променлива:  
        char symb='A';  
        // Логическа променлива:  
        boolean state=true;  
        // Показване на текст:  
        System.out.println("Използвани променливи:");  
        // Показване стойността на целочислената променлива:  
        System.out.println("Цяло число: "+number);  
        // Показване стойността на числовата променлива:  
        System.out.println("Реално число: "+x);  
        // Показване стойността на символна променлива:  
        System.out.println("Символ: "+symb);  
        // Показване стойността на логическа променлива:  
        System.out.println("Логическа стойност: "+state);  
    }  
}
```



4. Аритметични оператори

Оператор	Описание
+	Бинарен оператор за събиране от вида $A+B$
-	Бинарен оператор за изваждане от вида $A-B$
*	Оператор за умножение
/	Бинарен оператор за деление. Ако операндите са цели числа, тогава извършва целочислено деление ($15/4=3$). За да се извършва обикновено деление трябва да се ползва <code>Double</code> .
%	Бинарен оператор за изчисляване на остатък от деление
++	Унарн оператор за инкрементиране
--	Унарн оператор за декрементиране

