

ОБЕКТНО – ОРИЕНТИРАНО ПРОГРАМИРАНЕ

гл.ас. д-р Мария Евтимова
каб.2305а
e- mail:mevtimova@tu-sofia.bg

Учебен план за ООП- JAVA

- Лекции- 15 часа
- Упражнения- 15 часа
- Курсов Проект
- Изпит

Литература

1. Брус Екел, Да мислим на JAVA, I и II том, Софтпрес, 2009
2. Bruce Eckel, Thinking in JAVA, Prentice Hall, 2006
3. Joyce Farrel, JAVA Programming, 9th Edition, 2019
4. Herbert Schildt, Java A beginner's Guide, 7th Edition, 2018

Защо JAVA?

- Почти изцяло обектно – ориентиран
- Прост и мощен
- Има големи библиотеки от предефинирани обекти и операции
- Платформено независим език (Write Once, Run Anywhere)
- Повече сигурност
- Архитектурно неутрален
- Преносим
- Многонишково програмиране
- Интерпретативен
- Динамичен
- Разпределен
- Високо изпълнение

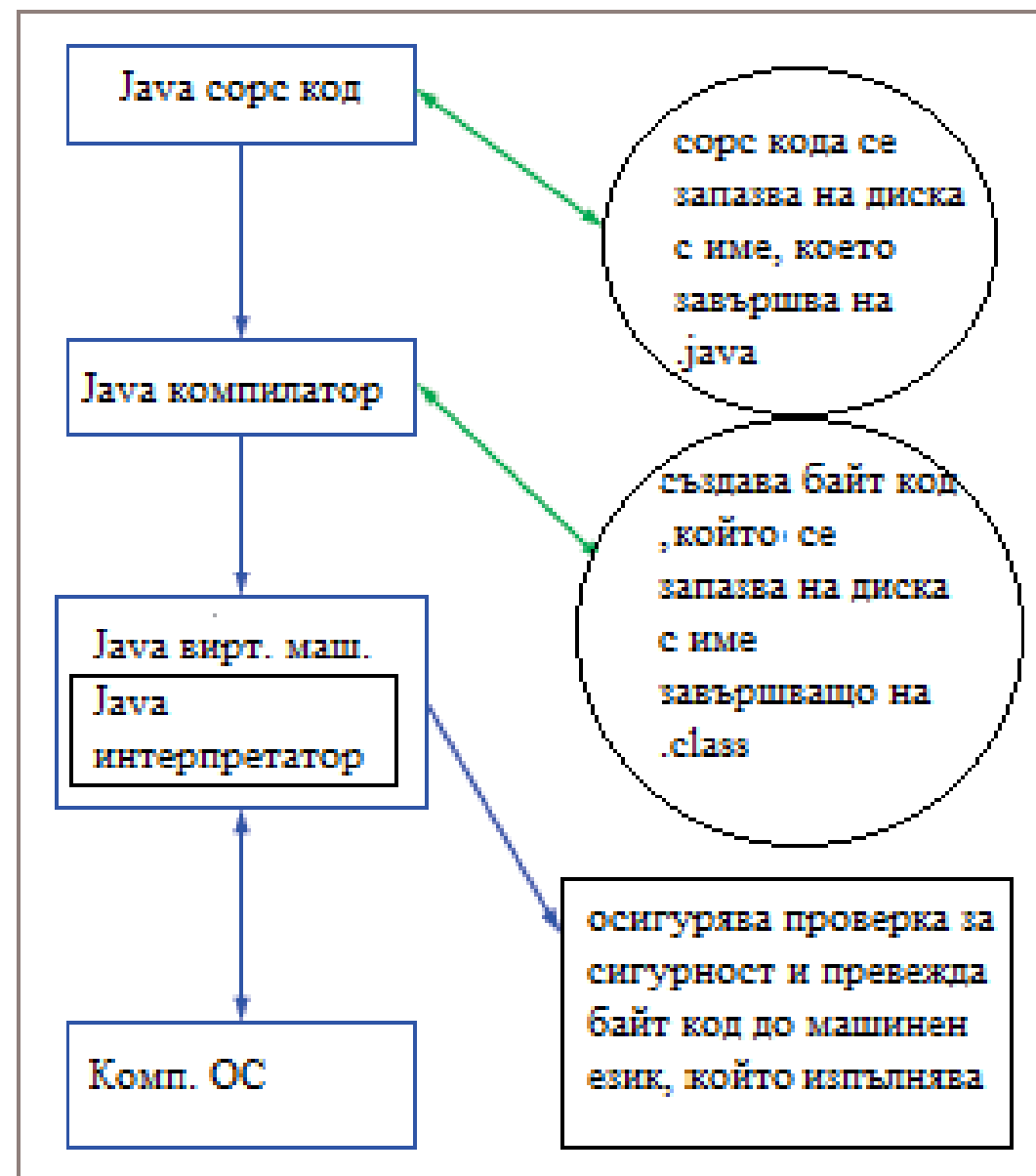
Езикът Java е създаден James Gosling

- Java /Sun - 1995
- Безплатен
- С отворен код
- GNU GPL (General Public license)

Java Virtual Machine

- Файловете .class генерирани от компилатора не са изпълними бинарни файлове
 - *Така Java комбинира компилиране и интерпретация*
- Вместо това те съдържат двоични кодове, които могат да се изпълняват от Java виртуалната машина
- Това осигурява на JAVA платформена независимост и по-добра сигурност

Java среда



JAVA програма

колекция от обекти, които си комуникират, като
извикват методите между тях

- Обектите имат състояния и поведения. Пример: Кола има условия - цвят, име, модел, година. Един обект е инстанция на клас.
- Клас може да бъде дефиниран като шаблон / план, който описва поведението / състоянието, което поддържа обектът от неговия тип.
- Методът е основно поведение. Класът може да съдържа много методи. Тя е в методи, където се записва логиката, данните се манипулират и всички действия се изпълняват.
- Променливи на образци - Всеки обект има свой уникален набор от променливи на екземпляри. Състоянието на обекта се създава от стойностите, присвоени на тези пробни променливи.

Структура на JAVA програма

- Обикновено Java файла изглежда, като :

```
import java.awt.*;  
import java.util.*;  
  
public class SomethingOrOther {  
    // определението на обектите е тук  
    . . .  
}
```

Това трябва да бъде във файл `SomethingOrOther.java` !

Основен синтаксис

- Java прави разлика между малки и големи букви;
`maxval`, `maxVal`, и `MaxVal` са три различни имена
- Имената на класовете започват с голяма буква:
`class MyFirstJavaClass`
- Имената на методите започват с малка буква:
`public void myMethodName()`
- Слятите думи започват с голяма буква: `theBigOne`
- Долна черта не се използва в имената
- Името на програмния файл трябва да бъде, като името на класа

Коментари в JAVA

/*Пример за коментари в JAVA*/

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

// Едноредов коментар

/* Много редов коментар*/

```
System.out.println(„Comment“);  }}
```

Пример за клас

```
public class Car {  
    string Brand;  
    int yearModel;  
  
    int priceModel;  
  
    void setYear(int year) {  
        yearModel = year;  
    }  
  
    int getPrice() {  
        return priceModel;  
    }  
  
    String getBrand() {  
        return brand; }  
}
```

Друг пример за клас

```
class Driver extends Person {  
    long driversLicenseNumber;  
    Date expirationDate;  
}
```

Конструктор

- Всеки клас има конструктор
- При създаване на обект- поне един конструктор се създава
- Конструктора има същото име, като класа
- Класа може да има повече от един конструктор

Пример за конструктор

```
public class Machine {  
  
    public Machine() { }  
  
    public Machine(String name) {  
        // Този конструктор има един аргумент име  
    }  
}
```


Създаване и използване на обект

- Person john;
john = new Person ();
john.name = „Ivan Petrov“;
john.age = 37;
- Person mary = new Person ();
mary.name = “Katq Ivanova”;
mary.age = 33;
mary.birthday ();

```
public class Machine {  
    public Machine(String name) {  
        // Този конструктор има един параметър  
        System.out.println("Passed Name is : " + name );  
    }  
    public static void main(String []args) {  
        // Създаване на обект myMachine  
        Machine myMachine = new Machine( "Vending machine" );  
    }  
}
```

Йерархия на класа

- Класовете са подредени в йерархия
- Всички класове наследяват един базов клас **Object**
- Всички обекти в еднобазовата йерархия имат общ интерфейс- един тип
- Класа може да има под класове
- Всеки клас наследява всички полета и методи на своя вероятностен над клас

Масивите са обекти

- `Person mary = new Person ();`
- `int myArray[] = new int[5];`
 - *or:*
- `int myArray[] = {1, 4, 9, 16, 25};`
- `String languages [] = {"Prolog", "Java"};`

Изброими типове данни

```
class HotChocolate {  
    enum HotChocolate{ SMALL, MEDIUM, LARGE }  
    HotChocolateSize size;  
}  
  
public class HotChocolateTest {  
    public static void main(String args[]) {  
        HotChocolate chocolate= new HotChocolate();  
        chocolate.size = HotChocolate.HotChocolateSize.MEDIUM ;  
        System.out.println("Size: " + chocolate.size);  
    }  
}
```

Изход на програмата:

Size: MEDIUM

Популярни редактори на JAVA

- Eclipse
- Netbeans

Ресурси

- Microsoft Windows systems
- Netbeans IDE
- Java SE (11)/Open JDK

JAVA SE 11

With JDK 11 Oracle has updated the license terms on which we offer the Oracle JDK. The new [Oracle Technology Network License Agreement for Oracle Java SE](#) is substantially different from the licenses under which previous versions of the JDK were offered. Please review the new terms carefully before downloading and using this product.

Oracle also offers this software under the [GPL License](#) on [jdk.java.net/11](#)

See also:

- [Java Developer Newsletter](#): From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- [Java Developer Day](#) hands-on workshops (free) and other events
- [Java Magazine](#)

[JDK 11.0.2 checksum](#)

Java SE Development Kit 11.0.2

You must accept the [Oracle Technology Network License Agreement for Oracle Java SE](#) to download this software.

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux	147.28 MB	jdk-11.0.2_linux-x64_bin.deb
Linux	154.01 MB	jdk-11.0.2_linux-x64_bin.rpm
Linux	171.32 MB	jdk-11.0.2_linux-x64_bin.tar.gz
macOS	166.13 MB	jdk-11.0.2_osx-x64_bin.dmg
macOS	166.49 MB	jdk-11.0.2_osx-x64_bin.tar.gz
Solaris SPARC	186.78 MB	jdk-11.0.2_solaris-sparcv9_bin.tar.gz
Windows	150.94 MB	jdk-11.0.2_windows-x64_bin.exe
Windows	170.96 MB	jdk-11.0.2_windows-x64_bin.zip

- [Technical Articles](#)
- [Demos and Videos](#)
- [Forums](#)
- [Java Magazine](#)
- [Developer Training](#)
- [Tutorials](#)
- [Java.com](#)

New Project

Steps

1. Choose Project
2. ...

Choose Project

Categories:

- Java
- JavaFX
- Maven
- NetBeans Modules
- +
- Samples

Projects:

- Java Application
- Java Class Library
- Java Project with Existing Sources
- Java Free-Form Project

Description:

Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

< Back

Next >

Finish

Cancel

Help

New Java Application

X

Steps

1. Choose Project

2. **Name and Location**

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class

☒ Set as Main Project

< Back

Next >

Finish

Cancel

Help

Java приложение

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package javaapplication1;

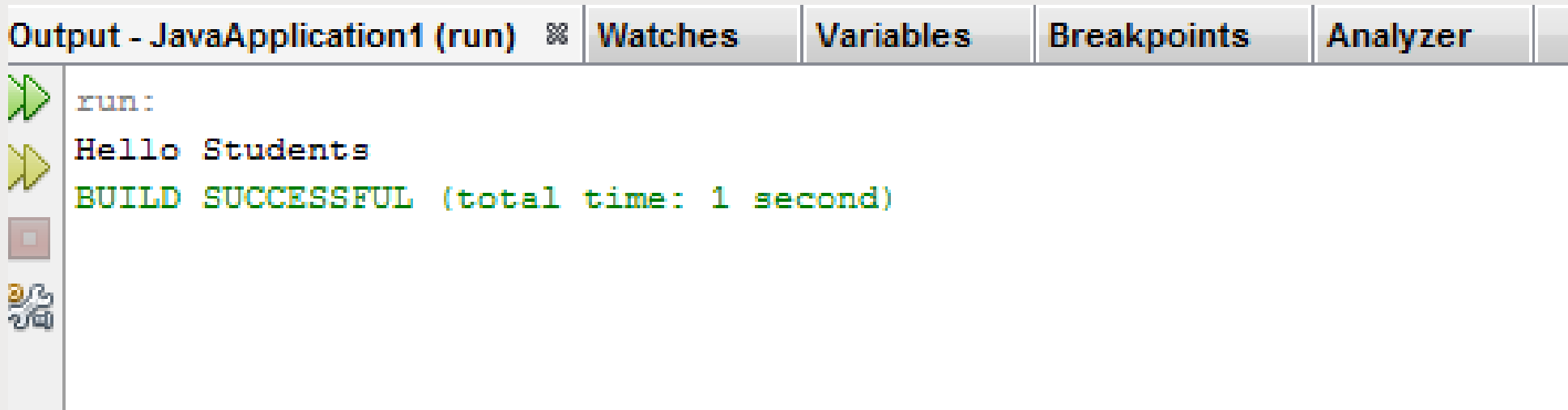
/**
 *
 * @author mari
 */
public class JavaApplication1 {

    /**
     * @param args the command line arguments
     */

    public static void main(String[] args) {
        // TODO code application logic here
        System.out.println("Hello Students");
    }
}

```

Результат



Ключови думи в JAVA

<code>abstract</code>	<code>continue</code>	<code>for</code>	<code>new</code>	<code>switch</code>
<code>assert</code>	<code>default</code>	<code>goto</code>	<code>package</code>	<code>synchronized</code>
<code>boolean</code>	<code>do</code>	<code>if</code>	<code>private</code>	<code>this</code>
<code>break</code>	<code>double</code>	<code>implements</code>	<code>protected</code>	<code>throw</code>
<code>byte</code>	<code>else</code>	<code>import</code>	<code>public</code>	<code>throws</code>
<code>case</code>	<code>enum</code>	<code>instanceof</code>	<code>return</code>	<code>transient</code>
<code>catch</code>	<code>extends</code>	<code>int</code>	<code>short</code>	<code>try</code>
<code>char</code>	<code>final</code>	<code>interface</code>	<code>static</code>	<code>void</code>
<code>class</code>	<code>finally</code>	<code>long</code>	<code>strictfp</code>	<code>volatile</code>
<code>const</code>	<code>float</code>	<code>native</code>	<code>super</code>	<code>while</code>

Ключови думи в JAVA

- static- едно място за съхраняване
- abstract- преобразуване на горе
- final- константа; стойност, която не се променя

Java спецификатори за достъп

- Public – интерфейсен достъп
- Private- не може да се пипа
- Protected- приятелски
- Default

Примитивни типове данни

- Основни типове данни са `int`, `double`, `boolean`, `char`
- Също `byte`, `short`, `long`, `float`
- `boolean` има стойности `true` или `false`
- Декларацията на типовете данни
 - *`double x, y;`*
 - *`int count = 0;`*

Примитивни типове данни

Тип	Памет [бита]	Минимална стойност	Максимална стойност	Стойност по подразбиране
byte	8	-128	127	0
short	16	-32768	32767	0
int	32	-2147483648	2147483647	0
long	64	-9E18	9E18	0L
float	32	$\approx -3.4E+38$ 7 зн.ц.	$\approx 3.4E+38$ 7 зн.ц.	0.0f
double	64	$\approx -1.7E+308$ 15 зн.ц.	$\approx 1.7E+308$ 15 зн.ц.	0.0d
char	16	'\u0000' (0)	'\uffff' (65535)	'\u0000'
boolean	*	false	true	false

Оператори в JAVA

- Съкратени преобразувания; +=, *=, /=, %=
- Аритметични знаци + - * / %
- Java също има ++ и --
- Логически оператори && || !
- Побитови оператори & | ~ ^ << >> >>>
- Релационни оператори < <= == != >= >

Съкратени преобразувания;

`x -= y; // x = x - y;`

`x *= y; // x = x * y;`

`x /= y; // x = x / y;`

`x %= y; // x = x % y;`

`q &&= p; // q = q && p;`

Литерали

литерал	интерпретация	ASCII означение
'\n'	Нов ред	new line
'\b'	Позиция назад	backspace
'\r'	връщане на каретката в началото на реда	carriage return
'\f'	нова страница	form feed
'\a'	звънец	bell
'\\'	обратна наклонена черта	backslash
'\''	апостров	apostrophe
'\0'	ASCII код 0	NULL
'\t'	хоризонтална табулация	HT
'\v'	вертикална табулация	VT

Условни оператори I

- `if (x < y) smaller = x;`
- `if (x < y){ smaller=x;sum += x;}`
`else { smaller = y; sum += y; }`
- `while (x < y) { y = y - x; }`
- `do { y = y - x; } while (x < y)`
- `for (int i = 0; i < max; i++)`
`sum += i;`

Условни оператори II

```
switch (n + 1) {  
    case 0: m = n - 1; break;  
    case 1: m = n + 1;  
    case 3: m = m * n; break;  
    default: m = -n; break;  
}
```

Break;

Continue;

Условен оператор

булев_израз ? стойност_0: стойност_1

```
static int alternative(int i){  
return i<10 ? i*100:i*10;
```

```
static int alternative(int i){  
if (i<10)  
return i*100;  
else  
return i*10;  
}
```

Java не поддържа

- указатели

- sizeof()

Java и C

- В C почти всичко е във функции
- В Java почти всичко е в класове
- Обикновено има само един клас във файл
- Трябва да има само един **public** клас на файл
- Името на файла трябва да бъде същото, като **public** класа, но с **.java** разширение

Използване на класа Scanner за ВХОДНИ данни

```
Scanner inputDevice= new Scanner (System.in);
```


Scanner Class

```
import java.util.Scanner;

public class GetUserInfo
{
    public static void main(String[] args)
    {
        String name;
        int age;
        Scanner inputDevice = new Scanner(System.in);
        System.out.print("Please enter your name : ");
        name = inputDevice.nextLine();
        System.out.print("Please enter your age :");
        age = inputDevice.nextInt();
        System.out.println("Your name is " + name + " and you are " + age + " years old.");
    }
}
```

Result:

Please enter your name: Jorsh

Please enter your age:22

Your name is Jorsh and you are 22 years old.

Методи

`nextDouble()`

`nextInt()`

`nextLine()`

`next()`

`nextShort()`

`nextByte()`

`nextFloat()`

`nextLong()`