

Dominic Mircea KRISTÁLY

Anca VULPE



# LIMBAJUL DE PROGRAMARE JAVA



EDITURA  
UNIVERSITĂȚII  
TRANSILVANIA  
DIN BRAȘOV



Îndrumar de laborator | Brașov, 2023



## Cuprins

1. Crearea și rularea programelor Java din linie de comandă .....	1
1.1. Crearea codului sursă .....	1
1.2. Compilarea programului .....	2
1.3. Lansarea în execuție a programului .....	4
2. Crearea și rularea programelor Java cu ajutorul platformei Eclipse.....	5
2.1. Interfața de lucru a platformei Eclipse.....	6
2.2. Crearea unui proiect Java.....	7
2.3. Crearea pachetelor.....	9
2.4. Crearea claselor.....	9
2.5. Aplicația Hello World.....	11
3. Structura unui program Java .....	13
3.1. Definiția clasei .....	13
3.2. Definiția metodelor .....	13
3.3. Instrucțiuni .....	13
4. Tipuri de date primitive.....	15
4.1. Tipuri întregi.....	15
4.2. Tipuri reale .....	16
4.3. Tipul caracter.....	16
4.4. Tipul logic .....	17
4.5. Conversii între tipurile de date primitive .....	18
4.6. Variabile.....	18
4.7. Constante .....	19
4.8. Operatori .....	19
4.9. Aplicație cu variabile și operatori.....	20
5. Citirea de la tastatură – clasa <code>Scanner</code> .....	21
5.1. Introducerea datelor de la tastatură și afișarea lor .....	22

6. Structuri de control .....	23
6.1. Instrucțiuni de decizie / selecție .....	23
6.1.1. Instrucțiunea <code>if</code> .....	23
6.1.1.1. Clauza <code>else</code> .....	24
6.1.1.2. Clauza <code>else if</code> .....	24
6.1.1.3. Instrucțiuni <code>if</code> imbricate .....	25
6.1.1.4. Utilizarea instrucțiunii <code>if</code> .....	26
6.1.2. Instrucțiunea <code>switch</code> .....	27
6.1.2.1. Instrucțiuni <code>switch</code> imbricate .....	28
6.2. Instrucțiuni iterative .....	29
6.2.1. Instrucțiunea <code>for</code> .....	29
6.2.1.1. Cicluri <code>for</code> imbricate .....	30
6.2.1.2. Utilizarea instrucțiunii <code>for</code> .....	31
6.2.2. Instrucțiunea <code>while</code> .....	35
6.2.2.1. Utilizarea instrucțiunii <code>while</code> .....	36
6.2.3. Instrucțiunea <code>do . . while</code> .....	37
6.3. Instrucțiuni de salt .....	38
6.3.1. Instrucțiunea <code>break</code> .....	38
6.3.2. Instrucțiunea <code>continue</code> .....	39
6.3.3. Instrucțiunea <code>return</code> .....	39
6.4. Utilizarea instrucțiunilor <code>do . . while</code> , <code>switch</code> , <code>break</code> și <code>continue</code> .....	40
7. Tablouri .....	43
7.1. Tablouri unidimensionale – Vectori .....	43
7.1.1. Declararea variabilei tablou .....	43
7.1.2. Instanțierea .....	43
7.1.3. Inițializarea .....	44
7.2. Tablouri multidimensionale – Matrice .....	45
7.2.1. Crearea unui tablou multidimensional .....	45
7.2.2. Atribuirea valorilor către elementele matricei .....	45
7.2.3. Proprietăți <code>length</code> .....	46

7.3. Clasa <code>Arrays</code> .....	46
7.3.1. Metoda <code>equals()</code> .....	46
7.3.2. Metoda <code>fill()</code> .....	47
7.3.3. Metoda <code>sort()</code> .....	48
7.3.4. Metoda <code>binarySearch()</code> .....	48
7.3.5. Metoda <code>arraycopy()</code> .....	49
7.4. Aplicație cu vectori .....	50
7.5. Aplicație cu vectori: Loteria.....	52
7.6. Aplicație cu o matrice bidimensională .....	53
8. Șiruri de caractere .....	55
8.1. Clasa <code>String</code> .....	55
8.2. Clasa <code>StringBuffer</code> .....	60
9. Clase și programare orientată obiect .....	63
9.1. Termeni utilizați în programarea orientată obiect .....	63
9.2. Definiția unei clase .....	63
9.2.1. Modificatori de acces .....	64
9.2.2. Alți modificatori.....	64
9.2.2.1. Modificatorul <code>final</code> .....	64
9.2.2.2. Modificatorul <code>static</code> .....	65
9.2.2.3. Modificatorul <code>synchronized</code> .....	65
9.2.3. Proprietăți .....	65
9.2.4. Metode .....	65
9.2.5. Constructori.....	67
9.2.6. Declararea unei instanțe a unei clase .....	68
9.2.7. Accesul la membrii unei clase .....	68
9.2.8. Supraîncărcarea metodelor.....	69
9.2.9. Cuvântul cheie <code>this</code> .....	70
9.3. Crearea și utilizare claselor (1) .....	72
9.4. Crearea și utilizarea claselor (2) .....	74
9.5. Clase interne.....	75

10. Moștenirea .....	81
10.1. Accesul la membrii unei clase moștenite.....	82
10.2. Apelarea constructorilor .....	83
10.2.1. Folosirea cuvântului cheie <code>super</code> .....	84
10.2.1.1. Folosirea cuvântului cheie <code>super</code> .....	84
10.2.2. Moștenirea pe mai multe niveluri .....	85
10.2.2.1. Moștenirea pe mai multe niveluri .....	85
10.2.3. Supradefinirea metodelor folosind moștenirea .....	87
10.2.3.1. Supradefinirea metodelor folosind moștenirea .....	88
10.2.4. Cuvântul cheie <code>final</code> și moștenirea .....	90
11. Tratarea excepțiilor .....	91
11.1. Proceduri de tratare a excepțiilor .....	91
11.1.1. Blocuri <code>try</code> imbricate .....	97
11.2. Lucrul cu excepții neinterceptate .....	98
11.3. Metode care nu tratează excepțiile .....	99
11.4. Excepții verificate și neverificate .....	100
11.5. Tratarea excepțiilor folosind superclasa <code>Exception</code> .....	101
11.6. Exemple .....	102
11.6.1. Excepții de I/O .....	102
11.6.2. Excepții: Depășirea indexului unui vector .....	103
11.6.3. Excepții: Vector cu dimensiune negativă .....	104
11.6.4. Excepții: <code>NullPointerException</code> .....	105
11.6.5. Excepții: <code>ArrayIndexOutOfBoundsException</code> .....	106
12. Interfețe .....	107
13. Fluxuri de intrare / ieșire (fișiere) .....	112
13.1. Fișiere și sisteme de fișiere .....	112
13.2. Clasa <code>FILE</code> .....	112
13.2.1. Afișarea listei de fișiere dintr-un director .....	115

13.3. Fluxuri.....	116
13.3.1. Scrierea într-un fișier.....	117
13.3.2. Citirea dintr-un fișier.....	118
13.3.3. Adăugarea datelor într-un fișier.....	119
13.3.4. Citirea și scriere unui obiect într-un fișier .....	120
14. Interfețe grafice .....	124
14.1. Pachetul <code>javax.swing</code> .....	125
14.2. O aplicație cu interfață grafică simplă.....	125
14.3. Crearea unei casete de dialog .....	127
14.4. Concatenarea a două șiruri de caractere .....	128





# LABORATORUL 1

## 1. Crearea și rularea programelor Java din linie de comandă

Pașii ce trebuie urmați pentru crearea unui program Java sunt prezentați schematic în figura de mai jos:

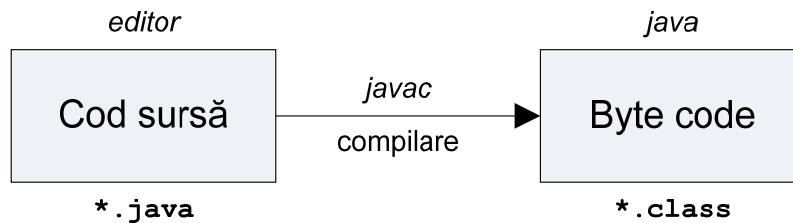


Fig. 1. Etapele necesare creării unui program Java

### 1.1. Crearea codului sursă

Codul sursă este scris în limbajul Java și rezidă într-unul sau mai multe fișiere text având extensia “.java”. Pentru scrierea programului se poate utiliza orice editor de texte. Dacă se lucrează sub sistemul de operare *Microsoft Windows*, se poate utiliza, de exemplu, aplicația *Notepad*. Dacă se lucrează sub Unix/Linux programul *vi* poate fi folosit pentru scrierea codului sursă.

Figura 2 prezintă codul sursă al unui program Java care afișează pe ecran textul „Hello World din Java!”, așa cum arată el în aplicația *Notepad*.

```

class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello world din Java!");
    }
}
  
```

Fig. 2. Editarea programelor Java cu ajutorul aplicației *Notepad* din Windows



Numele fișierului care conține codul sursă al programului trebuie să aibă numele identic cu numele clasei ce conține metoda `main()`.

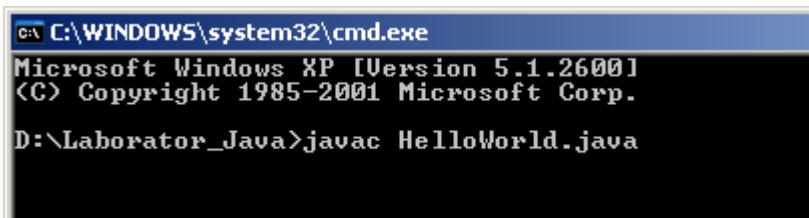
## 1.2. Compilarea programului

Transformarea codului sursă în codul de octeți (*byte code*) înțeles de JVM (*Java Virtual Machine*) se realizează prin compilarea programului. Pe sistemele Windows acest lucru este realizat de executabilul `javac.exe`, ce poate fi apelat dintr-o fereastră sistem.

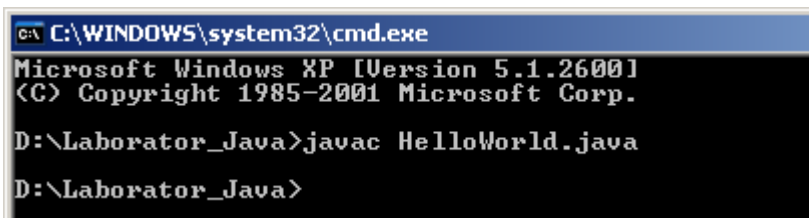
Compilerul Java face parte din pachetul JDK (*Java Development Kit*) care trebuie să fie instalat pe calculatorul pe care se dezvoltă programe Java. Acest pachet poate fi descărcat, gratuit, de pe site-ul companiei *Sun microsystems*. Programele prezentate în continuare au fost scrise și testate folosindu-se versiunea 6 a pachetului JDK. Acest pachet poate fi descărcat de la adresa: <http://java.sun.com/javase/downloads/index.jsp>.

Pentru a compila programul „HelloWorld”, prezentat în figura 2, se deschide o fereastră sistem (*Command Prompt*), în care se scrie următoarea comandă, urmată de tasta *CR (Enter)*:

```
javac HelloWorld.java
```



Dacă programul a fost compilat cu succes, pe ecran apare din nou „*command prompt*”-ul:



În directorul de lucru apare un nou fișier, numit `HelloWorld.class`, ce conține codul de octeți al programului (așa cum arată figura 3).

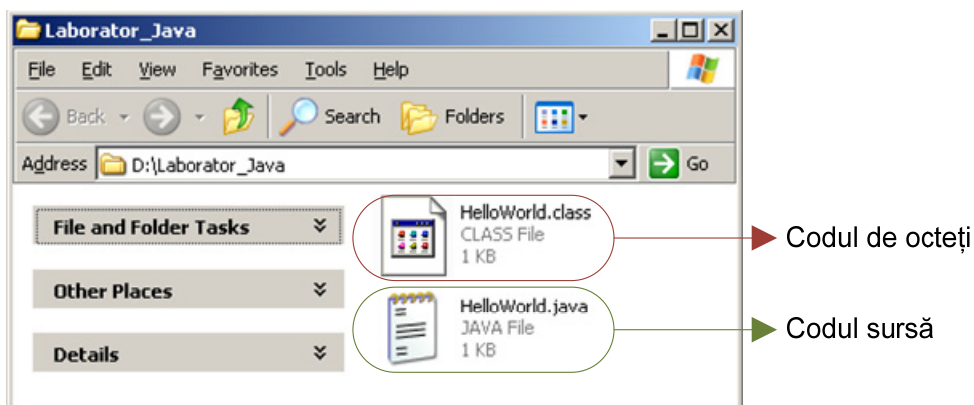


Fig. 3. Codul sursă și fișierul obținut în urma compilării