

## Fundamentele Java

**Durata:** 40 - 56 ore, funcție de ponderea practică

**Infrastructura folosită:** retroproiector, tabla, calculator personal pe care este instalat Java și Eclipse.

**Cine poate participa:** două categorii

- programatori ce vin din lumea programării procedurale, de exemplu C
- programatori ce stăpânesc deja cel puțin un limbaj orientat obiect (C++, C#, etc.)

Modalitatea de abordare a limbajului de programare Java diferă de background-ul participanților.

**Efectul cursului:** participanții vor primi o introducere solidă în folosirea Java pentru scrierea de programe de sine stătătoare (J2SE) precum și familiarizarea cu Eclipse ca și platformă de dezvoltare. Cursul acoperă noțiunile de bază pentru certificarea SCJP (Sun Certified Java Programmer).

**Cursuri conexe:** Fundamentele UML, Java Enterprise, Șabloane de proiectare.

**Verificarea cunoștințelor:** opțional, pe parcurs și/sau test final

**Suport de curs:** da

**Descriere:** Cursul abordează noțiunile fundamentale ale limbajului de programare Java, principalele pachete din distribuția standard. Se pune accent pe exersarea limbajului Java, însușirea particularităților și stereotipurilor folosite de programatorii în Java, exersarea practică a aspectelor legate de programarea orientată pe obiecte.

### Conținut:

1. Bazele programării în Java: abstractizarea, clase și obiecte, membrii instanță și membrii statici, moștenire și agregare
2. Fundamentele limbajului: elemente lexicale, declararea variabilelor, referințe
3. Operatori și asignări: precedența și asociativitatea operatorilor, conversii de tip, promovarea numerică, operatori, pasarea parametrilor

4. Declarații și controlul accesului: tablouri, metode, supraîncărcarea metodelor, constructori, pachete, modificatori, domeniul de existență și accesibilitatea membrilor, arhive jar
5. Controlul programului și tratarea erorilor, așezii
6. Programare orientată pe obiecte: moștenire, concepte POO, polimorfism, upcasting și downcasting, redefinirea metodelor, ascunderea variabilelor, interfețe, înlănțuirea constructorilor
7. Clase și interfețe încuibate: clase și interfețe membru static, clase membru non-stactice, clase locale, clase anonime
8. Ciclul de viață al obiectelor: garbage collection, finalizarea obiectelor, inițializatori, starea inițială a obiectelor
9. Fire de execuție: `java.lang.Runnable`, `java.lang.Thread`, sincronizarea, stările firelor de execuție, priorități, pornire/oprire, adormire/trezire, așteptare/notificare, `join`, tipuri de blocare, terminare, deadlock
10. Clase fundamentale: `Object`, wrapper, `String`, `StringBuffer`
11. Colecții și dicționare: interfețe, implementări, colecții, mulțimi, personalizare, particularități
12. Fișiere și stream-uri: clasa `File`, stream-uri byte și caracter, filtre, serializarea obiectelor
13. Componente AWT: componente și containere, controale GUI, meniu
14. Layout Management: tipuri layout, ierarhii de componente
15. Tratarea evenimentelor: ierarhii evenimente, modelul delegării evenimentelor, adaptoare de evenimente
16. Desenarea: culori, fonturi, trasare, clipping, thread AWT, tratarea evenimentelor
17. Introducere în Swing: pachete, componente, look & feel, tratare evenimente
18. Applet-uri: ciclu de viață, control, comunicare cu pagina context, securitate
19. Java 5.0, 6.0 – extinderea Java: `StringBuilder`, tipuri generice, enumerare, autoboxing/unboxing, `varargs`, `for/in`, importuri statice, formatare