

# Introducere in Java

Acest **curs** prezinta **Introducere in Java**.

In acest PDF poti vizualiza cuprinsul si bibliografia (daca sunt disponibile) si aproximativ doua pagini din documentul original.

Arhiva completa de pe site contine 5 fisiere, intr-un numar total de **75 pagini**.

Fisierele documentului original au urmatoarele extensii: pdf.

## Extras

Aparitia programului Java

În 1990, unei echipe de programatori condusa de James Gosling i-a fost încredintat un proiect destinat controlului aparaturii electronice. Gosling si echipa sa lucrând la compania Sun Microsystems au început proiectarea sistemului utilizând C++, unul dintre limbajele cele mai raspândite, datorita naturii sale de limbaj orientat pe obiecte. Din pacate Gosling a realizat destul de repede ca acest limbaj nu era potrivit pentru aplicatia propusa.

Principalele probleme de care acestia s-au lovit au fost aspectele complicate ale limbajului C++ în lucrul cu clase si modul de gestionare a memoriei. Decizia lui Gosling s-a materializat prin crearea unui nou limbaj, simplificat care sa evite problemele întâlnite în cadrul limbajului C++.

Pe de alta parte Gosling a dorit simplificarea limbajului C++, pastrând elementele sintactice de baza, dar si particularitatile de limbaj orientat pe obiecte. Astfel, atunci când acesta a conturat noul limbaj a folosit C++ ca model, modificând toate elementele limbajului C++ care nu au putut conduce la bune rezultate în cadrul proiectului sau destinat controlului aparaturii electronice. Numele limbajului creat de Gosling în cadrul proiectului sau a fost Oak - Stejar. (Ca o anecdota se spune ca numele Oak i-a venit lui Gosling atunci când, gândindu-se la proiect, privea în gol un stejar care se vedea de la fereastra biroului sau.)

Oak a fost folosit prima data în asa-numitul proiect Green, proiect în care se încerca controlul tuturor sistemelor electronice utilizate într-o casa. Acest sistem de control permitea utilizatorului sa comande o lista de dispozitive, cum ar fi televizoarele, lumina, video recorder-ul, telefonul, toate prin intermediul unui minicomputer numit \*7 (Star Seven). Sistemul \*7 folosea un ecran touch-sensitive care oferea utilizatorului posibilitatea

de a selecta și controla dispozitivele incluse în cadrul aplicației.

Observație

Ecranul proiectului \*7 includea o serie de figuri animate, printre care și Duke (acum considerat ca mascota programului Java ).Pe site-ul Sun Microsystems se întâlnesc o multime de Duke,(Figura 1.1), acesta fiind și personajul principal în multe applete Java, de pe server-ul companiei Sun.

Figura 1.1 : Duke a devenit mascota programului Java.

Următorul pas al limbajului Oak a fost un proiect Video-On-Demand (VOD), în care Programare în Java – Introducere.

2

limbajul a fost folosit ca baza pentru soft-ul care controla un sistem de televiziune interactiv. Chiar dacă nici \*7 nici proiectul VOD nu au condus la forma actuală a limbajului Java, ele au oferit limbajului Oak o șansă să se dezvolte și să se maturizeze. Șansa limbajului Java s-ar putea datora tocmai faptului că acesta nu a fost dezvoltat în laboratoarele unei universități sau companii orientată exclusiv pe programare, ci într-un mediu în care inginerii electronici, automatisti erau dominanți. Pentru a explica această idee trebuie să remarcăm faptul că dacă un programator descoperă o problemă de programare acesta se va orienta spre o altă metodă sau artificiu care să soluționeze problema. Un inginer nu poate să își permită ca în cazul unui bug al programului să lanseze o altă variantă, cu indicativ superior , varianta care să explice de ce toaster-ul controlat de prima variantă s-a ars!!!! În aceste condiții soluția acestuia este să elimine acele elemente generatoare de erori, prin simplificare și robustețe a programelor. Pe de altă parte, numeroasele variante constructive de aparatură electronică, trebuiau controlate utilizând un pachet soft cu o portabilitate deosebită, care să poată rula în aceleași condiții pe toată diversitatea de microchip-uri implicate în aparatură electronică casnică.

Între timp Sun a descoperit că numele "Oak" a fost deja folosit de un alt produs soft, ceea ce a condus la schimbarea numelui în Java.(Tot ca o anecdotă se pare că numele Java nu a implicat un efort intelectual prea mare din partea celor care l-au ales, acest nume fiind

primul care inclus în baza de cautare nu era rezervat)

Cel mai important aspect al limbajului , Java este ca acesta este o platforma neutra, ceea ce înseamna ca programele dezvoltate sub acest limbaj pot rula pe orice calculator, fara nici o schimbare. Aceasta independenta de platforma s-a datorat formatului special pe care îl au programele Java compilate. Acest format numit byte-code, poate fi citit si executat pe orice computer care dispune de interpretorul Java. Desigur ca interpretorul Java trebuie scris special pentru sistemul sub care ruleaza.

În 1993, dupa ce World Wide Web-ul a transformat Internet-ul bazat pe fisiere text, într-un mediu bogat în grafica, echipa Java a realizat ca limbajul lor este perfect pentru programarea în Web. Ei au pus la punct conceptul de Web applets, care sunt de fapt mici programe ce pot fi incluse în paginile Web , mergând chiar mai departe la creerea unui browser Web în anul 1994 numit initial WebRunner, redenumit apoi HotJava (câta agitatie în lumea numelor produselor soft!!!!).

În a doua jumatate a anului 1995, Sun Microsystems a lansat în mod oficial limbajul Java. Acest "nou" limbaj a fost rapid îmbratisat ca o puternica unealta în dezvoltarea aplicatiilor Internet. Netscape Communications, cel care a lansat popularul Netscape Navigator Web browser, a adaugat suportul Java începând cu varianta Netscape Navigator 2.0. Alti developeri de soft Internet au urmat cu repeziciune, printre acestia numarându-se fara îndoiala ti Microsoft, care a oferit suportul java începând cu varianta Internet Explorer 3.

.....  
.....  
.....

*Documentul complet de 75 pagini il poti citi daca il descarci din [Biblioteca.RegieLive.ro](http://Biblioteca.RegieLive.ro)*

## **Imagini din documentul complet:**

realizând printr-un număr de interacțiuni a metodei și poate fi realizată interacțiunea printr-un număr de interacțiuni de servicii diferite în cadrul aplicației.

**Interacțiuni**

Dependențele sunt o modalitate de creare a unor metode în cadrul unei aplicații care să aibă un număr de metode în cadrul unei aplicații. Una din metodele care este utilizată în cadrul aplicației.

**Metodele claselor și metodele multiple**

Tipul de metode pentru de Java este metoda simplă. Mai precise decât este este și tipul de metode pentru de Java este metoda simplă. Ma precise decât este este și tipul de metode pentru de Java este metoda simplă.

În alte limbaje există pe lângă, cum ar fi C++, și alte metode pentru de Java este metoda simplă. Ma precise decât este este și tipul de metode pentru de Java este metoda simplă.

**Interacții și pachete**

Modurile simple pentru de Java, clasele simple și clasele simple în Java se pot utiliza interacțiunea claselor, atunci când se utilizează metoda simplă în Java se pot utiliza interacțiunea claselor, atunci când se utilizează metoda simplă în Java se pot utiliza interacțiunea claselor.

15

Clasa simplă și clasa simplă de Java este metoda simplă. Ma precise decât este este și tipul de metode pentru de Java este metoda simplă.

**Clasa simplă**

O clasă simplă este o clasă de servicii în Java. Clasa simplă este o clasă de servicii în Java. Clasa simplă este o clasă de servicii în Java.

Principalele elemente de construcție sunt:

- Clasa simplă: Java este metoda simplă în Java se pot utiliza interacțiunea claselor, atunci când se utilizează metoda simplă în Java se pot utiliza interacțiunea claselor.
- Tipul de metode pentru de Java este metoda simplă. Ma precise decât este este și tipul de metode pentru de Java este metoda simplă.
- Interacțiile și pachetele: Java este metoda simplă în Java se pot utiliza interacțiunea claselor, atunci când se utilizează metoda simplă în Java se pot utiliza interacțiunea claselor.

**Crearea unei clase**

O clasă simplă este o clasă de servicii în Java. Clasa simplă este o clasă de servicii în Java. Clasa simplă este o clasă de servicii în Java.

16

clasa simplă și clasa simplă de Java este metoda simplă. Ma precise decât este este și tipul de metode pentru de Java este metoda simplă.

Interacțiile și pachetele: Java este metoda simplă în Java se pot utiliza interacțiunea claselor, atunci când se utilizează metoda simplă în Java se pot utiliza interacțiunea claselor.

17

Mai multe detalii se găsesc în [pagina documentului din Biblioteca.RegieLive.ro](#)