

Java technológiai áttekintés

Jeszenszky Péter

Debreceni Egyetem, Informatikai Kar

jeszenszky.peter@inf.unideb.hu

Utolsó módosítás: 2024. február 18.

Fogalmak

- *Java* (programozási nyelv)
- *Java virtuális gép (Java virtual machine)* (JVM)
- *Java Platform*
- *Java Platform, Standard Edition* (Java SE)
- *Java Platform, Enterprise Edition* (Java EE)
- *Java Runtime Environment* (JRE)
- *Java Development Kit* (JDK)
- *OpenJDK*
- *Java Community Process* (JCP)
- *Java Specification Request* (JSR)
- *JDK Enhancement Proposal* (JEP)

Java (programozási nyelv) (1)

- A nyelvet a 90-es években fejlesztették ki a *Sun Microsystems*-nél, melyet 2009-ben felvásárolt az *Oracle Corporation*.
 - A nyelv atyja: James Gosling
 - Webhely: <https://nighthacks.com/>
- Webhely: <https://www.oracle.com/java/>
<https://dev.java/>

Java (programozási nyelv) (2)

- Általános célú, konkurens, osztály alapú, objektumorientált programozási nyelv.
- Rokonságban van a C és C++ nyelvekkel, de azoktól meglehetősen eltér, elhagyja számos jellemzőjüket és néhány fogalmat más nyelvektől vesz át.
- Erősen típusos és statikusan típusos nyelv.
- A Java programok fordítása rendszerint egy gépfüggetlen bájt kód reprezentációba történik, melynek végrehajtására a Java virtuális gépek szolgálnak.
- Lásd: *The Java Language Specification, Java SE 21 Edition*
<https://docs.oracle.com/javase/specs/>

Java (programozási nyelv) (3)

- Többelvű programozási nyelv, mely az alábbi programozási paradigmákat támogatja:
 - Funkcionális (a Java SE 8-ban jelent meg)
 - Generikus
 - Imperatív
 - Objektumorientált (osztály alapú)
 - Konkurens
 - Strukturált

Java (programozási nyelv) (4)

- A legutóbbi verziója a Java 21.
- A nyelv specifikációja:
 - James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley, Daniel Smith, Gavin Bierman. *The Java Language Specification, Java SE 21 Edition*. August 2023. <https://docs.oracle.com/javase/specs/>

Java (programozási nyelv) (5)

- A TIOBE Index szerint jelenleg a Python, C és C++ után a negyedik legnépszerűbb programozási nyelv.
 - Lásd:
 - *TIOBE Index* <https://www.tiobe.com/tiobe-index/>
 - *TIOBE Programming Community Index Definition* <https://www.tiobe.com/tiobe-index/programming-languages-definition/>
- Lásd még:
 - *PYPL PopularitY of Programming Language* <https://pypl.github.io/PYPL.html>

Java virtuális gép (1)

- Egy saját utasításkészlettel rendelkező és vezérelhető absztrakt számítógép, mely semmit sem tud a Java programozási nyelvről, hanem csak egy speciális bináris formátumot ismer, a `class` állományformátumot.
 - Egy `class` állomány a virtuális gépnek szóló utasításokat (bájtkódokat), egy szimbólumtáblát valamint további járulékos információkat tartalmaz.
- A Java virtuális gép a Java platform sarokköve. A technológia ezen komponense felelős a hardver és operációs rendszer függetlenségéért, a lefordított kód kis méretéért és a felhasználók rosszindulatú programoktól való megvédésének képességéért.
- Specifikáció:
 - Tim Lindholm, Frank Yellin, Gilad Bracha, Alex Buckley, Daniel Smith. *The Java Virtual Machine Specification, Java SE 21 Edition*. August 2023.
<https://docs.oracle.com/javase/specs/>

Java virtuális gép (2)

- Szabad és nyílt forrású implementációk:
 - *HotSpot* (operációs rendszer: platformfüggetlen; licenc: GPLv2)
<https://openjdk.org/groups/hotspot/>
 - Az *OpenJDK* virtuális gépe.
 - *Eclipse OpenJ9* (operációs rendszer: AIX, Linux, macOS, Windows; licenc: *Apache License 2.0/Eclipse Public License 2.0*) <https://www.eclipse.org/openj9/>
<https://github.com/eclipse/openj9>
 - Eredetileg az IBM által fejlesztett virtuális gép (J9).
 - A *HotSpot* virtuális gép alternatívája az *OpenJDK*-hoz.
- Nem szabad implementációk:
 - *PTC Perc* (platform: beágyazott rendszerek)
<https://www.ptc.com/en/products/developer-tools/perc>
 - *Azul Platform Prime* (korábban *Zing*) (operációs rendszer: Linux)
[https://www.azul.com/products/prime/](https://www Azul.com/products/prime/)

Java platform (1)

- A platform kifejezés azt a hardver vagy szoftver környezetet jelenti, melyben egy program fut.
 - Például: beágyazott rendszer, operációs rendszer, webböngésző, ...
- A Java platform egy olyan szoftver platform, mely más hardver platformokon fut, és melynek az alábbi két fő komponense van:
 - Java virtuális gép (JVM)
 - Java alkalmazásprogramozási interfész (API)
- Lásd: *The Java Tutorials – Trail: Getting Started – Lesson: The Java Technology Phenomenon – About the Java Technology*
<https://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html>

Java platform (2)

- Történetileg az alábbi Java platformok léteznek:
 - *Java Card*
<https://www.oracle.com/java/technologies/java-card-tech.html>
 - *Java Platform, Enterprise Edition (Java EE)/Jakarta EE*
<https://www.oracle.com/java/technologies/java-ee-glance.html>
 - *Java Platform, Micro Edition (Java ME)*
<https://www.oracle.com/java/technologies/javameoverview.html>
 - *Java Platform, Standard Edition (Java SE)*
<https://www.oracle.com/java/technologies/java-se-glance.html>

Java SE (1)

- Java platform asztali és szerver számítógépekre történő alkalmazásfejlesztéshez.
 - Sokan a Java SE API-ját azonosítják a Java programozási nyelvvel.
- Webhely:
<https://www.oracle.com/java/technologies/java-se-glance.html>

Java SE (2)

- A legutóbbi LTS verzió a Java SE 21, mely 2023. szeptember 19-én jelent meg.
 - Lásd: *Oracle Releases Java 21 and Extends Support Roadmap*
<https://www.oracle.com/news/announcement/ocw-oracle-releases-java-21-2023-09-19/>
- Specifikáció:
 - JSR 396: *Java SE 21* <https://jcp.org/en/jsr/detail?id=396>
- Dokumentáció: <https://docs.oracle.com/en/java/javase/21/>
 - Áttérés: *Oracle JDK Migration Guide, Release 21*
<https://docs.oracle.com/en/java/javase/21/migrate/>
 - API dokumentáció:
<https://docs.oracle.com/en/java/javase/21/docs/api/index.html>

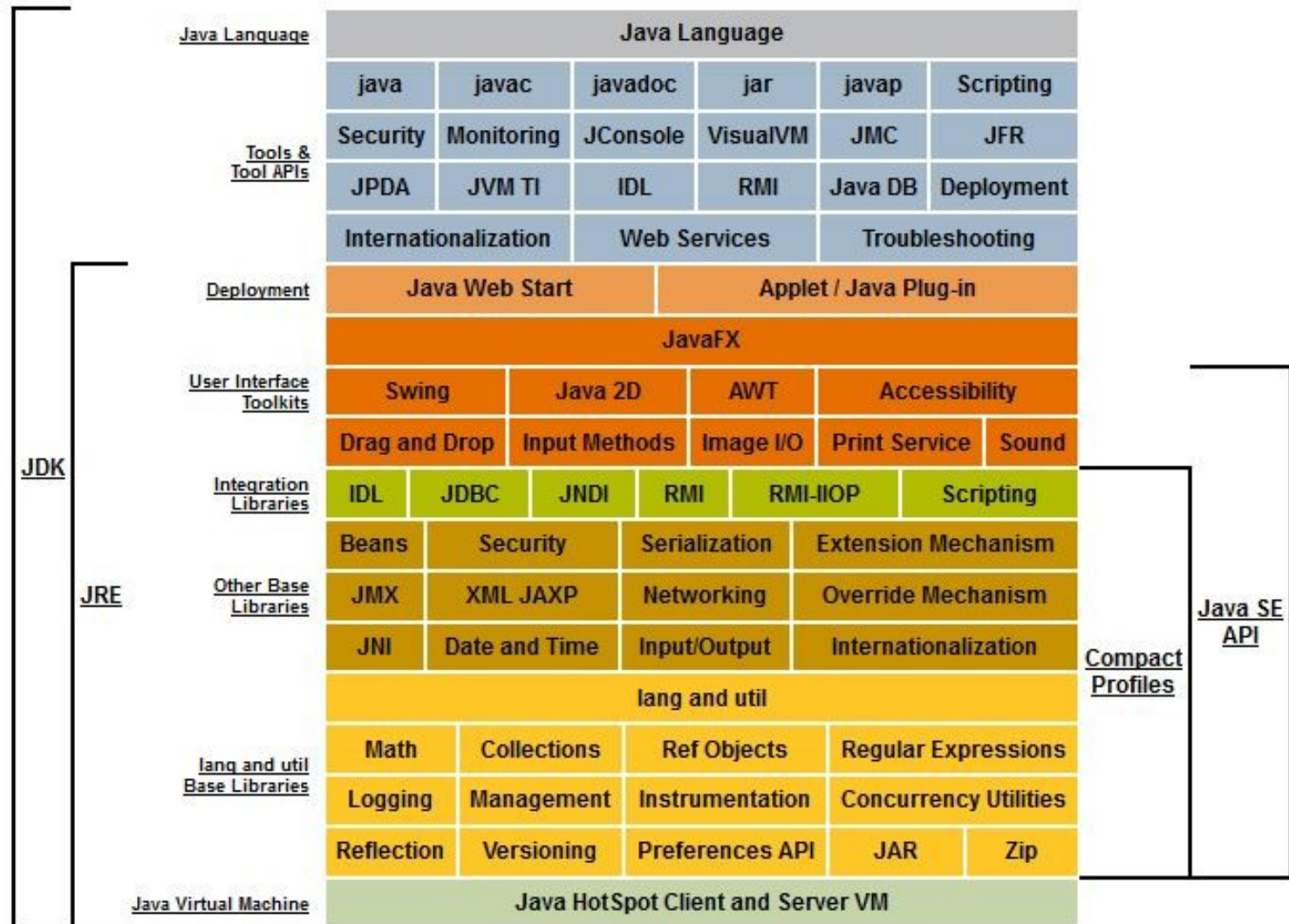
Java SE (3)

- A következő verzió a Java SE 22, melyet a JDK 22 implementál.
 - A JDK 22 2024. március 19-én kerül kiadásra.
 - Lásd: <https://openjdk.org/projects/jdk/22/>
- Specifikáció:
 - *JSR 397: Java SE 22*
<https://jcp.org/en/jsr/detail?id=397>
- API dokumentáció:
https://download.java.net/java/early_access/jdk22/docs/api/

Java SE (4)

- Java SE implementációk két fajtája:
 - Java futtató környezet (*Java Runtime Environment*) (JRE)
 - Java fejlesztői készlet (*Java Development Kit*) (JDK)
- A JDK 11-től kezdve nincs önálló JRE!

Java SE (5)



Java futtató környezet (JRE) (1)

- A Java programozási nyelven írt programok végrehajtásához szükséges állományokat tartalmazza: egy virtuális gépet, osztálykönyvtárakat és további kapcsolódó állományokat.
- Implementációk:
 - *OpenJDK* (operációs rendszer: Linux, macOS, Windows; licenc: GPLv2 + *Classpath Exception*)
<https://openjdk.org/> <https://github.com/openjdk/jdk>

Java fejlesztői készlet (JDK) (1)

- Fejlesztői környezet alkalmazások és komponensek a Java programozási nyelven történő készítéséhez.
 - Tartalmaz egy teljes Java futtató környezetet (JRE), valamint eszközöket Java nyelven írt alkalmazások fejlesztéséhez, teszteléséhez és monitorozásához.
 - Lásd: *Java Development Kit Version 21 Tool Specifications*
<https://docs.oracle.com/en/java/javase/21/docs/specs/man/>

Java fejlesztői készlet (JDK) (2)

- Nem szabad implementációk:
 - *Oracle JDK* (operációs rendszer: Linux, macOS, Windows)
<https://www.oracle.com/java/technologies/downloads/>
 - Licenc:
 - **Oracle JDK 8/11:** *Oracle Technology Network License Agreement for Oracle Java SE*
<https://www.oracle.com/downloads/licenses/javase-license1.html>
 - **Oracle JDK 21:** *Oracle No-Fee Terms and Conditions*
<https://www.oracle.com/downloads/licenses/no-fee-license.html>
 - *Azul Platform Core* (korábban *Zulu Enterprise*) (operációs rendszer: Linux, macOS, Solaris, Windows)
<https://www.azul.com/downloads/>

Java fejlesztői készlet (JDK) (3)

- Szabad és nyílt forrású implementációk (licenc: GPLv2 + *Classpath Exception*):
 - Oracle OpenJDK (operációs rendszer: Linux, macOS, Windows; licenc: GPLv2 + *Classpath Exception*) <https://jdk.java.net/>
 - Amazon Corretto (operációs rendszer: Linux, macOS, Windows) <https://aws.amazon.com/corretto/>
 - Azul Zulu Builds of OpenJDK (operációs rendszer: Linux, macOS, Windows) <https://www.azul.com/downloads/zulu/>
 - Eclipse Temurin (operációs rendszer: Linux, macOS, Windows) <https://adoptium.net/>
 - IBM Semeru Runtime (operációs rendszer: Linux, macOS, Windows, AIX) <https://developer.ibm.com/languages/java/semeru-runtimes/>
 - Microsoft Build of OpenJDK (operációs rendszer: Linux, macOS, Windows) <https://www.microsoft.com/openjdk>
 - ...

Java fejlesztői készlet (JDK) (4)

- Lásd még:
 - Stephen Colebourne. *Time to look beyond Oracle's JDK*. 3 September 2018.
<https://blog.joda.org/2018/09/time-to-look-beyond-oracles-jdk.html>
 - *Which Version of JDK Should I Use?*
<https://whichjdk.com/>

OpenJDK (1)

- „Az OpenJDK közösség fejlesztők egy olyan társulása, akik a JCP által definiált Java SE jelenlegi és jövőbeli verzióinak nyílt forrású implementációin és ezekhez szorosan kapcsolódó projekteken működnek együtt.”
 - Lásd: *OpenJDK Bylaws – Preamble*
<https://openjdk.org/bylaws>
- Webhely: <https://openjdk.org/>

OpenJDK (2)

- Projektek:
 - *Code Tools* <https://openjdk.org/projects/code-tools/>
 - *jmh* (*Java Microbenchmark Harness*) <https://github.com/openjdk/jmh>
 - *HotSpot* <https://openjdk.org/groups/hotspot/>
<https://wiki.openjdk.org/display/HotSpot>
 - *Mission Control* <https://openjdk.org/projects/jmc/>
<https://github.com/openjdk/jmc>
 - *OpenJFX* (*JavaFX*) <https://openjfx.io/>
<https://wiki.openjdk.org/display/OpenJFX/>
 - *Project Valhalla* <https://openjdk.org/projects/valhalla/>
 - ...

OpenJDK (3)

- 2006 májusában a *JavaOne* konferencián jelenti be a *Sun Microsystems*, hogy szabad és nyílt forrású szoftverként teszi elérhetővé a Java technológiát.
- Ennek első lépéseként 2006. november 13-án a cég a GPLv2 licenc hatálya alatt teszi közzé a Java HotSpot virtuális gép és a Java fordító (javac) forráskódját.
 - Lásd: *Sun Opens Java*
<https://web.archive.org/web/20070517164922/http://www.sun.com:80/2006-1113/feature/story.jsp>
- Végül 2007. május 8-án adja ki a cég a GPLv2 hatálya alatt a JDK közel teljes forráskódját.
 - Lásd:
 - *Open JDK is here!* (2007. május 8.)
<https://mail.openjdk.org/pipermail/announce/2007-May/000000.html>
 - *OpenJDK* <https://web.archive.org/web/20070510033950/http://openjdk.java.net:80/>

OpenJDK (4)

- Az OpenJDK kódbázisa:
<https://github.com/openjdk/jdk>
 - Az összes OpenJDK *build* ezen a kódbázison alapul.
- Az OpenJDK összeállítása forráskódból:
<https://openjdk.org/groups/build/doc/building.html>

OpenJDK (5)

- A 7-es számúval kezdve valamennyi Java SE verzió hivatalos referencia implementációja és az Oracle JDK-ja is az OpenJDK-n alapul:
 - *Java SE 7 Reference Implementations*
<https://jdk.java.net/java-se-ri/7>
 - ...
 - *Java SE 21 Reference Implementations*
<https://jdk.java.net/java-se-ri/21>

Oracle JDK és OpenJDK (1)

- A JDK 11 előtt:
 - A két JDK forráskódja nagymértékben azonos, az Oracle JDK azonban tartalmaz nem nyílt forrású üzleti funkciókat is.
 - Lásd: *Java Components – Commercial Features*
<https://web.archive.org/web/20180825011439/https://docs.oracle.com/java/components/index.html>
 - Példa: *JDK Mission Control (jmc)*
<https://www.oracle.com/java/technologies/jdk-mission-control.html>
 - Az Oracle azt ígérte, hogy nyílt forrásúként teszi elérhetővé az Oracle JDK üzleti funkcióit.
 - A cél a különbségek csökkentése az Oracle JDK és OpenJDK között.
 - Lásd: Mark Reinhold. *Accelerating the JDK release cadence*. September 6, 2017.
<https://mail.openjdk.org/pipermail/discuss/2017-September/004281.html>

Oracle JDK és OpenJDK (2)

- A JDK 11-től:
 - Az Oracle JDK és az OpenJDK funkcionálisan azonos és egymással felcserélhető.
 - Néhány különbség van, ezek egy része csupán felületi (eltérő például a `java -version` parancs kimenete).
 - Eltérő a csomagolás módja.
 - A felhasználási feltételek azonban lényegesen eltérőek!
 - Lásd: Donald Smith. *Oracle JDK Releases for Java 11 and Later*. September 11, 2018.
<https://blogs.oracle.com/java-platform-group/oracle-jdk-releases-for-java-11-and-later>

A Java SE kiadási modelljének változása (1)

- Lásd:
 - Mark Reinhold. *Moving Java Forward Faster*. September 6, 2017.
<https://mreinhold.org/blog/forward-faster>
 - Mark Reinhold. *Accelerating the JDK release cadence*. September 6, 2017.
<https://mail.openjdk.org/pipermail/discuss/2017-September/004281.html>
 - Donald Smith. *Faster and Easier Use and Redistribution of Java SE*. September 6, 2017.
<https://blogs.oracle.com/java-platform-group/faster-and-easier-use-and-redistribution-of-java-se>

A Java SE kiadási modelljének változása (2)

- A Java SE és a JDK nagy, szabálytalan és eléggé kiszámíthatatlan lépésekben fejlődött.

Verzió	Megjelenés dátuma	Az előző kiadás óta eltelt idő
JDK 1.0	1996-01-23	
JDK 1.1	1997-02-19	1 év 1 hónap
J2SE 1.2	1998-12-08	1 év 10 hónap
J2SE 1.3	2000-04-08	1 év 4 hónap
J2SE 1.4	2002-02-06	1 év 10 hónap
J2SE 5.0	2004-10-04	2 év 8 hónap
Java SE 6	2006-12-23	2 év 3 hónap
Java SE 7	2011-07-28	4 év 7 hónap
Java SE 8	2014-03-18	2 év 8 hónap
Java SE 9	2017-09-21	3 év 6 hónap

A Java SE kiadási modelljének változása (3)

- Egy fő kiadás tipikusan egy vagy néhány jelentős újdonságot tartalmaz.
 - A jelentős újdonságok kifejlesztése időigényes, miattuk a kisebb újdonságok megjelenése is tolódik.
- Hogy a Java versenyképes maradhasson a gyorsabb tempóban fejlődő platformokkal, nem elég fejlődnie, azt szaporábban is kell tennie.

A Java SE kiadási modelljének változása (4)

- A fejlesztők a gyors ütemű innovációt szeretik, a vállalatok inkább a stabilitást, azonban mindenki szereti a rendszeres és kiszámítható kiadásokat.
- Kompromisszumos megoldásként egy olyan fejlesztési modellt javasoltak, melyben kétfévente követik egymást a fő kiadások.
 - Lásd: Mark Reinhold. *Project Jigsaw: Late for the train*. July 17, 2012. <https://mreinhold.org/blog/late-for-the-train>
 - Ez a modell nem működött a gyakorlatban, és visszatekintve lassú is.

A Java SE kiadási modelljének változása (5)

- A javasolt kiadási modellt más platformok és különféle operációs rendszer disztribúciók kiadási modellje ihlette:
 - **Fő kiadások félévente:**
 - A fő kiadások bármilyen újdonságot tartalmazhatnak, az új és továbbfejlesztett API-kon túl akár nyelvi és a Java virtuális géppel kapcsolatos újításokat is.
 - A fő kiadások minden év márciusában és szeptemberében jelennek meg.
 - Az első 2018 márciusában érkezik.
 - **Frissítések negyedévente:**
 - A frissítések kizárólag biztonsági problémák és hibák javításait fogják tartalmazni.
 - Minden fő kiadáshoz két frissítés kerül kiadásra a következő fő kiadás előtt.
 - Minden év januárjában, áprilisában, júliusában és októberében jelennek meg.
 - **Hosszú távon támogatott (*long term support*, LTS) kiadások háromévente:**
 - Ezekhez legalább 3 évig lesznek frissítések.
 - Az első LTS kiadás (Java SE 11) 2018 szeptemberében jelent meg.
 - A három éves ciklust később egy két éves ciklusra cserélték.

A Java SE kiadási modelljének változása (6)

- Így a változás üteme összességében nagyjából olyan lesz, mint eddig.
 - A fő kiadások azonban az eddigiekhez képest kevesebb újdonsággal szolgálnak majd, így könnyebb lesz ezekhez alkalmazkodni.
 - A gyors innovációt kedvelő fejlesztők így a lehető leghamarabb élvezhetik az újdonságokat.
 - A stabilitást kedvelő vállalatok a hosszú távon támogatott kiadásokat használhatják.

A Java SE kiadási modelljének változása (7)

- A Java SE fejlesztésének ütemterve:
 - *Oracle Java SE Support Roadmap*
<https://www.oracle.com/technetwork/java/java-se-support-roadmap.html>

A Java SE kiadási modelljének változása (8)

- Egy új fő kiadás megjelenésétől elavultnak tekintendő az előző nem LTS kiadás.
 - Az elavult nem LTS kiadások nem támogatottak többé, tehát nem kapnak további frissítéseket sem.
- Az Oracle JDK LTS kiadásaihoz az Oracle legalább 8 évig biztosít frissítéseket.
 - Ezen frissítések üzleti célú felhasználásához üzleti licenc lehet szükséges!

A Java SE kiadási modelljének változása (9)

- 2021-ben az Oracle megváltoztatta a JDK LTS kiadások ütemét, három év helyett két évente érkeznek.
 - Lásd:
 - Donald Smith. *Moving the JDK to a Two Year LTS Cadence*. September 14, 2021.
<https://blogs.oracle.com/java/post/moving-the-jdk-to-a-two-year-lts-cadence>

Az Oracle JDK licencének változása (1)

- A Java SE 11 előtt az Oracle JDK és JRE a BCL hatálya alatt került terjesztésre.
 - A BCL megengedi az Oracle JDK és JRE felhasználását nem szabad szoftverekhez és azokkal együtt történő terjesztését.
 - Lásd:
 - *Oracle Binary Code License Agreement for the Java SE Platform Products*
<https://www.oracle.com/downloads/licenses/binary-code-license.html>

Az Oracle JDK licencének változása (2)

- A Java SE 11-től a Java SE 16-ig bezárólag az Oracle JDK ingyenesen az alábbi licenc hatálya alatt érhető el:
 - *Oracle Technology Network License Agreement for Oracle Java SE*
<https://www.oracle.com/downloads/licenses/javase-license1.html>
 - A licenc a következő célokra engedélyezi az ingyenes felhasználást: személyes asztali használat, fejlesztés és tesztelés, demonstráció, oktatás.
 - Azonban az üzleti célú felhasználásokhoz üzleti licenc szükséges.

Az Oracle JDK licencének változása (3)

- A Java SE 17-től kezdve az Oracle JDK az alábbi ingyenes licenc hatálya alatt érhető el, mely megengedi az üzleti célú felhasználást is:
 - *Oracle No-Fee Terms and Conditions (NFTC)*
<https://www.oracle.com/downloads/licenses/no-fee-license.html>
 - Lásd még: Donald Smith. *Introducing the Free Java License*. September 14, 2021.
<https://blogs.oracle.com/java/post/free-java-license>

Az Oracle JDK licencének változása (4)

- A Java SE 8-hoz az Oracle határozatlan ideig biztosít ingyenes frissítéseket magánfelhasználók számára.
- 2018-tól az Oracle által a Java SE 8-hoz és 11-hez kiadott frissítések üzleti célú felhasználásához is kereskedelmi licenc szükséges.
- A Java SE 17-től kezdve az LTS kiadások az NPTC hatálya alatt kapnak frissítéseket a következő LTS kiadás után még egy évig.
 - Ezután viszont a további frissítések üzleti célú felhasználásához üzleti licenc szükséges.

Az Oracle JDK licencének változása (5)

- További információk:
 - *Oracle Java SE Licensing FAQ*
<https://www.oracle.com/my/java/technologies/javase/jdk-faqs.html>
 - *Oracle Java SE Universal Subscription*
<https://www.oracle.com/java/java-se-subscription/>
 - *Oracle Java SE Universal Subscription FAQ*
<https://www.oracle.com/java/technologies/java-se-subscription-faq.html>
 - *Oracle Java SE Support Roadmap*
<https://www.oracle.com/technetwork/java/java-se-support-roadmap.html>

OpenJDK frissítések (1)

- *JDK Updates Project*

<https://openjdk.org/projects/jdk-updates/>

- A projekt célja frissítések fejlesztése az OpenJDK projekthez.
- Frissítések az Oracle által is használt negyedéves rendszerességgel kerülnek kiadásra.
- *JDK17u: OpenJDK 17 Updates*
<https://wiki.openjdk.org/display/JDKUpdates/JDK+17u>
- *JDK 21u: OpenJDK 21 Updates*
<https://wiki.openjdk.org/display/JDKUpdates/JDK+21u>

OpenJDK frissítések (2)

- Az OpenJDK projektnél nem létezik a hosszú távú támogatás fogalma.
- Az (Open)JDK gyártóknak azonban saját LTS kiadási stratégiájuk lehet.
 - Példa:
 - Amazon Corretto: *Amazon Corretto FAQs* <https://aws.amazon.com/corretto/faqs/>
 - Eclipse Temurin: *Temurin Support* <https://adoptium.net/support/>
- Lásd:
 - Azul Systems. *What Does Long-Term Support Mean for OpenJDK?* July 15, 2019. <https://www.javacodegeeks.com/2019/07/long-term-support-mean-openjdk.html>

Java EE (1)

- API-t és futtató környezetet biztosít többretegű, skálázható, megbízható és biztonságos szerver oldali alkalmazások fejlesztéséhez és futtatásához.
- A Java SE platformra épül.
- Webhely:
 - <https://www.oracle.com/java/technologies/java-ee-glance.html>
 - Legacy Java EE Home <https://javaee.github.io/>

Java EE (2)

- Az utolsó verzió a Java EE 8, mely 2017. szeptember 21-én jelent meg.
 - Lásd: *Oracle Announces Java SE 9 and Java EE 8*
<https://web.archive.org/web/20181112224512/https://www.oracle.com/corporate/pressrelease/java-se-9-and-ee-8-092117.html>
- Specifikáció:
 - JSR 366: *Java Platform, Enterprise Edition 8 (Java EE 8) Specification* <https://jcp.org/en/jsr/detail?id=366>
- API dokumentáció:
<https://javaee.github.io/javaee-spec/javadocs/>

Java EE (3)

- Java EE 8 referencia implementáció:
 - *Eclipse Glassfish* (licenc: *Eclipse Public License 2.0/GPLv2*) <https://glassfish.org/>
<https://github.com/eclipse-ee4j/glassfish>
- További Java EE implementációk:
 - Lásd: *Java EE Compatibility*
<https://www.oracle.com/java/technologies/compatibility-jsp.html>

Java EE (4)

- 2016-ban a fejlesztői közösség számára megkérdőjeleződik az Oracle a Java EE iránti elkötelezettsége.
 - Lásd: *Java EE Guardians*
<https://web.archive.org/web/20190510105354/https://javaee-guardians.io/>
- 2017 augusztusában Oracle bejelenti, hogy másnak adná át vezető szerepét a Java EE fejlesztésében.
 - Lásd: *Opening Up Java EE*. August 17, 2017.
<https://blogs.oracle.com/theaquarium/post/opening-up-java-ee>
- 2017 szeptemberében az Oracle bejelenti, hogy a Java EE fejlesztését az *Eclipse Foundation*-nek adja át.
 - Lásd: *Opening Up Java EE – An Update*. September 12, 2017.
<https://blogs.oracle.com/theaquarium/post/opening-up-java-ee-an-update>
- A Java EE fejlesztését továbbvivő Eclipse projekt:
 - *Eclipse Enterprise for Java (EE4J)* <https://projects.eclipse.org/projects/ee4j>
<https://github.com/eclipse-ee4j/ee4j>

Java EE (5)

- További ajánlott irodalom:
 - Reza Rahman. *Java EE 8 – A Bird's Eye View*. 2018.
https://www.eclipse.org/community/eclipse_newsletter/2018/november/birdseyejavaee8.php

Jakarta EE (1)

- 2018. február 26-án bejelentésre került, hogy a korábban Java EE-ként ismert technológia Jakarta EE-re lett átnevezve.
 - Lásd: Mike Milinkovich. *And the Name Is...* February 26, 2018.
<https://eclipse-foundation.blog/2018/02/26/and-the-name-is/>
- Webhely: <https://jakarta.ee/>
 - Specifikációk: <https://jakarta.ee/specifications/>

Jakarta EE (2)

- Az aktuális verzió a 2022. szeptember 13-án kiadott Jakarta EE 10.
 - Lásd: *Jakarta EE 10 Brings Java Development Into the Modern Cloud Native Era*
<https://eclipse-foundation.blog/2022/09/22/jakarta-ee-10-brings-java-development-into-the-modern-cloud-native-era/>
- Nem kompatibilis visszafelé a Jakarta EE 8 és Java EE 8 platformokkal.
- API dokumentáció:
<https://jakarta.ee/specifications/platform/10/apidocs/>

Jakarta EE (3)

- Implementációk:
 - *Eclipse GlassFish* (licenc: *Eclipse Public License 2.0/GPLv2 + Classpath Exception*) <https://glassfish.org/>
<https://github.com/eclipse-ee4j/glassfish>
 - *Wildfly* (licenc: LGPLv2.1) <https://wildfly.org/>
<https://github.com/wildfly/wildfly>
 - ...
- Lásd: *Jakarta EE Compatible Products*
<https://jakarta.ee/compatibility/>

Jakarta EE (4)

- További információk:
 - *Jakarta EE Tutorial*
<https://eclipse-ee4j.github.io/jakartaee-tutorial/>
<https://github.com/eclipse-ee4j/jakartaee-tutorial>
 - *Jakarta EE Ambassadors*
<https://jakartaee-ambassadors.io/>
 - *Jakarta Blogs* <https://jakartablogs.ee/>

Java Community Process (JCP) (1)

- 1998-ban bevezetett nyílt, a részvételt elősegítő folyamat a Java-hoz kötődő technológiai specifikációk, referencia implementációk és tesztkészletek fejlesztésére és felülvizsgálatára. A JCP program a nemzetközi Java fejlesztői közösséggel együttműködve segíti elő a Java platform fejlődését.
 - Lásd: *The Java Community Process Program – Introduction – FAQ*
<https://www.jcp.org/en/introduction/faq#jcp>
- Webhely: <https://www.jcp.org/>

Java Community Process (JCP) (2)

- A JCP mindenki számára nyílt.
 - Bárki – ideértve a magánszemélyeket, cégeket és nonprofit szervezeteket – számára a részvétel több szintje adott (megfigyelő, regisztrált felhasználó, tag).
 - A részvétel minden formája díjmentes.
- Lásd: *The Java Community Process Program – Participation – Overview: Getting Involved*
<https://www.jcp.org/en/participation/overview>

Java Community Process (JCP) (3)

- További ajánlott irodalom:
 - Simon Ritter. *Keeping The Community In The Java Community Process (JCP)*. November 21, 2016.
<https://www.sitepoint.com/keeping-community-in-java-community-process-jcp/>

Java Specification Request (JSR) (1)

- Egy vagy több tag által a JCP-hez benyújtott dokumentum, mely egy új specifikáció kifejlesztését vagy egy létező specifikáció jelentős átdolgozását indítványozza.
 - A JSR az ezekből az indítványokból eredő specifikáció fejlesztési munkát is jelenti.
 - Lásd: *The Java Community Process Program – Introduction – FAQ*
<https://www.jcp.org/en/introduction/faq#jsr>
- Az összes JSR: <https://www.jcp.org/en/jsr/all>

Java Specification Request (JSR)

(2)

- Bárki, akinek internet-hozzáférése van, átnézheti és véleményezheti a JSR-eket.
- Csak JCP tagok vehetnek részt szakértői csoportokban vagy járulhatnak hozzá közreműködőként egy JSR-hez.
 - Bárki jelentkezhet tagnak, a JCP tagság díjmentes.

Java Specification Request (JSR)

(3)

- Példa:
 - *JSR 376: Java Platform Module System*
<https://jcp.org/en/jsr/detail?id=376>
 - *JSR 385: Units of Measurement API 2.0*
<https://jcp.org/en/jsr/detail?id=385>
 - Referencia implementáció:
<https://github.com/unitsofmeasurement/indriya>
 - *JSR 396: Java SE 21* <https://jcp.org/en/jsr/detail?id=396>
 - *JSR 397: Java SE 22* <https://jcp.org/en/jsr/detail?id=397>

JDK Enhancement Proposal (JEP)

- Egy folyamat a JDK fejlesztésére irányuló javaslatok kezelésére, melyet az *OpenJDK* projektben használnak.
- Elsődleges célja javaslatok egy olyan rendszeresen frissített listájának előállítás, mely hosszú távú ütemtervként szolgálhat a JDK kiadásához.
 - Lásd:
 - *JEP 1: JDK Enhancement-Proposal & Roadmap Process*
<https://openjdk.org/jeps/1>
- Az összes JEP:
 - *JEP 0: JEP Index* <https://openjdk.org/jeps/0>
- Nem helyettesíti a JCP-t!

Előzetes lehetőségek (1)

- Egy előzetes lehetőség (*preview feature*) a Java nyelv, a virtuális gép vagy a Java SE API egy új lehetősége, mely pontosan meghatározott, teljesen implementált, de még nem végleges.
- Azért elérhető egy JDK fő kiadásban, hogy a valós használatból kapcsolatos visszajelzéseket generáljon a fejlesztőktől, mely egy jövőbeli Java SE platformban történő állandósulásához vezethet.
- Lásd:
 - *JEP 12: Preview Features* <https://openjdk.org/jeps/12>

Előzetes lehetőségek (2)

- Példák:
 - **JDK 21:**
 - *JEP 430: String Templates (Preview)*
<https://openjdk.org/jeps/430>
 - *JEP 445: Unnamed Classes and Instance Main Methods (Preview)* <https://openjdk.org/jeps/445>
 - **JDK 22:**
 - *JEP 459: String Templates (Second Preview)*
<https://openjdk.org/jeps/459>

Előzetes lehetőségek (3)

- A JDK parancssori eszközöknek (például `java`, `javac`, `jshe11`) a `--enable-preview` parancssori opciót kell megadni az előzetes lehetőségek engedélyezéséhez.
 - Lásd: *Java Language Updates – Preview Features*
<https://docs.oracle.com/en/java/javase/21/language/preview-language-and-vm-features.html>

Előzetes lehetőségek (4)

- Apache Maven:
 - Fordításhoz az alábbi módon lehet megadni a `--enable-preview` parancssori opciót a *Maven Compiler Plugin* számára:

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.12.1</version>
  <configuration>
    <compilerArgs>--enable-preview</compilerArgs>
  </configuration>
</plugin>
```


Előzetes lehetőségek (5)

- Apache Maven:
 - Programok az *Exec Maven Plugin* segítségével történő végrehajtásához a `.mvn/jvm.config` állományban lehet megadni a `--enable-preview` parancssori opciót.
 - Lásd: *Configuring Apache Maven – .mvn/jvm.config file*
<https://maven.apache.org/configure.html#mvn-jvm-config-file>

Előzetes lehetőségek (6)

- Példa: *JEP 430: String Templates (Preview)* (JDK 21)

```
var width = 1024;  
var height = 768;  
var s = STR."\\{width} * \\{height} = \\{width * height}";  
System.out.println(s);  
// 1024 * 768 = 786432
```

Előzetes lehetőségek (7)

- Példa: *JEP 430: String Templates (Preview)* (JDK 21)

```
var now = LocalDate.now();
var s = STR. """
{
    "meaningOfLife": \{2 * 3 * 7\},
    "java": {
        "vendor": "\{System.getProperty("java.vendor")\}",
        "version": "\{System.getProperty("java.version")\}"
    },
    "date": "\{now\}",
    "isLeapYear", \{now.isLeapYear()\}
}
""";
System.out.println(s);
```

Kapcsolódó programozási nyelvek és platformok

- *JavaFX/OpenJFX* <https://openjfx.io/>
- *Android* <https://www.android.com/>
- *Apache Groovy* <https://groovy-lang.org/>
- *Kotlin* <https://kotlinlang.org/>
- *Scala* <https://www.scala-lang.org/>
- *Clojure* <https://clojure.org/>
- *GraalVM* <https://www.graalvm.org/>
- *Spring Framework* <https://spring.io/projects/spring-framework>
- *Spring Boot* <https://spring.io/projects/spring-boot>

JavaFX (1)

- A Java SE-n alapuló platform olyan vastag kliens alkalmazások készítéséhez, melyek következetesen működnek különféle eszközökön.
- A Java SE 7 Update 6-tól az Oracle Java SE implementációjának része (volt).
- Dokumentáció:
<https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>

JavaFX (2)

- A JDK 11-ben eltávolításra került, helyét az OpenJFX hivatott betölteni.
 - Lásd:
 - Donald Smith. *The Future of JavaFX and Other Java Client Roadmap Updates*. March 7, 2018.
<https://blogs.oracle.com/java-platform-group/the-future-of-javafx-and-other-java-client-roadmap-updates>
 - *JDK 11 Release Notes*
<https://www.oracle.com/java/technologies/javase/11-relnote-issues.html>

OpenJFX (1)

- A Java-ra épülő nyílt forrású kliens alkalmazási platform asztali, mobil és beágyazott eszközökhöz.
- Az OpenJDK projekt része.
- Webhely:
<https://wiki.openjdk.org/display/OpenJFX/>
<https://openjfx.io/> <https://github.com/openjdk/jfx>

OpenJFX (2)

- Platform: Linux, macOS, Windows
- Licenc: *GPLv2 + Classpath Exception*
<https://openjdk.org/legal/gplv2+ce.html>
- Aktuális verzió: JavaFX 21 (követelmény: 17-es vagy későbbi JDK)
- Dokumentáció: <https://openjfx.io/openjfx-docs/>
 - API dokumentáció: <https://openjfx.io/javadoc/21/>

OpenJFX (3)

- A Gluon cég üzleti támogatást biztosít a JavaFX-hez.
<https://gluonhq.com/>
 - Támogatás mobil (Android, iOS) és beágyazott (Raspberry Pi) eszközökhöz.
- Fejlesztőeszközök:
 - Szabad és nyílt forrású:
 - *e(fx)clipse* (platform: Eclipse; licenc: *Eclipse Public License 1.0*)
<https://eclipse.dev/efxclipse/>
 - *Gluon Scene Builder* (operációs rendszer: Linux, macOS, Windows; licenc: *New BSD License*) <https://gluonhq.com/products/scene-builder/>
<https://github.com/gluonhq/scenebuilder>

Android

- A Linux kernelre épülő, a Google által fejlesztett mobil operációs rendszer és szoftver platform.
- Architektúra: *Platform Architecture* <https://developer.android.com/guide/platform/>
- Android alkalmazások C++, Java, Kotlin és Rust nyelven készíthetők. Az elsődleges nyelv a Kotlin.
 - Lásd:
 - *Application Fundamentals* <https://developer.android.com/guide/components/fundamentals.html>
 - *Android's Kotlin-first approach* <https://developer.android.com/kotlin/first>
- Az alkalmazások fordítása *Dalvik* bájtkódra történik, melyet az *Android Runtime* (ART) fordít le gépi kódra.
 - Lásd: *ART and Dalvik* <https://source.android.com/devices/tech/dalvik>
- Licenc: *Apache License 2.0*
- Webhely: <https://www.android.com/>

Apache Groovy (1)

- Objektumorientált programozási nyelv a Java platformhoz.
 - Opcionálisan típusos és dinamikus, de támogatja a statikus típuskezelést és a statikus fordítást is.
- Használható szkriptnyelvként, támogatja szakterület-specifikus nyelvek létrehozását, a futásidejű és fordítási idejű metaprogramozást valamint a funkcionális programozást.

Apache Groovy (2)

- Eltérések a Java nyelvtől: *Differences with Java*
<https://groovy-lang.org/differences.html>
- Specifikáció:
<https://groovy-lang.org/documentation.html>
- Licenc: *Apache License 2.0*
- Webhely: <https://groovy-lang.org/>
<https://github.com/apache/groovy>

Apache Groovy (3)

- „Helló, világ!” program:

```
println "Hello, World!"
```

Apache Groovy (4)

- Groovy-ban írt szoftverek:
 - **Gradle** (licenc: *Apache License 2.0*) <https://gradle.org/>
<https://github.com/gradle/gradle>
 - **Grails** (licenc: *Apache License 2.0*) <https://grails.org/>
<https://github.com/grails>
 - ...
 - Lásd:
 - <https://groovysource.com/>
 - <https://github.com/kdabir/awesome-groovy>
 - <https://github.com/trending/groovy>

Kotlin (1)

- Általános célú, statikusan típusos, az objektumorientált és funkcionális programozási paradigmákat is támogató programozási nyelv.
- A kliens és a szerver oldalon is használható, valamint az Android platformhoz is.
 - Hivatalosan is az Android platform programozási nyelve.
 - Lásd: *Kotlin on Android FAQ* <https://developer.android.com/kotlin/faq.html>
- Úgy tervezték, hogy interoperábilis legyen legyen a Java programozási nyelvvel.
 - Java kód hívható Kotlin kódból, és viszont, lásd:
 - *Calling Java code from Kotlin* <https://kotlinlang.org/docs/java-interop.html>
 - *Calling Kotlin from Java* <https://kotlinlang.org/docs/java-to-kotlin-interop.html>

Kotlin (2)

- Kotlin kód Java bájt kódra vagy JavaScript-re is fordítható.
- *Kotlin/Native*:
 - Kotlin kód gépi kódra fordítására szolgáló technológia.
 - Lásd:
<https://github.com/JetBrains/kotlin/tree/master/kotlin-native>
<https://kotlinlang.org/docs/native-overview.html>

Kotlin (3)

- Összehasonlítás a Java programozási nyelvvel:
 - See: *Comparison to Java Programming Language*
<https://kotlinlang.org/docs/comparison-to-java.html>
 - A Java-hoz képest tömörebb szintaxisa van, egyes becslések szerint 40%-kal kevesebb kódsort igényel.
 - Lásd: *Kotlin FAQ* <https://kotlinlang.org/docs/faq.html>
- Specifikáció: *Kotlin Language Specification*
<https://kotlinlang.org/spec/>
- Licenc: *Apache License 2.0*
- Webhely: <https://kotlinlang.org/>
<https://github.com/JetBrains/kotlin/>

Kotlin (4)

- „Helló, világ!” program:

- Lásd:

https://play.kotlinlang.org/byExample/01_introduction/01_Hello%20world

```
fun main() {  
    println("Hello, World!")  
}
```

Kotlin (5)

- Kotlin-ban írt szoftverek:
 - **Android Studio** (licenc: nem szabad) <https://developer.android.com/studio>
 - **IntelliJ IDEA** (licenc: *Apache License 2.0*/nem szabad) <https://www.jetbrains.com/idea/>
<https://github.com/JetBrains/intellij-community>
 - **Kotlin** (licenc: *Apache License 2.0*) <https://kotlinlang.org/>
<https://github.com/JetBrains/kotlin>
 - ...
 - Lásd:
 - <https://kotlin.link/>
 - <https://github.com/trending/kotlin>

Scala (1)

- Az objektumorientált és a funkcionális programozási paradigmát ötvöző Java-szerű, statikusan típusos programozási nyelv.
- Úgy tervezték, hogy interoperábilis legyen legyen a Java programozási nyelvvel.
 - Scala osztályok meghívhatnak Java metódusokat, létrehozhatnak Java objektumokat, származtathatók Java osztályokból, implementálhatnak Java interfészeket.
- A Scala kód fordítása Java bájtkódra történik.
- Lásd: *Tour of Scala*
<https://docs.scala-lang.org/tour/tour-of-scala.html>

Scala (2)

- Specifikáció: *Scala Language Specification*
<https://www.scala-lang.org/files/archive/spec/3.4/>
- Licenc: *New BSD License*
- Webhely: <https://www.scala-lang.org/>
<https://github.com/scala/scala>

Scala (3)

- „Helló, világ!” program:
 - Lásd: *Your First Scala Program*
<https://docs.scala-lang.org/scala3/book/taste-hello-world.html>

```
@main def hello() = println("Hello, World!")
```

Scala (4)

- Scala-ban írt szoftverek:
 - **Apache Kafka** (licenc: *Apache License 2.0*) <https://kafka.apache.org/>
<https://github.com/apache/kafka>
 - **Apache Spark** (licenc: *Apache License 2.0*) <https://spark.apache.org/>
<https://github.com/apache/spark>
 - **Play Framework** (licenc: *Apache License 2.0*)
<https://www.playframework.com/>
<https://github.com/playframework/playframework>
 - **sbt** (licenc: *Apache License 2.0*) <https://www.scala-sbt.org/>
<https://github.com/sbt/sbt>
 - ...
 - Lásd: <https://github.com/trending/scala>

Clojure (1)

- Egy dinamikus és általános célú funkcionális programozási nyelv.
- A Lisp egy Java bájtkódra fordított dialektusa.
- Lásd: *Rationale*
<https://clojure.org/about/rationale>

Clojure (2)

- Specifikáció: -
- Dokumentáció:
https://clojure.org/guides/getting_started
- Licenc: *Eclipse Public License 1.0*
- Webhely: <https://clojure.org/>
<https://github.com/clojure/clojure/>

Clojure (3)

- Kapcsolódó projektek:
 - **Leiningen** (programozási nyelv: Clojure; licenc: *Eclipse Public License 1.0*) <https://leiningen.org/>
<https://codeberg.org/leiningen/leiningen>
 - Egy eszköz Clojure projektek összeállításához.
 - **ClojureScript** (programozási nyelv: Clojure; licenc: *Eclipse Public License 1.0*) <https://clojurescript.org/>
<https://github.com/clojure/clojurescript>
 - Clojure-to-JavaScript fordító.

Clojure (4)

- „Helló, világ!” program:

```
(ns hello.core)

(defn -main
  [& args]
  (println "Hello, World!"))
```

Clojure (5)

- Clojure-ban írt szoftverek:
 - **Metabase** (licenc: AGPLv3) <https://www.metabase.com/>
<https://github.com/metabase/metabase>
 - **Penpot** (licenc: *Mozilla Public License 2.0*)
<https://penpot.app/> <https://github.com/penpot/penpot>
 - ...
 - Lásd:
 - https://clojure.org/community/success_stories
 - <http://open-source.braveclojure.com/>
 - <https://github.com/trending/clojure>

GraalVM (1)

- Nagy teljesítményű JDK Java és más nyelveken írt alkalmazások végrehajtásának gyorsításához. A támogatott nyelvek közé tartoznak:
 - JVM-alapú nyelvek, mint például a Java, Scala, Kotlin és Clojure
 - LLVM-alapú nyelvek, mint például a C és C++
 - ECMAScript
 - Python
 - ...
- A GraalVM Java bájtkódból natív végrehajtható programokat tud összeállítani (`native-image` eszköz).
- Lehetővé teszi poliglott alkalmazások fejlesztését, melyekben több nyelv keveredik egy programon belül többletráfordítás nélkül.
 - A nyelvek elérhetik egymás adatszerkezeteit és meghívhatják egymás metódusait.
- Az aktuális kiadás a *GraalVM for JDK 21*, melynek alapja az Oracle JDK/OpenJDK 21.
- Webhely: <https://www.graalvm.org/> <https://github.com/oracle/graal>

GraalVM (2)

- Platform: Linux, macOS, Windows
- Kiadások:
 - *Community Edition* (CE)
 - Licenc: GPLv2
 - *Enterprise Edition* (EE)
 - Kipróbálásra és egyéb magáncélú felhasználásokra ingyenes.
- Dokumentáció:
<https://www.graalvm.org/latest/docs/>

GraalVM (3)

- Truffle:
 - A Truffle nyelvi implementációs keretrendszer egy nyílt forrású könyvtár programnyelv implementációk absztrakt szintaxisfa értelmezőként történő létrehozásához.
 - Lehetővé teszi a fejlesztők számára programnyelv-agnosztikus eszközök, például hibakeresők és profilozók létrehozását.
 - Nyelvi implementációk:
 - <https://github.com/oracle/graal/blob/master/truffle/docs/Languages.md>
 - ECMAScript: Graal.js <https://github.com/oracle/graaljs>
 - Python: GraalPy (kísérleti) <https://github.com/oracle/graalpython>
 - ...
 - Lásd:
<https://www.graalvm.org/latest/graalvm-as-a-platform/language-implementation-framework/>

Spring Framework

- Nyílt forrású alkalmazás fejlesztési keretrendszer, mellyel tetszőleges Java alkalmazások létrehozhatók, beleértve a vállalati szintű webalkalmazásokat.
 - Az alapjául szolgáló fogalmak a vezérlés megfordítása (*Inversion of Control*, IoC) és a függőség befecskendezés (*Dependency Injection*, DI).
- Licenc: *Apache License 2.0*
- Webhely: <https://spring.io/projects/spring-framework>
- Tároló: <https://github.com/spring-projects/spring-framework>
- API dokumentáció:
<https://docs.spring.io/spring-framework/docs/current/javadoc-api/>

Spring Boot

- A Spring Framework-re épülő keretrendszer, mely megkönnyíti és meggyorsítja Spring-alapú alkalmazások létrehozását és fejlesztését.
 - Szükségtelessé teszi egy csomó *boilerplate* kód és konfiguráció írását.
- Licenc: *Apache License 2.0*
- Webhely: <https://spring.io/projects/spring-boot>
- Tároló: <https://github.com/spring-projects/spring-boot>
- API dokumentáció: <https://docs.spring.io/spring-boot/docs/current/api/>
- Spring Initializr: <https://start.spring.io/>
 - Egy webalkalmazás Spring Boot projektek egy vázának generálásához.

Integrált fejlesztői környezetek (1)

- Szabad és nyílt forrású szoftverek:
 - **Apache NetBeans** (platform: Linux, macOS, Windows; licenc: *Apache License 2.0*) <https://netbeans.apache.org/> <https://github.com/apache/netbeans>
 - **Eclipse IDE for Java Developers** (platform: Linux, macOS, Windows; licenc: *Eclipse Public License v2*) <https://www.eclipse.org/>
 - **IntelliJ IDEA Community Edition** (platform: Linux, macOS, Windows; licenc: *Apache License 2.0*) <https://www.jetbrains.com/idea/>
<https://github.com/JetBrains/intellij-community>
 - **Visual Studio Code** (platform: Linux, macOS, Windows; licenc: *MIT License*)
<https://code.visualstudio.com/> <https://github.com/Microsoft/vscode>
 - Lásd:
 - *Java in Visual Studio Code* <https://code.visualstudio.com/docs/languages/java>
 - *Extension Pack for Java*
<https://marketplace.visualstudio.com/items?itemName=vscjava.vscode-java-pack>

Integrált fejlesztői környezetek (2)

- Nem szabad szoftverek:
 - ***IntelliJ IDEA Ultimate Edition*** (platform: Linux, macOS, Windows; licenc: nem szabad)
<https://www.jetbrains.com/idea/>
 - Ingyenes oktatási licencek:
<https://www.jetbrains.com/community/education/>

SDKMAN!

- Eszköz különféle szoftverfejlesztő készletek (SDK-k) párhuzamos verzióinak kezeléséhez.
 - Rendszergazda jogosultságot nem igényelve teszi lehetővé a felhasználók számára SDK-k telepítését és kezelését.
- Platform: Unix-szerű
- Licenc: *Apache License 2.0*
- Webhely: <https://sdkman.io/>
<https://github.com/sdkman/sdkman-cli>
- Elérhető SDK-k: <https://sdkman.io/jdks>
<https://sdkman.io/sdks>

JShell (1)

- A JDK parancssori eszköze Java deklarációk, utasítások és kifejezések interaktív kiértékeléséhez.
 - Úgynevezett *Read-Evaluate-Print Loop* (REPL) eszköz.
- Segíti a Java programozási nyelv elsajátítását, ismeretlen kódok és API-k kipróbálását és a prototípus gyártást.
- Egy API-t is biztosít a JShell funkciók külső kódból történő eléréséhez.
 - Lásd a `jdk.jshell` modult:
<https://docs.oracle.com/en/java/javase/21/docs/api/jdk.jshell/module-summary.html>
- A JDK 9-ben vezették be.

JShell (2)

- Kényelmi funkciók:
 - Sok gyakran használt csomag felső szintű típusai alapértelmezésben importálásra kerülnek
 - Elhagyható az utasításokat záró pontosvessző karakter
 - Automatikus kiegészítés (tabulátor billentyű)
 - ...

JShell (3)

- A JShell-be begépett Java kódokat kódrészleteknek (*snippets*) nevezzük.
- Parancsok szolgálnak a környezet vezérlésére és a kódrészek kezelésére.
 - A parancsok mindig egy / karakterrel kezdődnek, mint például:
 - `/imports`
 - `/help`
 - `/exit`
 - `/list`
 - `/vars`
 - `/methods`
 - `/types`
 - ...

JShell (4)

- IDE integráció:
 - **Apache NetBeans:** *Tools | Open Java Platform Shell*
 - **IntelliJ IDEA:** *JShell Console (Tools | JShell Console)*
<https://www.jetbrains.com/help/idea/jshell-console.html>
- Build eszköz integráció:
 - **Apache Maven:** *jshell-maven-plugin*
<https://github.com/johnpoth/jshell-maven-plugin>
 - **Gradle:** *JShell Plugin*
<https://plugins.gradle.org/plugin/com.github.mrsarm.jshell.plugin>
<https://github.com/mrsarm/jshell-plugin>
- Online eszközök:
 - <https://tryjshell.org/>

JShell (5)

- Dokumentáció:
 - *The jshell Command*
<https://docs.oracle.com/en/java/javase/21/docs/specs/man/jshell.html>
 - *Java Shell User's Guide*
<https://docs.oracle.com/en/java/javase/21/jshell/>

További ajánlott olvasnivaló

- *Java Programming – The State of Developer Ecosystem in 2023 Infographic*
<https://www.jetbrains.com/lp/devecosystem-2023/java/>