جافا - الوحدة الأولى: الخطوات الأولى وأول برنامج

جافا لغة برمجة تستخدم في أنظمة الأعمال وخدمات الويب. نكتبها بنص عادي، ثم يحوله برنامج إلى تعليمات يفهمها الحاسوب. ويكفينا للتدرب حزمة تطوير مجانية ومحرر مجاني، ويمكن تكرار كل خطوة هنا على أي حاسوب. وتنقلنا أربع خطوات من ملف فارغ إلى برنامج يعمل: الكتابة، والترجمة، والتشغيل، وقراءة الرسائل والإصلاح. وفي هذه الوحدة يستخدمها سمير، محلل لوجستيات في شركة شحن، ليطبع أول رسالة شحنة له، وتجيب كل خطوة عن سؤال عملي واحد.

1. الكتابة

يبدأ كل برنامج جافا بملف نص عادي بامتداد دوت جافا. ويقع الكود داخل صنف، ويجب أن يطابق اسم الصنف اسم الملف عندما يكون الصنف عاما. ويبدأ كل برنامج من طريقة خاصة اسمها main، ومنها تبدأ جافا التنفيذ. وداخلها تطبع عبارة println من الكائن System out سطرا من النص على الشاشة. وجافا حساسة لحالة الأحرف، فكلمة System تختلف عن system، وتنتهي معظم العبارات بفاصلة منقوطة. ويتجاهل المترجم التعليقات التي تبدأ بشرطتين مائلتين، وهي تفيد القراء من البشر. وسؤال هذه الخطوة هو: ما أصغر برنامج كامل يفعل شيئا مرئيا؟

مثال: تكتب متدربة في مصرف ملفا اسمه Welcome، وفيه صنف يحمل الاسم نفسه. وتضع داخل طريقة main عبارة println واحدة بالكلمتين Welcome aboard. ويتطابق اسم الملف مع اسم الصنف تماما، حتى في الحرف الكبير. ويصير البرنامج كله خمسة أسطر فقط.

في العمل: يفتح سمير محررا مجانيا، مثل Eclipse أو Visual Studio Code، وينشئ ملفا اسمه ShipmentLabel. ويكتب صنفا بالاسم نفسه وطريقة main وسطرا واحدا يطبع الكلمتين Shipment ready: `System.out.println("Shipment ready");`. ويرسل السطر النص الذي بين علامتي الاقتباس إلى الشاشة. ويسأل نفسه سؤال الخطوة، فتكون الإجابة هذه الأسطر القليلة.

2. الترجمة

لا يستطيع الحاسوب تنفيذ نصنا مباشرة، لذلك تترجمه أداة تسمى المترجم أولا. واسم المترجم javac، وهو يأتي مع حزمة تطوير جافا، أي JDK، التي تضم أيضا بقية الأدوات التي نحتاج إليها. يقرأ المترجم الملف المصدري، ويفحص القواعد والأنواع، ويكتب ملفا جديدا بامتداد دوت كلاس. ويحتوي ذلك الملف على bytecode، وهو مجموعة تعليمات مضغوطة لا ترتبط بنظام تشغيل واحد. وإذا خالفت الشيفرة قاعدة، توقف المترجم وذكر الملف والسطر والمشكلة، فتكتشف الأخطاء قبل أن يعمل أي شيء. والتحذير يختلف عن الخطأ: فالمترجم ينتج ملف الصنف معه، لكنه ينصح بالنظر عن كثب. وسؤال هذه الخطوة هو: هل يقبل المترجم شيفرتنا؟

مثال: يكتب فريق مالي برنامجا صغيرا لجمع المصروفات ويترجمه. يشتكي المترجم من كلمة ينقصها حرف كبير في السطر السابع. ولم يعمل شيء بعد، ولم تلمس أي بيانات. يصلح الفريق الحرف ويترجم مرة أخرى، فيظهر ملف دوت كلاس.

في العمل: يفتح سمير نافذة أوامر في مجلد ملفه ويكتب javac ثم اسم الملف. لا يظهر شيء للحظة، وهذا خبر جيد، لأن الصمت يعني النجاح. ويظهر الآن بجوار الملف المصدري ملف جديد اسمه ShipmentLabel دوت كلاس. ويجيب سمير عن سؤال الخطوة: نعم، المترجم يقبل الشيفرة.

3. التشغيل

لتشغيل البرنامج نستخدم الأمر java مع اسم الصنف، وهو يشغل آلة جافا الافتراضية، أي JVM. تحمل الآلة bytecode وتفحصه وتنفذه على الحاسوب الفعلي. ولأن bytecode واحد في كل مكان، يعمل ملف الصنف نفسه على Windows

حقوق الطبع والنشر محمية © - أكاديمية أساسيات (شركة أساسيات للخدمات الاستشارية)

و macOS و Linux ما دامت بيئة تشغيل جافا مثبتة. وهذه هي الفكرة وراء قابلية جافا المشهورة للنقل بين الأنظمة. وتستطيع الإصدارات الحديثة من جافا تشغيل ملف مصدري واحد مباشرة بأمر واحد، وهذا مفيد للتجارب الصغيرة، لكن الترجمة أولا تظهر ما يحدث فعلا. ومن المشاكل الأولى الشائعة ألا يوجد الأمر java، وهذا يعني غالبا أن النظام لا يعرف بعد مكان أدوات JDK المثبتة. وسؤال هذه الخطوة هو: أي صنف نشغل، وأين تبحث عنه آلة جافا؟

مثال: ترسل شركة برمجيات برنامجا واحدا مترجما إلى عملاء يستخدمون أنظمة تشغيل مختلفة. ويثبت كل عميل بيئة تشغيل جافا ويشغل ملف الصنف نفسه بالأمر java. وتحمل آلة جافا على كل حاسوب ملف الصنف نفسه وتؤدي العمل نفسه. ولا تعيد الشركة بناء البرنامج لكل نظام. فلديها إصدار واحد فقط للصيانة والشرح.

في العمل: يكتب سمير java ثم ShipmentLabel دون أي امتداد. تجد الآلة الصنف، وتبدأ من main، وتطبع على الشاشة السطر Shipment ready. ويصل إلى جواب سؤال الخطوة: اسم الصنف، في المجلد الذي يوجد فيه الملف المترجم. ثم يري مديره الشاشة، فيطلب المدير رسالة باسم العميل بعد ذلك.

4. قراءة الرسائل والإصلاح

يواجه المبتدئون الأخطاء كل يوم، وقراءتها بهدوء مهارة أساسية. يظهر خطأ الترجمة قبل تشغيل البرنامج ويذكر الملف والسطر. ويظهر خطأ التشغيل أثناء عمل البرنامج، وتطبع جافا stack trace، وهو قائمة بسلسلة الاستدعاءات التي أدت إلى الفشل. نقرأ الرسالة الأولى أولا، لأن خطأ مبكرا واحدا يسبب كثيرا من الشكاوى اللاحقة. ومن الأخطاء الأولى الشائعة فاصلة منقوطة ناقصة، وقوس مغلق ناقص، وحرف كبير في غير مكانه، واسم ملف لا يطابق اسم الصنف. وقراءة الرسالة بصوت عال بكلمات بسيطة، مثل في السطر الرابع هذا الاسم غير معروف، تحول كتلة حمراء مخيفة إلى بند بسيط في قائمة المهام. وسؤال هذه الخطوة هو: ماذا تقول الرسالة، وفي أي سطر؟

مثال: تنسى مهندسة دعم علامة اقتباس مغلقة، فترى عدة رسائل دفعة واحدة، كل واحدة في سطر مختلف. ولا تحاول إصلاحها كلها. بل تقرأ الرسالة الأولى وتصلحها وحدها وتترجم مرة أخرى. وتختفي معظم الرسائل الأخرى، لأن مصدرها العلامة الناقصة نفسها. فكان الخطأ المبكر الواحد سبب القائمة الحمراء كلها.

في العمل: يكتب سمير println بحرف L كبير فتصير printLn، فيقول المترجم إنه لا يجد الرمز في السطر الرابع. يقرأ رقم السطر ويكتشف الحرف الكبير ويصححه. ثم يترجم من جديد ويشغل البرنامج بنجاح. بعدها يكرر الدورة بسطر ثان يطبع مدينة الوجهة.

الخاتمة

في الخلاصة، يمر برنامج جافا بأربع خطوات: الكتابة، والترجمة، والتشغيل، وقراءة الرسائل والإصلاح. نكتب صنفا فيه طريقة main في ملف يطابق اسمه اسم الصنف. يحوله المترجم javac إلى bytecode، وتشغل JVM هذا bytecode على أي نظام فيه بيئة تشغيل جافا. والأخطاء ليست فشلا بل توجيهات، فنقرأ الرسالة الأولى ورقم السطر. ولدى سمير الآن أول برنامج يعمل، وفي المرة القادمة سيعطيه قيما يتذكرها ويحسب بها. ومن العادات الجيدة تشغيل البرنامج بعد كل تغيير صغير، ليبقى آخر تغيير هو أول مشتبه به.

المصادر

- Oracle: دروس Java، البدء (تطبيق Hello World والترجمة والتشغيل) – https://docs.oracle.com/javase/tutorial/getStarted/cupojava/index.html
- Oracle: دروس Java، البدء (تقنية Java: ما هي وكيف تعمل) – https://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html
- dev.java: تعلّم Java (موقع التعلّم الرسمي لمجتمع جافا) – https://dev.java/learn/
- Java علامة تجارية لشركة Oracle و/أو الشركات التابعة لها. هذا الفيديو مادة تدريبية مستقلة وغير تابعة لـ Oracle.
- OpenJDK: JEP 330, Launch Single-File Source-Code Programs (Java 11) – https://openjdk.org/jeps/330

حقوق الطبع والنشر محمية © - أكاديمية أساسيات (شركة أساسيات للخدمات الاستشارية)