UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

vs

GRUPO 22


Java™

1991	1995	1996	2014	2019
Sun Microsystems lanza “OAK”	Versión Alpha de JAVA	JDK 1.0	JAVA versión 8	JAVA versión 12


Kotlin

2011

JetBrains lanza la primer versión

2012

Comienza a ser de software libre

2016

Versión 1.0

2017

Soporte de Google

VS

Objetivo

Comparar el rendimiento de un algoritmo.

Se decidió llevar a cabo un algoritmo que ordena una colección de 1000 elementos para así ver su tiempo de ejecución.

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

```java
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class Main {
    public static void main(String[] args) {
        // Crear una lista de ejemplo con 1000 elementos no ordenados
        List<Integer> lista = new ArrayList<>();
        for (int i = 0; i < 1000; i++) {
            lista.add((int) (Math.random() * 1000)); // Agregar números aleatorios
        }

        // Ordenar la lista
        Collections.sort(lista);

        // Imprimir la lista ordenada
        System.out.println("Lista ordenada:");
        for (Integer elemento : lista) {
            System.out.print(elemento + " ");
        }
    }
}
```

Ejecución
0.669 seconds

Kotlin

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

```kotlin
fun main() {
    // Crear una lista de ejemplo con 1000 elementos no ordenados
    val lista = mutableListOf<Int>()
    repeat(1000) {
        lista.add((0..999).random()) // Agregar números aleatorios
    }

    // Ordenar la lista
    lista.sort()

    // Imprimir la lista ordenada
    println("Lista ordenada:")
    lista.forEach { elemento ->
        print("$elemento ")
    }
}
```

Ejecución
5.059 seconds

VS

Objetivo

Comparar el rendimiento de un algoritmo que da la cantidad de dígitos de Pi según lo indique la entrada. En este caso se pidió los 100 primero dígitos.

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

```java
import java.math.BigInteger;

final class piDigits {
  static final int L = 10;

  Run | Debug
  public static void main (String args[]) {
    int n = 100;
    int j = 0;

    PiDigitSpigot digits = new PiDigitSpigot();

    while (n > 0) {
      if (n >= L) {
        for (int i = 0; i < L; i++) System.out.print(digits.next());
        j += L;
      } else {
```

```
.\Java Codes> java piDigits 100
```

Ejecución
0.63 seconds

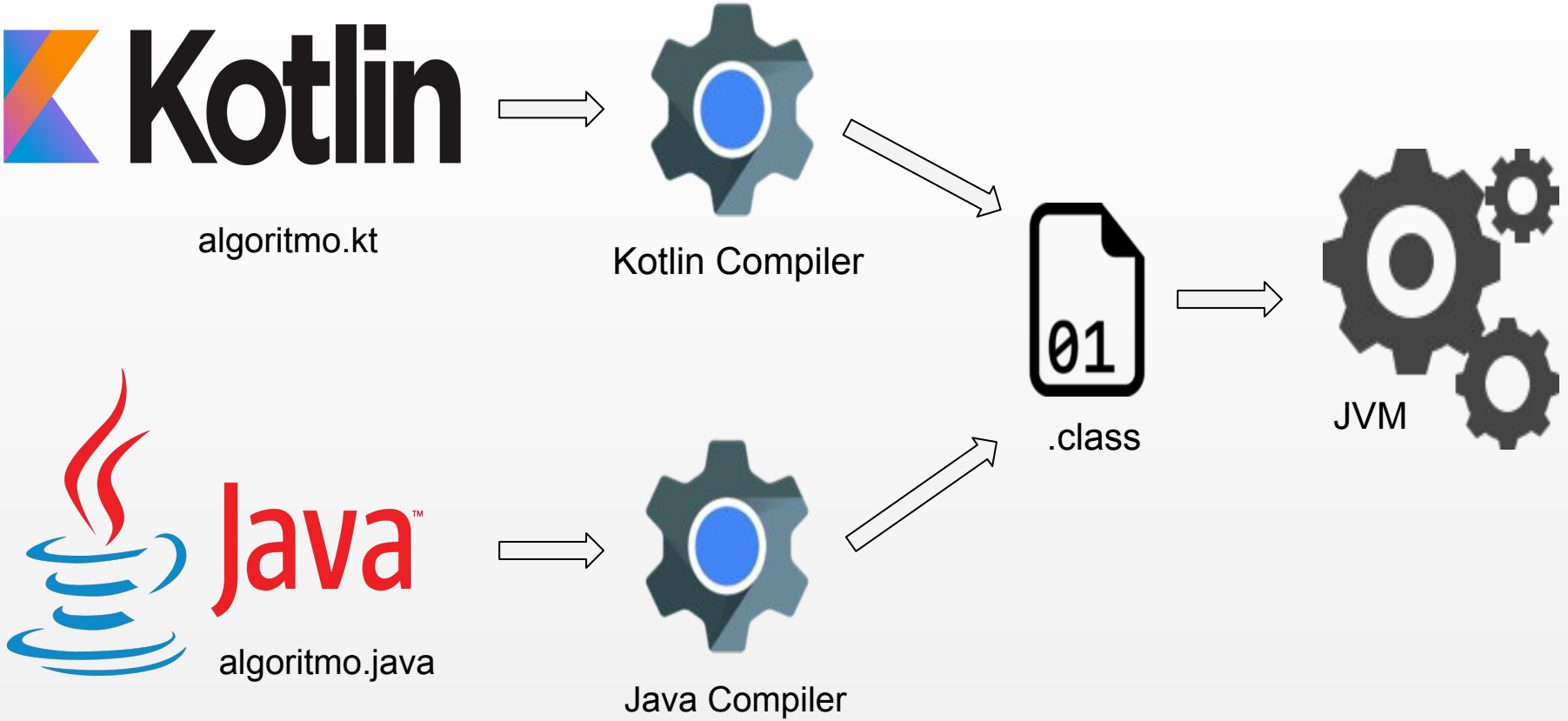
Kotlin

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

```kotlin
import java.math.BigInteger;

fun main(args: Array<String>) {
    val L = 10

    var n = if(args.size > 0) args[0].toInt() else 27
    var j = 0

    val digits = PiDigitSpigot()

    while (n > 0) {
        if (n >= L) {
            for (i in 0..L - 1) print(digits.next())
```

```
\Kotlin Codes> kotlinc piDigits.kt -include-runtime -d piDigits.jar
\Kotlin Codes> java -jar piDigits.jar 100
```

Ejecución
5.718 seconds


Kotlin
algoritmo.kt
Kotlin Compiler
01
.class
JVM
Java
algoritmo.java
Java Compiler

VS

Conclusión

Java resultó ser más rápido a la hora de su ejecución. Sin embargo Kotlin suele ser recomendado ya que es más conciso y seguro. Además puede consumir los recursos, librerías y hasta la máquina virtual de Java. Por lo que ambos lenguajes pueden convivir en un mismo proyecto.

Sin embargo, para otros tipos de desarrollo, la elección entre Java y Kotlin puede depender de varios factores.

¿Preguntas?