

# Práctica 2: Apache Thrift

**Alumno:** Raúl Rodríguez Pérez

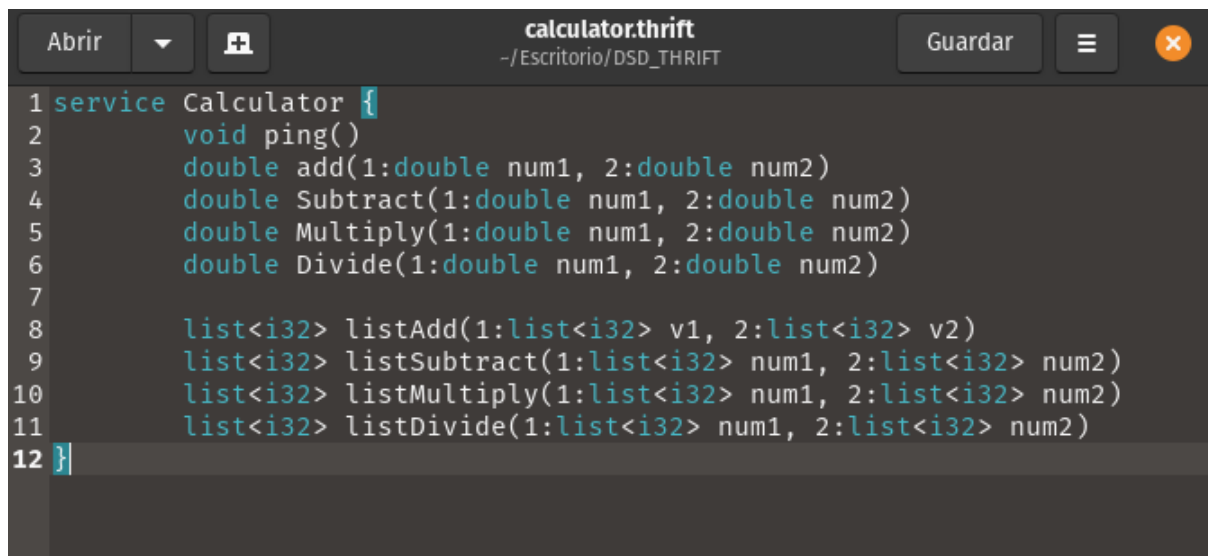
**Grupo:** dsd2

Para la realización de esta práctica he decidido implementar por un parte, el servidor en Python y por la otra, el cliente en java. Teniendo en cuenta de que voy a trabajar con el ide Pycharm para Python, e IntelliJ para Java y que estoy usando una distribución de Linux -> "Pop\_os".

Lo primero que hice fue realizar el ejemplo que el profesor de prácticas nos mostró en clase. En dicho ejemplo "calculator.thrift" era un archivo muy simple, el cual, solo contenía una función que sumaba 2 enteros y devuelve el resultado de dicha suma. Una vez generé los archivos correspondientes para python, con el comando: "thrift -gen py calculator.thrift", implementé tanto el servidor como el cliente. Tras esto, aún me quedaba añadir las bibliotecas de thrift para resolver algunos problemas que me aparecían. Busqué la forma de añadir dichas librerías a mi ide y encontré que puedes añadirlas siguiendo estos pasos:

Settings > Python Interpreter > '+' insertar/instalar > buscamos thrift e instalamos.

Con esto, ya estaba listo para ejecutar mi programa. Y en efecto, funcionaba perfectamente. Una vez que me funcionaba dicho ejemplo, implementé un par de funciones más al archivo "calculator.thrift", añadiendo las operaciones básicas que me faltaban (resta, multiplicación, división), y haciendo dichas operaciones básicas con listas de enteros.



```
1 service Calculator {
2     void ping()
3     double add(1:double num1, 2:double num2)
4     double Subtract(1:double num1, 2:double num2)
5     double Multiply(1:double num1, 2:double num2)
6     double Divide(1:double num1, 2:double num2)
7
8     list<i32> listAdd(1:list<i32> v1, 2:list<i32> v2)
9     list<i32> listSubtract(1:list<i32> num1, 2:list<i32> num2)
10    list<i32> listMultiply(1:list<i32> num1, 2:list<i32> num2)
11    list<i32> listDivide(1:list<i32> num1, 2:list<i32> num2)
12 }
```

Tras implementar estos cambios tanto en el .thrift como en el servidor y el cliente, los resultados eran los siguientes

```

Operaciones básicas con double
1.5+1.5 = 3
4.2-4.1 = 0
2.5*2.5 = 6
5/2 = 2
Operaciones básicas con vectores enteros
Primer vector v1: [4, 4, 4]
Segundo vector v2: [2, 2, 2]
v1 + v2: [6, 6, 6]
v1 - v2: [2, 2, 2]
v1 * v2: [8, 8, 8]
v1 / v2: [2, 2, 2]

```

En este punto, ya tenía implementado las operaciones básicas con servidor y cliente en Python. Y ahora lo único que me faltaba era crear el cliente de Java. En primer lugar generé los archivos necesarios para thrift con java, usando el comando: “thrift -gen java calculator.thrift”. A continuación, creé el cliente de java, fijándome en como estaba en la página del tutorial de thrift:

<https://gitbox.apache.org/repos/asf?p=thrift.git;a=tree:f=tutorial;hb=HEAD>

Ahora, tal y como hice con Python, debía añadir las librerías de thrift para java en mi ide. Para añadirlas, descargue los .jar desde maven, y los añadí a IntelliJ de la siguiente manera: File < Project structure < Modules < Dependencies < ‘+’ añadimos las librerías. La librerías que he instalado son:

<https://mvnrepository.com/artifact/org.apache.thrift/libthrift>

<https://mvnrepository.com/artifact/javax.annotation/javax.annotation-api/1.3.2>

<https://mvnrepository.com/artifact/org.slf4j/slf4j-api>

Una vez instaladas todas las librerías, simplemente quedaba probar a ejecutar el servidor de python, y el cliente de Java. Este es el resultado:

```

ping()
Operaciones básicas con double
1.5+1.5 =3.0
4.2-4.1 = 0.100000000000000053
2.5*2.5 = 6.25
5/2 = 2.5

Operaciones básicas con vectores enteros
Primer vector v1: [4,4,4]
Segundo vector v2: [2,2,2]
v1 + v2: [6,6,6]
v1 - v2: [2,2,2]
v1 * v2: [8,8,8]
v1 / v2: [2,2,2]

Process finished with exit code 0

```

```

Starting the server...
ping()
add( 1.5 , 1.5 )
subtract( 4.2 , 4.1 )
multiply( 2.5 , 2.5 )
divide( 5.0 , 2.0 )

```