

1. ¿Qué tipo de ejecución (compilado, interpretado, etc) tiene el lenguaje?

Clojure: Compilado



Kotlin: Compilado



2. ¿Para qué tipo de desarrollo se utiliza normalmente el lenguaje?

Clojure: Clojure es un lenguaje de programación de propósito general. Es particularmente eficiente en el procesamiento de datos y la programación concurrente.

Kotlin: Programación de aplicaciones de teléfono para Android.

3. ¿Con qué ide o editor de texto puede utilizar el lenguaje? Nombre de una librería o framework famoso del mismo.

Clojure:

IDE: Vscode, Vim, Emacs

Librería de Clojure: data.zip, librería para manipular archivos zip. (Otra librerías:

https://clojure.org/dev/contrib_libs)

Kotlin:

IDE: Android Studio

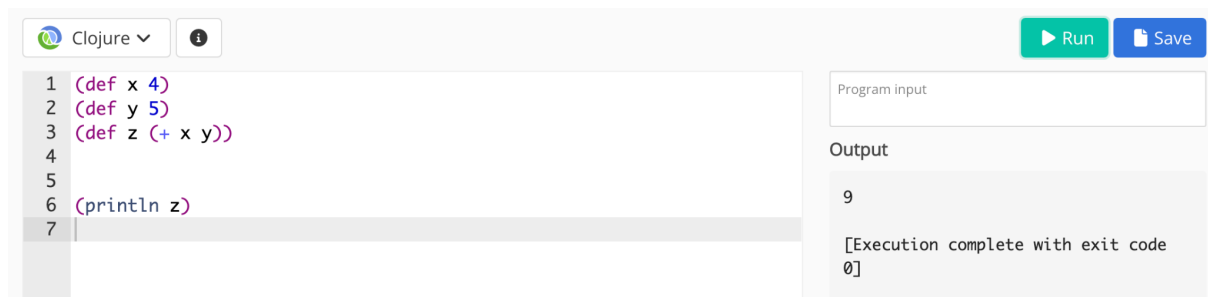
Librería de Kotlin: KAndroid diseñada para explotar una de las características más importantes de Kotlin: las extensiones, que permiten añadir comportamiento a las propiedades y métodos de cualquier clase.

4. Investigar y realizar en la sintaxis del lenguaje dado, la siguiente operación matemática:

Clojure:

```
(def x 4)
(def y 5)
(def z (+ x y))
```

```
(println z)
```



The screenshot shows a Clojure REPL interface. The language is set to 'Clojure'. The code editor contains the following code:

```
1 (def x 4)
2 (def y 5)
3 (def z (+ x y))
4
5
6 (println z)
7
```

On the right side, there are buttons for 'Run' and 'Save'. Below them is a 'Program input' field. The 'Output' section displays the result of the execution:

```
9
[Execution complete with exit code 0]
```

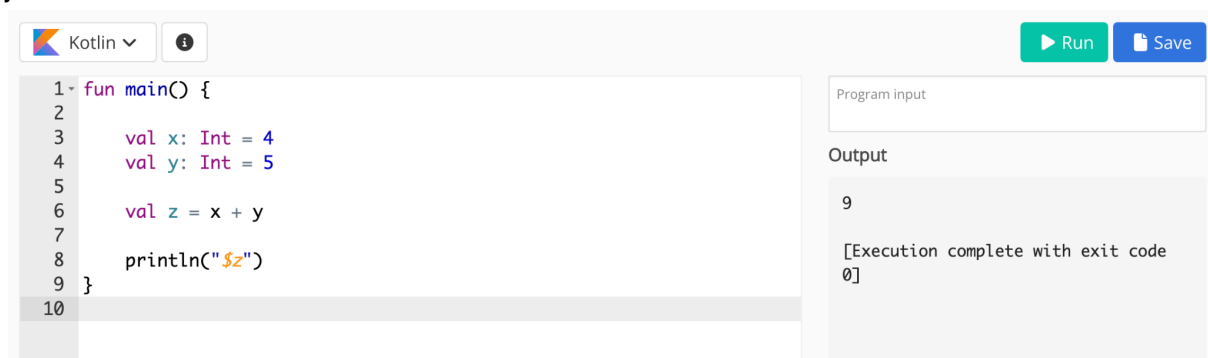
Kotlin:

```
fun main() {

    val x: Int = 4
    val y: Int = 5

    val z = x + y

    println("$z")
}
```



The screenshot shows a Kotlin REPL interface. The language is set to 'Kotlin'. The code editor contains the following code:

```
1 fun main() {
2
3     val x: Int = 4
4     val y: Int = 5
5
6     val z = x + y
7
8     println("$z")
9 }
10
```

On the right side, there are buttons for 'Run' and 'Save'. Below them is a 'Program input' field. The 'Output' section displays the result of the execution:

```
9
[Execution complete with exit code 0]
```