

Respuestas:

1. ¿Qué tipo de ejecución (compilado, interpretado, etc) tiene el lenguaje?
 - a. GO
Go es un lenguaje de programación concurrente y compilado con tipado estático inspirado en la sintaxis de C, pero con seguridad de memoria y recolección de basura. Go es un lenguaje de programación creado por Google en 2009, más conocido como Golang.
 - b. C++
C++ es un ejemplo de lenguaje de programación compilado, multiparadigma, principalmente de tipo imperativo y orientado a objetos, incluyendo también programación genérica y funcional
2. ¿Para qué tipo de desarrollo se utiliza normalmente el lenguaje?
 - a. GO
Golang ha ganado mucha fuerza, empresas de renombre confían en golang, entre ellas Dropbox, Docker, Facebook, Netflix, Uber y Twitter, puedes consultar la lista de compañías de todo el mundo que actualmente usan golang en <https://github.com/golang/go/wiki/GoUsers>. Con golang puedes programar tu aplicación una sola vez y luego compilarla para cualquier sistema operativo, esta característica es conocida como cross compilation. Go es un lenguaje de programación creado por Google en 2009, más conocido como Golang.
 - b. C++
Microsoft Visual C + + es un entorno de desarrollo integrado pensado, como su nombre lo dice, para escribir código con C++ y aporta numerosas facilidades al programador. En este caso se utiliza principalmente para el desarrollo de aplicaciones para el sistema operativo Windows
3. ¿Con que ide o editor de texto puede utilizar el lenguaje? Nombre de una librería o framework famoso del mismo.
 - a. GO
En <https://golang.org/pkg/> puedes encontrar las librerías internas de Go y otras que puedes obtener con go get.
 - b. C++: dev c++
La librería principal de c + + es <iostream>.
4. Investigar y realizar en la sintaxis del lenguaje dado, la siguiente operación matemática:
 - $x = 4$
 - $y = 5$
 - $z = x + y$
 - mostrar por pantalla z
 - a. GO

```
package main

import "fmt"

func main() {
```

```
x := 4
y := 5
z := x+y
fmt.Println(z)
}
```



myCompiler

Enter a title...

GO Go ▾



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     x := 4
7     y := 5
8     z := x+y
9     fmt.Println(z)
10 }
11 |
```

▶ Run

📄 Save

Program Input

Output


9

[Execution complete with exit code 0]

b. ++

```
#include <iostream>
```

```
int main() {  
    int x = 4;  
    int y = 5;  
    int z = x+y;  
    std::cout << z << std::endl;  
}
```

 myCompiler

Enter a title...

 C++ ▾



```
1  #include <iostream>  
2  
3  int main() {  
4      int x = 4;  
5      int y = 5;  
6      int z = x+y;  
7      std::cout << z << std::endl;  
8  
9  }
```

 Run

 Save

Program Input

Output

9