

Curso: Sistema de informação

ARQUITETURA DE COMPUTADORES

Hello world em diversas linguagens

ADA:

```

    with Text_IO; use Text_IO;
procedure hello is
begin
    Put_Line("Hello world!");
end hello;

```

////////////////////////////////////

ASSEMBLY:

; Hello World for Intel Assembler (MSDOS)

```
mov ax,cs
mov ds,ax
mov ah,9
mov dx, offset Hello
int 21h
xor ax,ax
int 21h
```

Hello:

```
db "Hello World!",13,10,"$
```

Bash:

echo Hello World

C:

$$\}$$

C#:

}

C++:

```
#include <iostream>
```

```
int main{
```

```
    cout << "Hello, world!" << endl;
```

```
return 0;
```

```
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

COBOL:

```
* Hello World in COBOL
```

```
*****
```

```
IDENTIFICATION DIVISION.
```

```
PROGRAM-ID. HELLO.
```

```
ENVIRONMENT DIVISION.
```

```
DATA DIVISION.
```

```
PROCEDURE DIVISION.
```

```
MAIN SECTION.
```

```
DISPLAY "Hello World!"
```

```
STOP RUN.
```

```
*****
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

DART:

```
void main() { print('Hello, World!'); }
```

ELIXIR:

```
defmodule HelloWorld do
```

end

ERLANG:

f#:

```
printfn "Hello, world!"
```

FORTRAN:

STOP

```
100 FORMAT (' Hello World! ' /)
```

```
END
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

GO:

```
package main
```

```
import "fmt"
```

```
func main() {
```

```
    fmt.Printf("Hello World\n")
```

```
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

GROOVY:

```
static void main(String[] args) {
```

```
    // Using a simple println statement to print output to the console
```

```
    println('Hello World');
```

```
}
```

```
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

HASKELL:

```
main = do
```

```
putStrLn "Hello World!"
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

HTML:

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Hello, World!</title>
```

```
  </head>
```

<body>

Hello World!

</body>

////////////////////////////////////

JAVA:

```
class HelloWorld {
```

```
static public void main( String args[] ) {
```

```
System.out.println( "Hello World!" );
```

////////////////////////////////////

JavaScript:

```
console.log("Hello World");
```

////////////////////////////////////

JULIA:

```
println("Hello, World!")
```

////////////////////////////////////

KOTLIN:

```
fun main() {
```

```
println("Hello, world!")
```

$$\}$$

////////////////////////////////////

LUA:

```
print("Hello World")
```

////////////////////////////////////

MATLAB:

```
disp('Hello, world!')
```

////////////////////////////////////

OBJECTIVE-C:

RUBY:

puts "Hello, World!"

////////////////////////////////////

RUST:

```
fn main() { println!("Hello, World!"); }
```

////////////////////////////////////

SCALA:

```
object HelloWorld { def main(args: Array[String]): Unit = { println("Hello, World!") } }
```

SCHEME:

```
(display "Hello, World!") (newline)
```

////////////////////////////////////

SHELL SCRIPT:

```
#!/bin/sh
```

```
echo "Hello, World!"
```

SQL:

```
SELECT 'Hello, World!' AS greeting;
```

SWIFT:

```
import Swift print("Hello, World!")
```

////////////////////////////////////

TYPESCRIPT:

```
console.log("Hello, World!");
```

VBScript:

```
WScript.Echo "Hello, World!"
```


VB:

```
Module Module1 Sub Main() Console.WriteLine("Hello, World!") End Sub End  
Module
```