

Rules Of The Road

- **modules** must start with a **lowercase** letter. The module name must match its file name without the `.erl` extension.

- **export** statements identify which functions defined in module are public. The statement:

```
export ([ factorial / 1 ]).
```

indicates that the function called **factorial** is public and accepts one argument.

- **Variables are immutable.** Once a value has been assigned to variable:

```
X = 1.
```

any attempt to assign a new value to *X* will fail.

- **Variable names must start with an uppercase letter.**

- **Atoms start with lowercase letters.**

- To compile from the command line use the **erlc** command:

```
erlc file.erl
```

- To execute a program from the command line use the command:

```
erl -noshell -s <module> <method> -s init stop
```

- Like English sentences, erlang statements end with period(`.`).

Hello World

Here is the erlang version of the inevitable HelloWorld program:

Listing 1: "helloWorld.erl"

```
-module(helloWorld).  
-export([start/0]).  
  
start() ->  
    io:format("Hello ,_world!~n").
```

Compiling and running programs from erl

To compile and run the program, start the command shell by entering the **erl** command, then:

Listing 2: "running our helloWorld program"

```
$ erl
Erlang/OTP 20 [erts -9.0] [source] [64-bit] [smp:8:8] [ds
:8:8:10] [async-threads:10] [hipe] [kernel-poll:false]
[dtrace]

Eshell V9.0 (abort with ^G)
1> c(helloWorld).
{ok,helloWorld}
2> helloWorld:start().
Hello, world!
ok
3>
```