

Uses

```
format!("{}", " World! "); // Returns a
formatted string from a fmtstr.
let mut vec = vec![""];
// Destination (first arg) must &impl mut
io::Write or a &impl mut fmt::Write
write! (&mut vec, " Hello {}", " World! "); //
Writes a fmtstr
writeln! (&mut vec, " Hello Me!"); // writeln! but
appends a newline
print! ("Hello {}", " World! "); // Prints a
fmtstr to stdout
println! ( " Hello {}", " World! "); // println! but
appends a newline
eprint!("{}", " World! "); // Prints a
fmtstr to stderr
eprintln! ("Hello {}", " World! "); // eprintln!
but appends a newline
format_! ("Hello {}", " World! "); //
Returns fmt::A rgu ments, see https://d oc.r us -
t-l ang.or g/s td/ fmt /#f orm at_ args
```

Referencing Arguments

{}	Next positional argument ¹
{n}	n-th positional argument, starting from 0
{ident}	"ident" named argument, if nonexistent "ident" variable from the scope
{{	Literal {
}}	Literal }

¹Ex. The second usage of {} would be the second positional argument.

Specifying Arguments

All format macros start with a format string, with the exception of `writeln!` and `writeln!` requiring a additional destination argument. It can contain a number of formats specified below. Positional arguments are specified immediately after, and then named arguments in the `ident = val format`.



By **ambi (ambiguous)**
cheatography.com/ambiguous/

Not published yet.
 Last updated 19th November, 2023.
 Page 1 of 1.

Sponsored by **CrosswordCheats.com**
 Learn to solve cryptic crosswords!
<http://crosswordcheats.com>