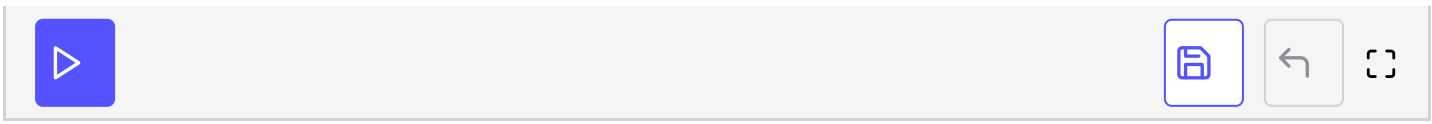
Declarations

The syntax for C requires you to write the type of the variable you want to declare before the variable's name.

Unlike in languages like Python, R, Octave/Matlab, etc., which are **dynamically typed** languages, the C language is a **statically typed** language. From a practical point of view, this means in C we have to declare, up front, the **type** of every variable we use. In languages like Python we can do crazy stuff like this:

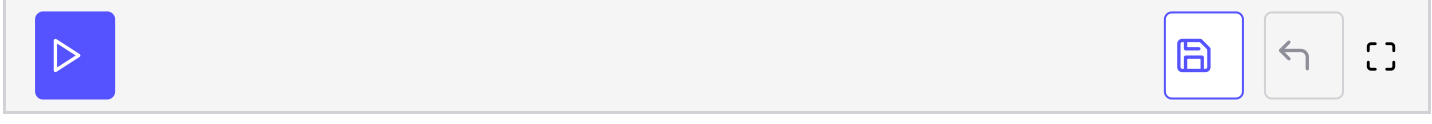
```
a = 123.456
b = 50.2
c = 100.0
d = [a, b, c]
print a, b, c, d
```



The Python interpreter will figure out what type to assign to `a`, `b`, `c` and `d` based on evaluating the right-hand side of each declaration. In C, we have to explicitly declare the type of each variable like this:

```
#include <stdio.h>

int main() {
    double a = 123.456;
    double b = 50.2;
    double c = 100.0;
    double d[] = {a, b, c};
    printf("a=%.3f, b=%.3f, c=%.3f, d=[%.3f, %.3f, %.3f]\n",
                a, b, c, d[0], d[1], d[2]);
    return 0;
}
```



We haven't talked about arrays yet but we will later in the tutorial.

For now, let's move on to making meaningful statements with our declared variables.