

## - Example

We'll take a look at an example of raw and cooked forms of user-defined literals.

### WE'LL COVER THE FOLLOWING ^

- Example
- Explanation

## Example #

main.cpp

distance.h

unit.h



```
// average.cpp

#include "distance.h"
#include "unit.h"

using namespace Distance::Unit;

int main(){

    std::cout << std::endl;

    std::cout << "1.0_km: " << 1.0_km << std::endl;
    std::cout << "1.0_m: " << 1.0_m << std::endl;
    std::cout << "1.0_dm: " << 1.0_dm << std::endl;
    std::cout << "1.0_cm: " << 1.0_cm << std::endl;

    std::cout << std::endl;

    std::cout << "0.001 * 1.0_km: " << 0.001 * 1.0_km << std::endl;
    std::cout << "10 * 1_dm: " << 10 * 1.0_dm << std::endl;
    std::cout << "100 * 1.0cm: " << 100 * 1.0_cm << std::endl;
    std::cout << "1_.0km / 1000: " << 1.0_km / 1000 << std::endl;

    std::cout << std::endl;
    std::cout << "1.0_km + 2.0_dm + 3.0_dm + 4.0_cm: " << 1.0_km + 2.0_dm + 3.0_dm + 4.0_cm
    std::cout << std::endl;
```

```

auto work= 63.0_km;
auto workPerDay= 2 * work;
auto abbreviationToWork= 5400.0_m;
auto workout= 2 * 1600.0_m;
auto shopping= 2 * 1200.0_m;

auto distPerWeek1= 4*workPerDay-3*abbreviationToWork+ workout+ shopping;
auto distPerWeek2= 4*workPerDay-3*abbreviationToWork+ 2*workout;
auto distPerWeek3= 4*workout + 2*shopping;
auto distPerWeek4= 5*workout + shopping;

std::cout << "distPerWeek1: " << distPerWeek1 << std::endl;

auto averageDistance= getAverageDistance({distPerWeek1,distPerWeek2,distPerWeek3,distPerWeek4});
std::cout<< "averageDistance: " << averageDistance << std::endl;

std::cout << std::endl;
}

```



## Explanation #

- We did our computations based on user-defined literals of the type `long double` in the cooked form. To make our calculation in the raw form, we must only adjust the literal operators.
- It is only necessary to convert the arguments of the literal operator from type C string to `long double`. That is quite easy to do with the new function `std::stold`.

---

Now that we have gone over user-defined literals, we will study built-in literals in the next lesson.