- Example

In this lesson, we'll look at an example of the automatic return type.

WE'LL COVER THE FOLLOWING

Example: Automatic Template Return Type

```
// templateAutomaticReturnType.cpp

#include <iostream>
#include <typeinfo>

template<typename T1, typename T2>
auto add(T1 first, T2 second) -> decltype(first + second){
  return first + second;
}

int main(){

  std::cout << std::endl;

  std::cout << "add(1, 1)= " << add(1,1) << std::endl;
  std::cout << "typeid(add(1, 1)).name()= " << typeid(add(1, 1)).name() << std::endl;

  std::cout << std::endl;

  std::cout << "add(1, 2.1)= " << add(1,2.1) << std::endl;
  std::cout << "typeid(add(1, 2.1)).name()= " << typeid(add(1, 2.1)).name() << std::endl;

  std::cout << std::endl;

  std::cout << "add(1000LL, 5)= " << add(1000LL,5) << std::endl;
  std::cout << "typeid(add(1000LL, 5)).name()= " << typeid(add(1000LL, 5)).name() << std::enc

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

Explanation

The example has a function `add` which takes two arguments and returns their sum. The return type of the function is deduced by the compiler by applying the `decltype` operator on the sum of the arguments. The expression `typeid(add(1, 2.1)).name()` such as in line 21 returns a string representation of the type of result.

We'll solve an exercise in the next lesson.