Pointer to Char

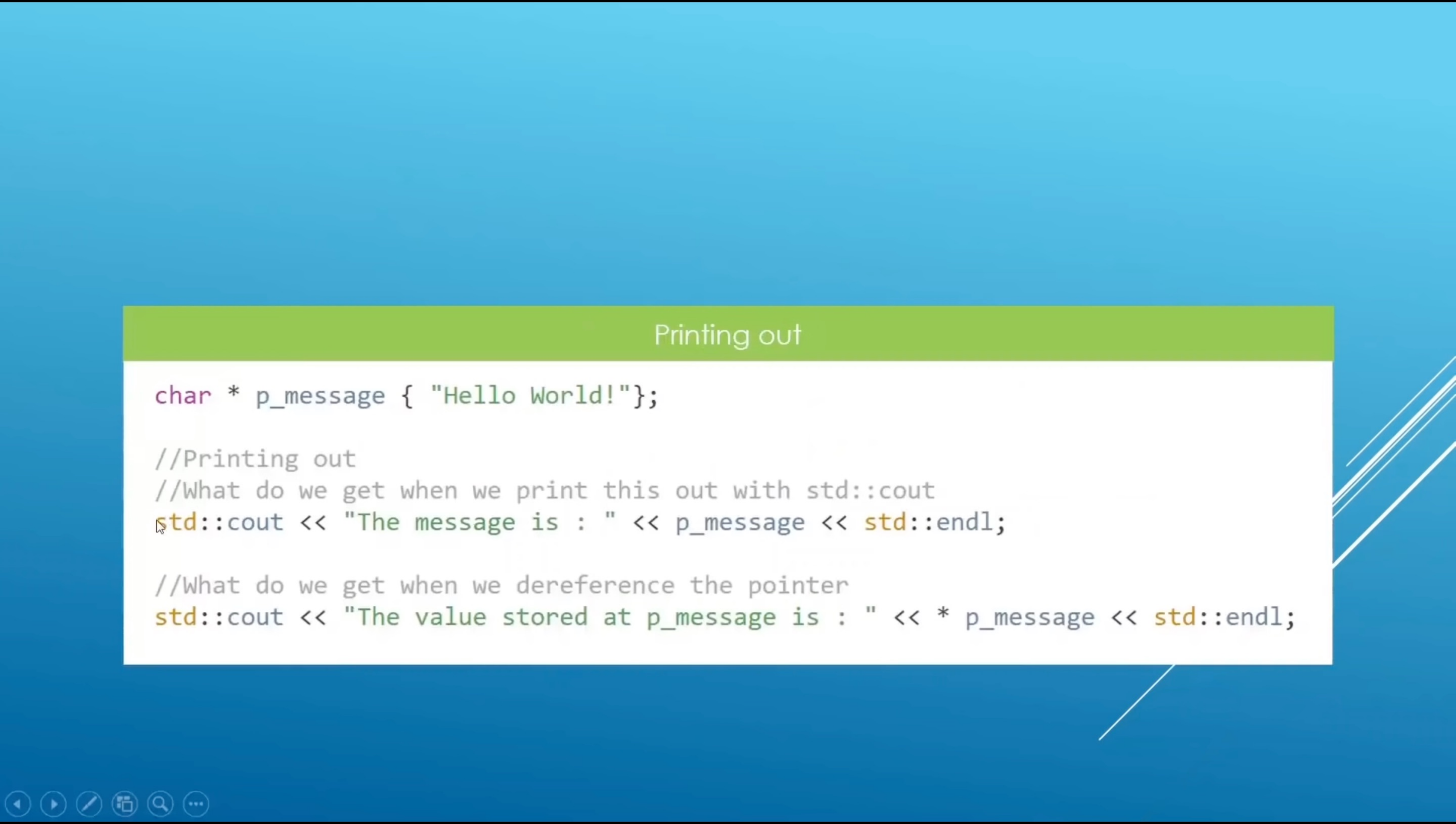
Declaring pointers to char

```
//Can use normal pointer to char like we've been doing for
//double, int and anything else really.
char *p_char_var {nullptr};
char char_var {'A'};

p_char_var = &char_var;

std::cout << "The value stored in p_char_var is : " << * p_char_var << std::endl;

char char_var1 {'C'};
p_char_var = &char_var1;

std::cout << "The value stored in p_char_var is : " << * p_char_var << std::endl;
```

Pointer to Char

Initialize with string literal

```
//Pointer to char can also do something special.
//You can initialize with a string literal : C-String

char * p_message { "Hello World!"};
```



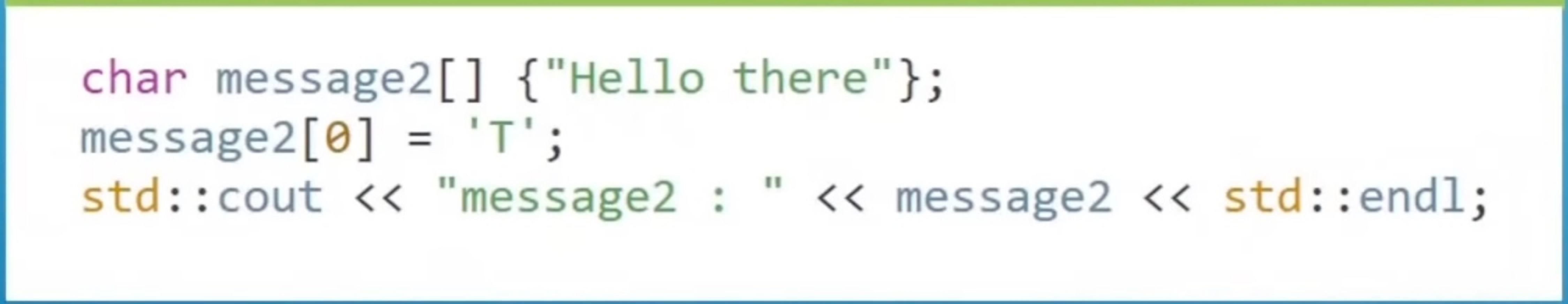
Printing out

```cpp
char * p_message { "Hello World!"};

//Printing out
//What do we get when we print this out with std::cout
std::cout << "The message is : " << p_message << std::endl;

//What do we get when we dereference the pointer
std::cout << "The value stored at p_message is : " << * p_message << std::endl;
```

The string literal is made up of const char. Trying to modify any of them will result in a disaster!

```
*p_message = 'B'; // We want the message to say Bello World!
std::cout << "The message is (after modification of first char to B) : " << p_message << std::endl;
```

Can modify the pure array initialized with string literal



```
char message2[] {"Hello there"};
message2[0] = 'T';
std::cout << "message2 : " << message2 << std::endl;
```