

C++ Mapping for Constants

Slice [constant](#) definitions map to corresponding C++ constant definitions. For example:

Slice

```
const bool      AppendByDefault = true;
const byte      LowerNibble = 0x0f;
const string    Advice = "Don't Panic!";
const short     TheAnswer = 42;
const double    PI = 3.1416;

enum Fruit { Apple, Pear, Orange };
const Fruit     FavoriteFruit = Pear;
```

Here are the generated definitions for these constants:

C++

```
const bool      AppendByDefault = true;
const Ice::Byte LowerNibble = 15;
const std::string Advice = "Don't Panic!";
const Ice::Short TheAnswer = 42;
const Ice::Double PI = 3.1416;

enum Fruit { Apple, Pear, Orange };
const Fruit     FavoriteFruit = Pear;
```

All constants are initialized directly in the header file, so they are compile-time constants and can be used in contexts where a compile-time constant expression is required, such as to dimension an array or as the case label of a switch statement.

See Also

- [Constants and Literals](#)
- [C++ Mapping for Identifiers](#)
- [C++ Mapping for Modules](#)
- [C++ Mapping for Built-In Types](#)
- [C++ Mapping for Enumerations](#)
- [C++ Mapping for Structures](#)
- [C++ Mapping for Sequences](#)
- [C++ Mapping for Dictionaries](#)
- [C++ Mapping for Exceptions](#)