The table that follows shows the most common types used in C++

TYPE	DESCRIPTION	USAGE

int
signed
	Positive and negative integers; range depends on compiler	int i = -7;
		signed j = -5;

(int)	Short integer (usually 2 bytes)	short s = 13;
long (int)	Long integer (usually 4 byets)	long l = -7L;
long long (int)	Long long integer; range depends on compiler, but at least the same as long (usually 8 bytes)	long long ll = 14LL

unsigned (int)
unsigned short (int)
unsigned long (int)
unsigned long long (int)
	Limits the preceding types to values >=0	unsigned int i = 2U;
		unsigned j = 5U;
		unsigned short s = 23U;
		unsigned long l = 5400L;
		unsigned long long ll = 140ULL;

float	Floating-point numbers	float f = 7.2f;
double	Double precision numbers; precision is at least the same as for float	double d = 7.2;
long double	Long double precision numbers; precision at least the same as for double	long double d = 16.98L;
char	A single character	char ch = 'm';
char16_t	a single 16-bit character	char16_t c16 = u'm';
char32_t	A single 32-bit character	char32_t c32 = U'm';
wchar_t	A single wide-character; size depends on compiler	wchar_t = L'm';
bool	true or false	bool b = true;

The best way to to cast a type to another type, as an example a float to an int is shown

```
float myFloat = 3.14f;
int i = static_cast<int>(myFloat)
```

In some context, variables can be automatically cast, or *coerced.* For example, a short can be automatically cast into a long because a long represents the same type of data with at least the same precision.

```
long someLong = someShort // no explicit cast needed
```

When automatically casting variables, you need to be aware of the potential loss of data.