

Example of Constructor Declaration in a Class Definition

Let's do an example. Here's a class:

```
class MyClass {
private:
    int32_t one;
    int32_t two;
public:
    MyClass (int32_t arg,
            int32_t second);
};
```

Annotations:

- for clarity (points to `class MyClass {`)
- a couple of fields (points to `int32_t one;` and `int32_t two;`)
- a public constructor taking two `int32_t` arguments (points to `MyClass (int32_t arg, int32_t second);`)

9

Arrays Require a Constructor with No Arguments

A question: **given the class just shown, is the following code acceptable?**

```
MyClass m[42];
```

No!

- **No constructor without arguments** is declared,
- nor is the default produced,
- since a constructor is declared.

10

Example of Constructor Declaration in a Class Definition

How does the code look?

In a source file somewhere...

```
MyClass::MyClass
(int32_t arg, int32_t second)
: one (arg), two (second) {
    // code
};
```

Annotations:

- constructor name (points to `MyClass::MyClass`)
- parameter list (points to `(int32_t arg, int32_t second)`)
- optional initializer list (preceded by a colon) (points to `: one (arg), two (second)`)

11

Initializers Describe Base Class and Fields Initialization

What's an initializer?

A **brief specification for initialization**

- consisting of **another constructor, a base class, or a field name** and
- an arbitrary **expression** in parentheses
- (function calls are allowed).

For base classes and instances, lists of expressions are passed to constructors.

12