

## C++ Code

```
1 #include "MediaManager.h"
2
3 // Create Photo
4 std::shared_ptr<Photo> MediaManager::createPhoto(const std::string &name, const
5 std::string &filename, double lat, double lon)
6 {
7     std::shared_ptr<Photo> p(new Photo(name, filename, lat, lon));
8     objects[name] = std::static_pointer_cast<MultimediaObject>(p);
9     return p;
10 }
11
12 // Create Video
13 std::shared_ptr<Video> MediaManager::createVideo(const std::string &name, const
14 std::string &filename, int duree)
15 {
16     std::shared_ptr<Video> v(new Video(name, filename, duree));
17     objects[name] = std::static_pointer_cast<MultimediaObject>(v);
18     return v;
19 }
20
21 // Create Film
22 std::shared_ptr<Film> MediaManager::createFilm(const std::string &name, const
23 std::string &filename, int duree)
24 {
25     std::shared_ptr<Film> f(new Film(name, filename, duree));
26     objects[name] = std::static_pointer_cast<MultimediaObject>(f);
27     return f;
28 }
29
30 // Create Groupe
31 GroupePtr MediaManager::createGroupe(const std::string &name)
32 {
33     GroupePtr g(new Groupe(name));
34     groups[name] = g;
35     return g;
36 }
37
38 // Display object
39 void MediaManager::displayObject(const std::string &name, std::ostream &out) const
40 {
41     auto it = objects.find(name);
42     if (it == objects.end())
43     {
44         std::cout << "Objet '" << name << "' introuvable." << std::endl;
45         return;
46     }
47     it->second->affiche(out);
48     out << std::endl;
49 }
```

```
48 // Display groupe
49 void MediaManager::displayGroupe(const std::string &name, std::ostream &out) const
50 {
51     auto it = groups.find(name);
52     if (it == groups.end())
53     {
54         std::cout << "Groupe '" << name << "' introuvable." << std::endl;
55         return;
56     }
57     it->second->affiche(std::cout);
58     std::cout << std::endl;
59 }
60
61 // Play object
62 void MediaManager::playObject(const std::string &name, std::ostream &out) const
63 {
64     auto it = objects.find(name);
65     if (it == objects.end())
66     {
67         out << "Objet '" << name << "' introuvable." << std::endl;
68         return;
69     }
70     it->second->jouer(out);
71 }
72
73 // Remove object
74 bool MediaManager::removeObject(const std::string &name)
75 {
76     auto it = objects.find(name);
77     if (it == objects.end())
78         return false;
79     objects.erase(it);
80     return true;
81 }
82
83 // Remove groupe
84 bool MediaManager::removeGroupe(const std::string &name)
85 {
86     auto it = groups.find(name);
87     if (it == groups.end())
88         return false;
89     groups.erase(it);
90     return true;
91 }
92
```