



BUILDING ASSOCIATION

Constructors and setters



BUILDING ASSOCIATION: SETTER FUNCTIONS

project.h

```
class contractor;  
  
class project  
{  
    private:  
        contractor* theContractor;  
    public:  
        void set_contractor(contractor* a_c)  
        {  
            theContractor = a_c;  
        }  
};
```

contractor.h

```
class project;  
  
class contractor;  
{  
    private:  
        project* theProject;  
    public:  
        void set_project(project* a_p)  
        {  
            theProject = a_p;  
        }  
};
```

CALLING SETTER FUNCTIONS

```
#include "contractor.h"
#include "project.h"

int main()
{
    project    big;
    contractor fred;
    . . . .
    set_contractor(&fred);
    set_project(&big);

    return 0;
}
```

```
#include "contractor.h"
#include "project.h"

int main()
{
    project*    big = new project;
    contractor* fred = new contractor;
    . . . .
    set_contractor(fred);
    set_project(big);

    return 0;
}
```



BUILDING ASSOCIATION WITH CONSTRUCTORS

```
#pragma once
#include <iostream>
using namespace std;

class contractor;
#include "contractor.h"

class project
{
private:
    contractor* theContractor;
public:
    project();
    project(contractor* a_c);
};
```

```
#pragma once
#include <iostream>
using namespace std;

class project;
#include "project.h"

class contractor
{
private:
    project* theProject;
public:
    contractor(project* a_p);
    contractor();
};
```



ASSOCIATION CLASSES: AFTER PREPROCESSING #INCLUDE

```
class contractor
{
    private:
        project* theProject;
};

class project
{
    private:
        contractor* theContractor;
};
```

```
class contractor;
class project;

class contractor
{
    private:
        project* theProject;
};

class project
{
    private:
        contractor* theContractor;
};
```



COMPLEMENTARY ASSOCIATION CONSTRUCTORS

project.cpp

```
#include "project.h"

project::project()
{
    theContractor = new contractor(this);
}

project::project(contractor* a_c)
{
    theContractor = a_c;
}
```

contractor.cpp

```
#include "contractor.h"

contractor::contractor(project* a_p)
{
    theProject = a_p;
}

contractor::contractor()
{
    theProject = new project(this);
}
```



BUILDING ASSOCIATION: project FIRST

project.cpp

```
#include "project.h"

project::project()
{
    theContractor = new contractor(this);
}

project::project(contractor* a_c)
{
    theContractor = a_c;
}
```

APPLICATION

```
#include "project.h"

int main()
{
    project        little;
    project*       big = new project;

    return 0;
}
```



BUILDING ASSOCIATION

contractor FIRST

APPLICATION

```
#include "contractor.h"
#include "project.h"

int main()
{
    contractor    foo;
    contractor*   bar = new contractor;

    return 0;
}
```

contractor.cpp

```
#include "contractor.h"

contractor::contractor(project* a_p)
{
    theProject = a_p;
}

contractor::contractor()
{
    theProject = new project(this);
}
```