```cpp
#include <iostream>
using namespace std;
class Employee
{
  public:
    Employee()
    {
      cout<<"Constructor Invoked"<<endl;
    }
    ~Employee()
    {
      cout<<"Destructor Invoked"<<endl;
    }
};
int main(void)
{
  Employee e1; //creating an object of Employee
  Employee e2; //creating an object of Employee
  return 0;
}
```

```cpp
#include<iostream>
using namespace std;
class Demo {
  private:
  int num1, num2;
  public:
  Demo(int n1, int n2) {
    cout<<"Inside Constructor"<<endl;
    num1 = n1;
```

```cpp
    num2 = n2;
  }
  void display() {
    cout<<"num1 = "<< num1 <<endl;
    cout<<"num2 = "<< num2 <<endl;
  }
  ~Demo() {
    cout<<"Inside Destructor";
  }

};
int main() {
  Demo obj1(10, 20);
  obj1.display();
  return 0;
}




#include <iostream>

using namespace std;
class Line {
  public:
    void setLength( double len );
    double getLength( void );
    Line();   // This is the constructor declaration
    ~Line();  // This is the destructor: declaration

  private:
    double length;
};
```

```cpp
// Member functions definitions including constructor
Line::Line(void) {
   cout << "Object is being created" << endl;
}
Line::~Line(void) {
   cout << "Object is being deleted" << endl;
}
void Line::setLength( double len ) {
   length = len;
}
double Line::getLength( void ) {
   return length;
}

// Main function for the program
int main() {
   Line line;

   // set line length
   line.setLength(6.0);
   cout << "Length of line : " << line.getLength() <<endl;

   return 0;
}
```