**Constructors**

A special member function whose task is to initialize the data members of a class.

**Properties of constructors**

* Have same name as the class name
* Should be declared in the public section of a class
* Invoked automatically whenever objects are created.
* Have no return types
* Cannot be inherited.
* Can have default arguments.

//**example of constructor**

#include<iostream>

#include<conio.h>

using namespace std;

class integer

{

private:

int a,b;

public:

integer();

void display();

};

integer :: integer()

{

a=0;

b=2;

}

void integer :: display()

{

cout<<"values of a and b\n";

cout<<"\na = "<<a<<"\t"<<"b = "<<b<<endl;

}

int main()

{

integer int1;

int1.display();

return 0;

}

**Parameterized constructors**

Constructors that do have arguments in the function list.

//parameterized constructors example

#include<iostream>

#include<conio.h>

using namespace std;

class integer

{

private:

int x,y;

public:

integer(int,int);

};

integer :: integer(int m,int n)

{

x=m;

y=n;

//x=x+5;

cout<<"\nx = "<<x<<endl;

cout<<"\ny = "<<y<<endl;

}

int main()

{

integer int1(9,7); //constructor called implicitly

integer int2=integer(1,3); //constructor called explicitly

return 0;

}

//Multiple constructors example

#include<iostream>

#include<conio.h>

using namespace std;

class integer

{

private:

int x,y;

public:

integer(int,int); //constructor1

integer(); //constructor2

void display();

};

integer :: integer()

{

a=0;

b=2;

}

void integer :: display()

{

cout<<"values of a and b\n";

cout<<"\na = "<<a<<"\t"<<"b = "<<b<<endl;

}

integer :: integer(int m,int n=345)

{

x=m;

y=n;

//x=x+5;

cout<<"\nx = "<<x<<endl;

cout<<"\ny = "<<y<<endl;

}

int main()

{

integer int3;

integer int1(9,7); //constructor called implicitly

integer int2=integer(1); //constructor called explicitly

int3.display();

return 0;

}

*//****default constructor***

*#include<iostream>*

*#include<conio.h>*

*class integer*

*{*

*private:*

*int a,b;*

*public:*

*integer(int,int);*

*void display();*

*};*

*integer :: integer(int x,int y=9)*

*{*

*a=x,b=y;*

*}*

*void integer :: display()*

*{*

*std::cout<<"\na= "<<a<<"\n";*

*std::cout<<"\nb = "<<b<<"\n";*

*}*

*int main()*

*{*

*integer int1(3);*

*int1.display();*

*return 0;*

*}*