```
1. class GFG
   {
   public:
         GFG()
         {
            cout << "Hi from GFG. ";
         }
   } g;

   main()
   {
       cout << "You are in Main";
   }
```

```
2. class construct
   {
       int a, b;
   public:
       construct()
       {
           a = 0;
           b = 0;
       }
   };

   main()
   {
       construct c;
       cout<< "a: "<< c.a << endl << "b: "<< c.b;
   }
```

```
3. class constructor
   {
       int x, y;
       public:
       constructor(int a = 10, int b = 20 )
       {
           x = a;
           y = b;
       }
       void Display()
       {
           cout<< x << " " << y << endl;
       }
   };
   main()
   {
       constructor objBix;
       objBix.Display();
   }
```

```
4. class constructor
   {
       int x;
       public:
       constructor(short ss)
       {
           cout<< "Short" << endl;
       }


       constructor(int xx)
       {
           cout<< "Int" << endl;
       }
       constructor(float ff)
       {
           cout<< "Float" << endl;
       }
   };
   main()
   {
       constructor c('B');
   }

5. class Example {
       public:
           int a;
           int b;
   };
    main()
   {
      Example Ex1  = {10, 20};
      cout << "a = " << Ex1.a <<", b = " << Ex1.b;
   }

6. class Example {
       public:
           void ~Example()
           {
               cout<<"Destroying the object";
           }
   };
   main()
   {
       Example Ex;
   }

7. class Example {
       private:
           int a;
           int b;
       public:
           Example(int a, int b)
           {
               this->a = a;
               this->b = b;
           }
           int get_a()
           {
               return a;
           }
           int get_b()
           {
               return b;
           }
   };
   main()
   {
       Example Ex(10,20);
       cout<<"a = "<<Ex.get_a()<<", b = "<<Ex.get_b();
   }

8. class A{
         int a;
         public:
               A(int i){
                     a = i;
               }
               void assign(int i){
                     a = i;
               }
               int return_value(){
                     return a;
               }
         };
         int main(int argc, char const *argv[])
         {
               A obj;
               obj.assign(5);
               cout<<obj.return_value();
         }

9. class Test
         {
             int x;
             public:
             Test(int xx, float yy)
             {
                 cout<< char(yy);
             }
         };
         main()
         {
             Test *p = new Test(35, 99.50f);
         }
10.      class Bit
         {
               int x;
             public:
               Bit();
              ~Bit();
               void Show();
         };
         Bit::Bit()
         {
             x = 25;
         }
         void Bit::Show()
         {
             cout<< x;
         }
         main()
         {
             Bit objB;
             objB.Show();
         }
```

11.

```
int val = 0;
class Test
{
    public:
    Test()
    {
        cout<< ++val;
    }
    ~Test()
    {
        cout<< val--;
    }
};
main()
{
    Test objBix1, objBix2, objBix3;
    {
        Test objBix4;
    }
}
```

12.

```
class Test
{
    int *p;
    public:
    Test(int xx, char ch)
    {
        p  = new int();
        *p = xx + int(ch);
        cout<< *p;
    }
    ~Test()
    {
        delete p;
    }
};
main()
{
    Test obj(10, 'B');
}
```

13.

```
class Test
{
    int x, y;
    public:
    Test(int xx = 10, int yy = 20 )
    {
        x = xx;
        y = yy;
    }
    void Display()
    {
        cout<< x << " " << y << endl;
    }
    ~Test()
    {
cout<<”Good”;
}
};
```

```
         main()
         {
             Test objBix;
             objBix.Display();
          }
14.      class Test
         {
             int x;
             public:
             Test(short ss)
             {
                 cout<< "Short" << endl;
             }
             Test(int xx)
             {
                 cout<< "Int" << endl;
             }
             Test(float ff)
             {
                 cout<< "Float" << endl;
             }
             ~Test()
             {
                 cout<< "Final";
             }
         };
         main()
         {
             Test *ptr = new Test('B');
         }

15.      class Test
         {
             int x, y;
             public:
                 Test()
                 {
                     x = 0;
                     y = 0;
                 }
                 Test(int xx, int yy)
                 {
                     x = xx;
                     y = yy;
                 }
                 Test(Test *objB)
                 {
                     x = objB->x;
                     y = objB->y;
                 }
                 void Display()
                 {
                     cout<< x << " " << y;
                 }
         };
         main()
         {
             Test objBix( new Test(20, 40) );
             objBix.Display();
            }
```