Array of Objects

Q How to create an array of obj?

A Just like you create an array of int.

~~stu~~ int arr[10];

↓

replace w/
class name

↓

student arr_obj [10];

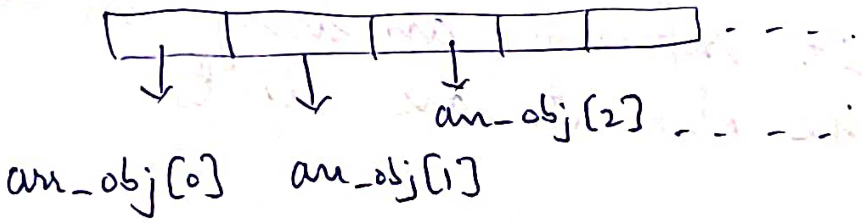
Q What does array of obj represent?

A It represents an array in which all the elements are objects.

Q WAP to show use of an arr of obj?

A
```
class student
{
        private:
        int rn, m1, m2;
        public:
        void set (int x, int y, int z);
        void get ();
        void add ();
};
void student :: set (int x, int y, int z)
{
        rn = x;
        m1 = y;
        m2 = z;
}
```

```
void student::get()
{
    cout<<rn<<m1<<m2;
}
void student::add()
{
    cout<<m1+m2;
}
int main()
{
    student arr_obj[10]; //arr of 10 obj created
```



```
    //These 10 obj can be accessed like
     above or,
    // for loop can be used to call
    // set, get, add fun for each obj


    ~~//1- for loop to assign values~~
    ~~for(int i=0; i<9; i++)~~
    ~~{~~
        ~~arr_obj[i]~~
```

```
// Values will be input by the
   user using cin

  int a, b, c;
  //for loop-1 : to assign values
  for( int i=0; i<10; i++)
  {
          cin>>a>>b>>c;
          arr_obj [i].set (a,b, c);
        //We are not writing
        //arr_obj[i].set(1,10,20) becoz all
        //obj will be assigned same values.
        //So, instead we take input from
        //user.
  }

// for loop-2 : to print values.
for(int i=0; i< 10; i++)
{
       arr_obj [i]. get ();

 }

//for loop-3 to add values.
for (int i=0; i<10; i++)
 {      arr_obj[i].add();


 }

}//main ends.
```

Array of Obj w/ Parameterised constructor

Q What care ~~sta~~ should be taken when creating arr of obj w/ Param^{zd} const?

A • When we create a single obj w/o using const., it is done like this

```
student obj1;
```

• But when we create obj with const, then above stmt. will result in error. We must pass values like this:

```
student obj1(1, 10, 20);
```

• Similarly, ~~if~~ we create arr with/out const like this:

```
student arr_obj[5];
```

• But we must create arr w/ const. like this:-

```
student arr_obj[5] = { student(1,10,20),
                       student(2,5,5),
                       student(3,15,15),
                       student(4,100,50),
                       student(5,6,7)
                     };
```

Q WAP to show use of arr of obj w/ parameterized const.?

A ~~student~~

```
class student
{
        private:
        int rn, m1, m2;
        public:
        void set (int x, int y, int z);
        void get ( );
        void add ();
        ~~to~~ student( int x, int y, int z);
        ~student ();
}
void student :: set (int x, int y, int z)
{
        rn=x;
        m1=y;
        m2=z;
}
void student :: get ( )
{
        cout<<rn<<m1 <<m2; }
void student :: add ( )
{
        cout<< m1+m2;    }
~~to~~ student :: student (int x, int y, int z)
{
        rn=x,
        m1=y;
        m2=z;  cout<<" Const Called"<<;
}
student :: ~student ( )
{
        cout<< "dest called" ;  }
```

```
int main ( )
{
        // inside main - create arr of obj
        Student arr_obj[5] = { student (1, 10, 20),
                               Student (2, 5, 5),
                               student (3, 6, 10)
                               student (4, 8, 9);
                               student (5, 100, 50);
                        };
    // values r assigned using const.
    // Now print & add

    for (int i=0; i<10; i++)
    {
            arr_obj [i]. get ( );
    }
    for (int i=0; i<10; i++)
    )
            arr_ obj (i] . add ( );

    }
}.
```

Output:

```
Const Called
Const Called
Const Called
Const Called
Const Called
1, 10, 20
2, 5, 5
3, 6, 10
4, 8, 9
5, 100, 50
30
10
16
17
150
Dest called
Dest Called
Dest Called
Dest Called
Dest Called
```

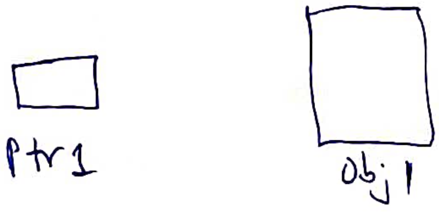
Pointer to objects

Q What is the use of obj?
A obj r used to call the set, get, add fun.

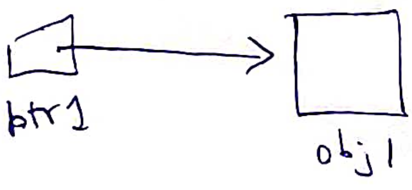
Q Is there any alternative to obj for calling these fun?
A Ptr can be used instead of obj to call these fun.

Q Steps to use a ptr to obj.
A
1) Create obj
2) Create ptr.
3) point the ptr to obj
4) Use pointer to call the fun.

1) Create obj



2) Create a ptr.



3) Point the ptr to obj

4) Call the set, get, add fun using ptr.

Q WAP to show use of ptr to obj

A
```
~~Stru~~ class student
{
        private:
        int ~~rn~~ int m1, int m2;
        public:
        void set (int x, int y, int z);
        void get ();
        void add ();
};
void student :: set (int x, int y, int z)
{
    ~~cou~~ rn = x;
        m1 = y;
        m2 = z;
}
void student :: get ( )
{
    cout << ~~rn~~ rn << m1 << m2; }
void student :: ~~get~~ add ( )
{
    cout << m1 + m2;  }
int main ( )
{
    Student obj1, obj2; //create obj;
    Student *ptr1, *ptr2;
   //create ptr. There should be a
  //separate ptr for each obj.
    ptr1 = &obj1;
    ptr2 = &obj2;
// Point the ptr to obj.
// &obj (here & operator will assign
// address of obj to the ptr. So, now
// ptr will ~~ptr~~ point to that obj :)

// Now set, get, add fun can be called
// by using ptr & arrow operator.

    ptr1 -> set (1, 10, 20) // same as:
                               obj1.set(1,10,20);
    ptr1 -> get();          // obj1.get();
    ptr1 -> add();          // obj1.add();

    ptr2 -> set (2, 50, 50);
    ptr2 -> get ();
    ptr2 -> add ();

} // main ends.
```

output

```
1,10,20
30
2,50,50
100.
```

<u>Remember</u>: You can mix the use of ptr & obj. ~~It is not necessary to call get using ptr~~, For eg, After calling set using <u>ptr</u>, you can use <u>obj</u> to call get fun.