~~Object Passing~~

Data Hiding

Q What all tasks can you perform on the variables of class?

A Tasks like printing, assigning values, adding, finding max etc.

Q How many ways can we perform these tasks?

A Two ways:

1) Thru variables
2) Thru fun.

eg:-

Q Out of these

Task	Thru variables	Thru fun.
1) assign values	obj1.rn = 1 obj1.m1 = 10 obj1.m2 = 20	obj1.set(1,10,20);
2) Print	cout << obj1.rn << obj1.m1 << obj1.m1;	obj1.get();
3) add	cout << obj1.m1 + obj1.m2;	obj1.add();
4) find max	if (obj1.m1 > obj1.m2) cout << obj1.m1 else cout << obj1.m2;	obj1.max();

Q Out of these two ways, which is the prefered way to perform these tasks? Why?

A

A Preferred way is thru fun.
Because when accesing variables, there is a chance of unintentional mistakes

for eg instead of writing

obj1.m1 + obj1.m2

user may write (by mistake)

obj1.m1 = obj1.m2;

- So to add two variables, use fun

obj1.add( );

// There is no chance of making mistakes.

Q How can we avoid such mistakes?

A To avoid such unintentional mistakes, the accers to variables is disabled /by declaring them as private.
If you declare var as private, they can't be accesed like this

- obj1.m1 + obj1.m2 // Not allowed for private var.

- obj1.add() // Fun must be used to accen var.

Q Can you perform these tasks thru var if var are declared as private?

A No. Then fun must be used to perform these tasks.

Q What is data hiding? why is it done?

A DH means to declare variables in the class as private.
This is done so that objects cant access them directly.
It is a feature of OOPs to avoid unintentional modification of variables.

Private Member Variables

Q What happens when you declare the variables as private?

A Objects can't access the var.

For eg `Cout<<obj1.m1;`
`// this is not allowed.`

Q How do you access the var, if you declare them as private?

A Private var will be accessed by using fun.

For eg:- `obj1.add();`
`obj1.get();`

Q What is the reason behind declaring var as private?

A To avoid unintentional mistakes.

For eg `obj1.m1 + obj1.m2`

mistakenly written
- instead of +.

Q what challenge do you face if you declare var as private?

A Now you cannot perform operations on variables of different objects.

For eg If you want to add m1 of obj1 and m1 of obj2, then this is not allowed; because var r private.

obj1.m1 + obj2.m1

- It becomes impossible to perform inter-object operations.

Q How do you overcome this challenge?

A. To overcome this challenge, obj passing is done in fun.

- obj passing is done to perform inter-object operations.

Access Specifier

Q How many Access specifiers r there?

A There r three acces specifiers

1) Public
2) Private
3) Protected.

1) Private :

Private members cant be accesed by objects

eg obj1.m1; // not allowed as m1 is private

2) Public :-

Public members can be accesed by obj

eg ~~obj1.ge~~
obj1.get(); // Allowed as get() is public.

3) Protected :-

Related to inheritence.