object-oriented programming

(OOP)

KIAN _ ACADEMY

Constructor :-

special methods in class Why ?

- 1 - name is the same name of the class
- 2 - has not a return type
- 3 - must not return any values

Note **:-**

1 - The constructor is called automatically when an object is created.

2 - Constructors are typically public

Why We are using constructor ?

1 - to initialize the data of an object

2 - to creat objects

Example :-

public :

Rectangle( )

{

length 6;

width =8;

}

****

Constructors – Initialization list

public :
Person( ): name("karim"), age(20)
{
cout <<"person is : "<<name<<" And his
age is :"<<age<<endl;
}

Output:
person is : karim And his age is :20

Overloading Methods and Constructors:

- → Two or more methods in a class may have the same name BUT their signatures are different.
- → Method signature (No of Args – Types of Args – Order of Args)

Example :-

```
int add(int num1, int num2)
{
int sum = num1 + num2;
return sum;
}
Float add(float num1, float num2)
{
float sum = num1 + num2;
return sum;
}
```

```
int add(int num1, int num2, int num3)
{

int sum = num1 + num2 + num3 ;
return sum;
}
```

Default constructor (no-arg) :

The default constructor is used to provide the default values to the object like 0, null

---

Example:

```
Rectangle::Rectangle ( ):length(0),width(0)
{
}
```

Note :-

If there is no constructor in a class , compiler automatically creates a default Constructor

Overloaded constructor:

```
Rectangle::Rectangle(float l , float w):length(l),width(w)
{
}
*In The Main:
Rectangle box1() ;  //Default constructor
Rectangle box2(5.0, 10.0);
//Overloaded constructor
```

What is the use of a constructor ?

- **Constructor** is a special function having same name as class name.
- **Constructor** is called at the time of creating object to your class.
- **Constructor** is used to initialize the instance variables of an object while creating it.
- **Constructor** is also used to create virtual tables for virtual functions.

Exam questions:

- **(1) What is the role of a constructor in classes?**

a) To modify the data whenever required

b) To destroy an object

c) To initialize the data members of an object when it is created

d) To call private functions from the outer world

- **(2) Why constructors areefficient instead of a function init() defined by the user to initialize the data members of an object?**

a) Because user may forget to call init() using that object leading segmentation fault

b) Because user may call init() more than once which leads to overwriting values

c) Because user may forget to define init() function

d) All of the mentioned

- **(3) What happens if a user forgets to define a constructor inside a class?**

a) Error occurs

b) Segmentation fault

c) Objects are not created properly

d) Compiler provides a default constructor to avoid faults/errors

- **(4) How constructors are different from other member functions of the class?**

a) Constructor has the same name as the class itself

b) Constructors do not return anything

c) Constructors are automatically called when an object is created

d) All of the mentioned

Answer the questions :-

- 1 - c
- 2 - d
- 3 - d
- 4 - d