FREEMIND

Date

Page

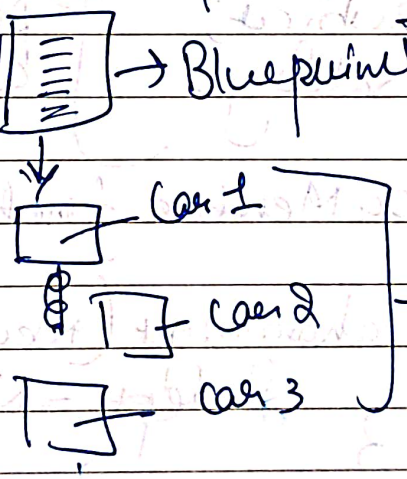
OOPS - C++
Object Oriented Programming System

→ Object and Object related information.

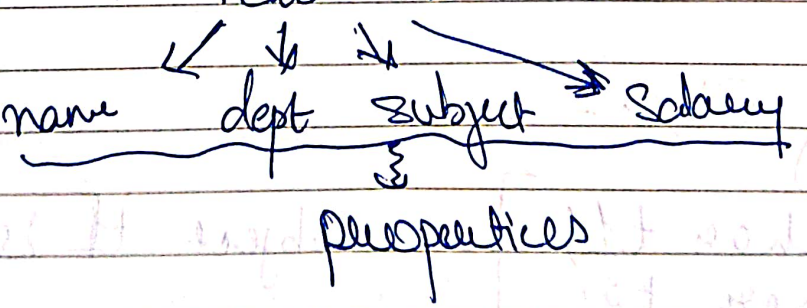
→ Libraries like vector, string, stack exist in STL because of OOPS.

Classes & Objects

- Object are entities in real world
  - Like → pen, laptop etc.
- Class is like a blueprint of these entities.
- Toyota → Blueprint → Class



- Object ← Teacher → change dept() → method



FREEMIND

Date

Page

- class → Blueprint
  - obj 1
  - obj 2
  - obj 3

  Now we can create as many as we want.

Syntax

1 Creating a Class

By Taking an example of Teacher

```
class Teacher {
    String name;
    String dept;
    String subject;
    double salary;
```

→ These are properties or attributes

```
// Method or Member function
    void changedept(Newdept) {
        dept = Newdept;
    }
};
```

2 Creating Object

```
Teacher t1;
Teacher t2;
```

Object t1 is created

Hari t1 automatically has all the properties.

By default when we will execute this programme it will show error because everything is private here by default.
To execute it successfully we use Access Modifier.

Access Modifier :
There are 3 types of Access Modifier

(i) private → By Default

(ii) public → Accessible to everyone

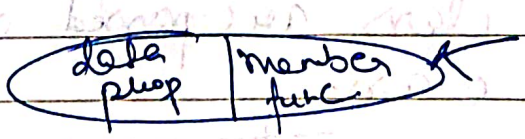
(iii) protected → Accessible inside class and to its derived class → Inheritance.

---

4 pillars of OOPS

(i) Encapsulation (iii) Inheritance
(ii) Abstraction (iv) Polymorphism.

ENCAPSULATION
wrapping up data & member function in single unit called class.



It helps in data hiding - private access modifier.

FREEMIND

Date

Page

Constructor — Special method invoked automatically at time of obj creation.

→ Used for Initialization
→ Same name as Class
→ Constructor doesn't have returntype
→ Only called once (auto), at obj creation.
→ Memory allocation happen when const is called.
→ Always declared publicly

Constructor → Non Parameterized
→ Parameterized.
→ Copy

→ One Class can have multiple constructors also.

Destructor → Opposite of Constructor
↓ ↓
Deallocate. allocate.

→ Destructor has same name as class.
→ Automatically called.

Inheritance ⇒ Properties & member function of base class are passed onto derived class.

Class A → Parent, Base
↓ Inherit
class B ⇝ child, derived

Code Reusability

FREEMIND

Date

Page

Types of Inheritance:
1. Single / Normal
2. Multilevel
3. Two or more at once
4. Hybrid
5. Hierarchial Inheritance.

Polymorphism

Polymorphism is the ability of object to take on different forms or behave in different ways depending on context in which they are used.

Types of Polymorphism: (i) Compile Time
(ii) Run Time

Compile Time — (i) Constructor Overloading
(ii) Function Overloading

Function Overloading → Class & para
same func name but func (1) para
diff func para. func (2) para.

int x = y
int y = 10 → operator Overloading

Run Time Polymorphism :

⇒ Function Over Riding

Overloading → Ek hi class me 2 function

Overriding → parent & child class both / Inheritance

⇒ Parent & child both contain same function with different Implementation.

The parent class is said to be overridden.

P → C
overridden ↳ override

Virtual functions - Run Time
Dynamic In Nature
Keyword → "virtual" inside base class and are always declared with base class & overridden in child class.

Called During Run Time.

ABSTRACTION - using abstract classes

① Used to provide base class from which other class can be derived

② Cannot be instantiated & are meant to be inherited

③ Typically used to define an interface for derived classes.

FREEMIND

Date

Page

Static Keyword

- Static Variable

⇒ Variable declared as Static in function are created & initialized once for a lifetime of program

// In function

⇒ Static variable in class are created & init once. They are shared by all obj of class.

// In class

- Static Object → Once for Lifetime of programme.