

# Object Oriented Programming

Composition and Aggregation

# Composition

- An object may be composed of other smaller objects, the relationship between the “part” objects and the “whole” object is known as composition.
- Composition is stronger relationship.
- Composition is a stronger relationship, because
  - the composed object becomes a part of the composer.
  - the composed object can't exist independently.

```
class Engine{
    private:
        string eng_num;

    public:
        void set_eng_num(string num)
        {
            eng_num = num;
        }
        string get_eng_num(){
            return eng_num;}

};
```

```
class Car{
    private:
        string car_num;
        Engine engine;
    public:
        Car(){
            car_num="111";
            engine.set_eng_num("222");
        }
        void show_Car(){
            cout<<car_num;
            cout<<engine.get_eng_num();
        }

};

main(){
    Car c1;
    c1.show_Car();
}
```

# Aggregation

- An object may contain a collection (aggregate) of other objects, the relationship between the container and the contained object is called aggregation.
- Aggregation is represented by a line with unfilled-diamond head towards the container

```
class Address{
    private:
        int house_num;
        int street_num;
    public:
        void set_house_num(int i)
        {
            house_num = i;
        }
        int get_house_num(){
            return house_num;
        }
        void set_street_num(int i)
        {
            street_num = i;
        }
        int get_street_num(){
            return street_num;
        }
};
```

```
class Student{
    private:
        string name;
        Address *address;

    public:
        Student(Address address1, string name1)
        {
            name=name1;
            address=&address1;
            address->set_house_num(1);
            address->set_street_num(5);
        }
        void show()
        {
            cout<<name;
            cout<< address->get_house_num();
            cout<< address->get_street_num();
        }
};
```

```
int main(){  
    string name1= "student's name ";  
    Address address2;  
  
    Student s1(address2, name1);  
    s1.show();  
  
    return 0;  
}
```

- Aggregation is weaker relationship.
- Aggregation is weaker relationship, because aggregate object is not a part of the container.
- Aggregate object can exist independently.

# Class exercise

- Write a C++ program that has a class named Clinic which has at least three doctors and ten patients. The clinic class also keeps the information of clinic's name and address. Each patient and doctor is allotted an Id which is generated order-wise, e.g., the patient 1 is allotted the id 1, patient 2 is allotted id 2, and so on. Similarly, the doctors are also allotted the Ids.
- The information required to be kept for a doctor include doctor's name, Id, and his/her specialization while the information related to a patient include patient's name, Id, age, and disease.
- The class clinic should take input of 3 doctors and 10 patients from user and display that on screen.
- Implement the given problem by applying aggregation/composition.

Note: recall aggregation/composition, array of objects, constructor, static members.



# References

- C++ How to Program  
By Deitel & Deitel
- The C++ Programming Language  
By Bjarne Stroustrup
- Object oriented programming using C++ by Tasleem Mustafa, Imran Saeed, Tariq Mehmood, Ahsan Raza
- <https://www.tutorialspoint.com/cplusplus>
- <http://ecomputernotes.com/cpp/introduction-to-oop>
- [https://www.tutorialspoint.com/cplusplus/cpp\\_inheritance.htm](https://www.tutorialspoint.com/cplusplus/cpp_inheritance.htm)
- <https://www.guru99.com/c-loop-statement.html>
- [www.w3schools.com](http://www.w3schools.com)
- <https://www.geeksforgeeks.org/>