- 1. Inheritance
- 2. Encapsulation
- 3. Abstraction
- 4. Polymorphism
- 5. Method Overriding
- 6. Method Overloading
- 7. Objects
- 8. Classes
- 9. Constructors and Destructors

Inheritance

Inheritance is a feature to reuse the existing class without makin any changes in it.

Syntax: InheritingClass : InheritedClass

```
#include <iostream>
using namespace std;

class Papa {
public :
    int a = 10;
};

class Beta : public Papa {
public :
    int b = 20;
};

int main() {
    Beta obj;
}
```

```
cout << obj.a + obj.b<< endl;
}</pre>
```

In above program, Beta class inherited Papa class.

2. Encapsulation

Encapsulation is defined as binding together the data and the functions that manipulates them.



```
private:
    int a;
    public:
        void show() {
            cout<<a;
        }
};
```

In above program,

class Encapsulation binds together variable a and function show().

3. Abstraction

Abstraction refers to providing only essential information to the outside world and hiding their background details.

For example,

```
#include <iostream>
using namespace std;
int main() {
   cout << "Abstraction";
   return 0;
}</pre>
```

In above program,

User dont need to understand how cout works. User should know only how to use it.