c++ 语法总结

1.引用：

> 语法：**类型 &变量名 = 已有变量名;**

```
int a = 10;
//引用的初始化
int &b = a;    //b是a的引用,b和a指向同一个内存
```

a和b指向同一个内存，b相当于a的别名，修改a或者b都会修改二者的值。

应用：
函数使用引入传参可以改变实参的值

```
void swap(int& a,int& b){
    int t=a;
    a=b;
    b=t;
}
int main(){
    int a=10,b=20;
    swap(a,b);                  // a = 20, b = 10; 注意调用时不用&
    cout << a << " " << b << endl;
}
```

2.结构体

- **方式1**

```
struct Node{
    int id;
    char name[10];
}node;

int main(){
    //声明
    node a;
    //赋值
    a.id = 100;
    strcpy(a.name,"tianle");
}
```

- **方式2** *typedef*

```
typedef struct Node{
    int id;
    string name; // c: char name[10];
    struct Node* next;
}node;

int main(){
    //利用指针一定先开空间
    node* a = new node; //c++
    node* a = (node*)malloc(sizeof(node)); //c

    //赋值
    a->id = 100;
    a->name = "tianle"; //c++
    strcpy(a->name,"tianle"); //c
    a->next = NULL;
}
```

3.输入输出

- **输入（cin >>)**
- **输出（cout <<)**
- **换行（endl)**

```
int main(){
    int a = 22379230;
    string b = "zjw";
    cin >> a >> b;              // 依次输入整数a,字符串b
    cout << "输出是: " << endl;  // 换行
    cout << a << " " << b ;     // 依次输出 a 空格 b
}
```

输出是:
22379230 zjw

4.类 class

- **类的声明:**

```
class 类名{
类的属性:   // public、private
    成员变量
    成员函数(方法)
};

//类外实现成员函数
返回值 类名::成员函数名(参数列表){
    //函数体
}
```

- **实例化:**

```
int main(){
    类名 对象名;   // 声明对象，对象名可以自定义，类名是定义的类名。
    对象名.成员变量 = 值;   // 给成员变量赋值，注意成员变量是类中的变量。
    对象名.成员函数();   // 调用成员函数，注意成员函数是类中的函数。
}
```

- **示例:**

```
class Dog{ // 定义一个狗类
public:
    int age;
    string name; //狗类一共2个成员变量：  狗年龄、狗名

    void initial(int x,string y){ //定义狗的初始化函数
        age = x;
        name = y;
    }
    void bark(); // 可以不在类中实现函数，但要声明
};

//在类外实现bark函数，需要先在类中声明
void Dog::bark(){
        cout << "woaf!" << endl;
}


int main() {
    Dog a;
    Dog b;

    //利用自己写的initial方法初始化狗a
    a.initial(12,"nan gua");
    cout << a.age << a.name << endl;
    a.bark();

    //利用成员变量直接给狗b赋值
    b.age = 19;
    b.name = "lele";
    cout << b.age << b.name << endl;
    b.bark();

    return 0;
}
```