```cpp
//Saiyan.h


#ifndef SDDS_SAIYAN_H
#define SDDS_SAIYAN_H

#include <iostream>
#include <cstring>

const int MAX_LENGTH = 30;

namespace sdds
{
    class Saiyan
    {
        char m_name[MAX_LENGTH + 1];
        int m_dob;
        int m_power;
        bool m_super;

    public:
        //constructors
        Saiyan();
        Saiyan(const char* nam, int date, int pow);
        //
        bool isSuper() const;
        bool isValid() const;
        void setEmpty();
        void display() const;
        void set(const char* name, int dob, int power, bool super = false);
        bool hasLost(const Saiyan& other) const;
    };


}


#endif


//Saiyan.cpp


#define _CRT_SECURE_NO_WARNINGS


#include <iostream>
#include "Saiyan.h"
using namespace std;

namespace sdds
{
    Saiyan::Saiyan() {
        setEmpty();
    }

    Saiyan::Saiyan(const char* nam, int date, int pow) {
        set(nam, date, pow);
      // this->m_super = false;
```

```cpp
        }

        bool Saiyan::isSuper() const {
            return m_super;
        }

        bool Saiyan::isValid() const {

            bool valid = false;


            if (m_name != nullptr && m_name[0] != '\0' && m_dob > 0 && m_dob < 2020 &&
    m_power > 0) {

                valid = true;
            }

            return valid;
        }

        void Saiyan::setEmpty() {
            m_name[0] = '\0';
            m_dob = 0;
            m_power = 0;
            m_super = false;
        }

        void Saiyan::display() const {
            if (!isValid()) {
                cout << "Invalid Saiyan!" << endl;
            }
            else {
                cout << m_name << endl;
                cout << "DOB: " << m_dob;
                cout << " Power: " << m_power << endl;
                cout << "Super Saiyan Ability: " << (isSuper() ? "S" : "Not s") <<
    "uper Saiyan." << endl;
                cout << endl;
            }
        }

        void Saiyan::set(const char* name, int dob, int power, bool super) {
            if(!name || strlen(name) == 0 || dob > 2020 || power < 0) {
            //if ( name != nullptr && name[0] != '\0' && dob < 2020 && power > 0) {
                setEmpty();

            }
            else {
                strcpy(m_name, name);
                m_dob = dob;
                m_power = power;
                m_super = super;
            }
            }


        bool Saiyan::hasLost(const Saiyan& other) const {
            bool result = false;
```

```
            if (!this->isValid() || !other.isValid()) {
                result = false;
            }

            if (this->m_power < other.m_power) {
                result = true;
            }

            return result;
        }

    }
```