

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

struct Monster
{
    string Name;
    bool Friendly;
    int Age;
    float Height;
    char Race;
};

int main()
{
    Monster M1; //New data type...

    M1.Name = "Zorkie";
    M1.Friendly = true;
    M1.Height = 5.46;
    M1.Age = 124;
    M1.Race = 'N';
}
```

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

class Monster //A blueprint for the monster class.
{
    public: //Listing of properties or attributes.
    string Name;
    bool Friendly;
    int Age;
    float Height;
    char Race;
};

int main()
{
    Monster M1; //New data type...

    M1.Name = "Zorkie";
    M1.Friendly = true;
    M1.Height = 5.46;
    M1.Age = 124;
    M1.Race = 'N';
}
```

```

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

class Monster //A blueprint for the monster class.
{
    public: //Listing of properties or attributes.
    string Name;
    bool Friendly;
    int Age;
    float Height;
    char Race;
void GetIt( ) // A method.
{
    cout << "Monster Name: ";
    cin >> Name;
    cout << "Is he friendly (1/0): ";
    cin >> Friendly;
    cout << "How old is he? : ";
    cin >> Age;
    cout << "How tall? : ";
    cin >> Height;
    cout << "What is his race? : ";
    cin >> Race;
    cout << "Thanks for your input.";
    cin >> HoldIt;
}
};

int main()
{
    Monster M1, M2;
    M1.GetIt();
    M2.GetIt();
}

```

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

class Monster //A blueprint for the monster class.
{
    public: //Listing of properties or attributes.
    string Name;
    int Age;
};

int main()
{
    Monster M1;
    M1.Name = "Zorkie";
    M1.Age = 123;
}
```

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

class Monster //A blueprint for the monster class.
{
    private: //Listing of properties or attributes.
    string Name;
    int Age;

    public:
    void Input(string N, int A)// A method used for changing values.
    {
        Name = N;
        Age = A;
    }
};

int main()
{
    Monster M1;
    M1.Input("Zorkie", 123);
}
```