

## 一. 创建.h文件

```
#ifndef GOBJECT_SRC_HUMAN_H
#define GOBJECT_SRC_HUMAN_H

#include <glib-object.h>

G_BEGIN_DECLS //如果时c++编译器, 则按c的标准编译
#define HUMAN_TYPE \
    (human_get_type())

#define HUMAN(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST((obj), HUMAN_TYPE, Human))

#define IS_HUMAN(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE((obj), HUMAN_TYPE))

typedef struct
{
    GObject parent;
    gchar *name;
    gint age;
    // void (*voice)(void);
}Human;

typedef struct
{
    GObjectClass parent_class;
    void (*talk)(void);
}HumanClass;

GType human_get_type(void);

Human *human_new(void);

gchar *human_get_name(Human *human);
void human_set_name(Human *human, gchar *name);

gint human_get_age(Human *human);
void human_set_age(Human *human, gint age);

G_END_DECLS
#endif
```

## 二.创建.c文件

```
#include "human.h"

G_DEFINE_TYPE(Human, human, G_TYPE_OBJECT);

enum { HUMAN_TALK, LAST_SIGNAL };
static gint human_signals[LAST_SIGNAL] = { 0 };

void human_talk()
{
    g_printf("human_talk\n");
}
```

```

static void human_init(Human *human)
{
    human->name = "liuhw";
    human->age = 25;
    // human->voice = voice_func;
}

static void human_class_init(HumanClass *humanclass)
{
    humanclass->talk = human_talk;
    human_signals[HUMAN_TALK] = g_signal_new("human-talk",
        HUMAN_TYPE,
        G_SIGNAL_RUN_FIRST,
        G_STRUCT_OFFSET(HumanClass, talk),
        NULL,
        NULL,
        g_cclosure_marshal_VOID__VOID,
        G_TYPE_NONE,
        0,
        NULL);
}

Human *human_new(void)
{
    return HUMAN(g_object_new(HUMAN_TYPE, NULL));
}

gchar *human_get_name(Human *human)
{
    g_signal_emit_by_name(human, "human-talk");
    return human->name;
}

void human_set_name(Human *human, gchar *name)
{
    human->name = name;
}

gint human_get_age(Human *human)
{
    return human->age;
}

void human_set_age(Human *human, gint age)
{
    human->age = age;
}

```

### 三.测试文件

```

#include "human.h"

int main(int argc, char *argv[])
{
    Human *human;
    human = human_new();
    gchar *name = NULL;
    gint age = 0;
    name = human_get_name(human);
    age = human_get_age(human);
    g_printf("name = %s\n", name);
}

```

```
g_printf("age = %d\n", age);
g_signal_emit_by_name(human, "human-talk");
return 0;

}
```