

字符串、通讯、协议等基础概念和简单实现

作为计算机程序员或者从业人员，对字符串、通讯、协议这些，应该具有一些基本概念。下面我举一些例子来简单说说，希望以此为敲门砖，使一部分爱学习的学生能够取得进步，解决实际问题。

这里，我们说到了“协议”这个概念。不要觉得协议是多么高大上的东西，它本质上就是一种约定。比如你早上离家前跟妈妈说：今天晚上我不回来吃晚饭。这就是一个约定，你妈得知这个消息，就不做你的饭，或者不等你回来吃饭了。如果你说：今天晚上我可能迟点回来。你妈得知这个消息，就不等你了，但还是为你准备好晚饭，从程序员的角度来说，这个叫做“超时”（timeout）。

一、什么是字符串呢？

我们知道，对计算机来说，没有什么字符串不字符串的概念，它全是二进制数据。所谓“字符串”，其实只是我们程序员的一个约定，也可以理解为一个协议。字符串被定义为由可视字符组成，并且以'\0'结尾的一块内存区域。这里说的“可视字符”，并不是人的肉眼一定可识别的符号，有些肉眼直接看并不能识别，但不妨碍它是可视字符，通常称之为“乱码”。即使是字符串，也有不同字符集的不同约定（协议）。

请记住'\0'，它是字符串核心中的核心，所谓字符串，必须用它结尾。

“abcd”，这是一个字符串，ansi 英文字符集下，它虽然是 4 个字符，但不要忘了，它应该还有一个'\0'结尾，所以，它至少占 5 个字节的空間。如果是 unicode 字符集下，一个字符占 2 个字节，它实际就是占 10 个字节的空間。示意图如下：

Ansi:

a	b	c	d	\0
---	---	---	---	----

Unicode:

a	\0	b	\0	c	\0	d	\0	\0	\0
---	----	---	----	---	----	---	----	----	----

下面的例子，我们都用 ansi 来说。字符串的约定（定义）清楚了，求这个字符串的长度，那就简单了，给字的字符串变量，它实际上就是指向开始的一个指针位置。求长度，就是从开头起判断，到'\0'为止，中间就是长度。

```
String ls_str = "abcd"
```

```
Int li_len
```

```
li_len = len(ls_str)
```

这里的 len() 函数，它实际是这样工作的：

```
Int len(char *str)
```

```
{
    int n = 0;
    char*p = str;
    while(*p++ != '\0')
        n++;
    return(n);
}
```

它就是循环整个字符串，计算长度。显然，这个求长度的过程，效率有点低下。所以，还有另外一种表示方式：我可以定义一个长度，放入内容，然后连这个长度一起算作一个单元，这就是 blob 类型。

```
int blob.len = 4;
```

```
char* blob.data = "abcd";
```

我们用 len(blob) 取长度时，它直接返回 blob.len 变量，效率就非常高了。而且，你发现没有，它根本就没有管'\0'这个东西，它就表示一段内存的长度，所以 blob 能保存所有数据，并不仅限于肉眼可视的符号。

二、通讯就如何发生的？

生活在现代社会，通讯无处不在。但普通人对通讯是如何发生的，根本无感。我们拿手机上网、打电话、查资料，连上设备开始工作，这些后面实际上都是通讯。而通讯的基础就是协议。我们上网什么的，

使用的是 tcp/ip 协议，这是底层协议，在它的基础上，我们使用最多的是应用层的 http 协议。

当然实际生活中，还有更多的协议。有些是你正常使用但你没意识到那是协议的内容。例如：你给你的朋友分批次发了 3 件货过去，快递揽单，你通知你朋友你发了 3 件货，你朋友收货时就按 3 件收，这样就不会遗漏。如果你不通知你朋友，那你朋友就有可能漏收。这实际上就是一个具体的协议。快递公司只管收货发货，不管你的实际内容，你发的是字符串还是发的图片或者其他什么，快递公司是不管的。它只管按你约定的件数发过去，你朋友按照你约定的收，收足了，就算快递公司完成任务了。

接下来，我们来简单制订一个串口的通讯协议，协议双方一个是服务端一个是客户端。通常请求由客户端发起。制定一个协议如下：

客户端命令：

01 请求

01 01 请求姓名

01 02 请求照片

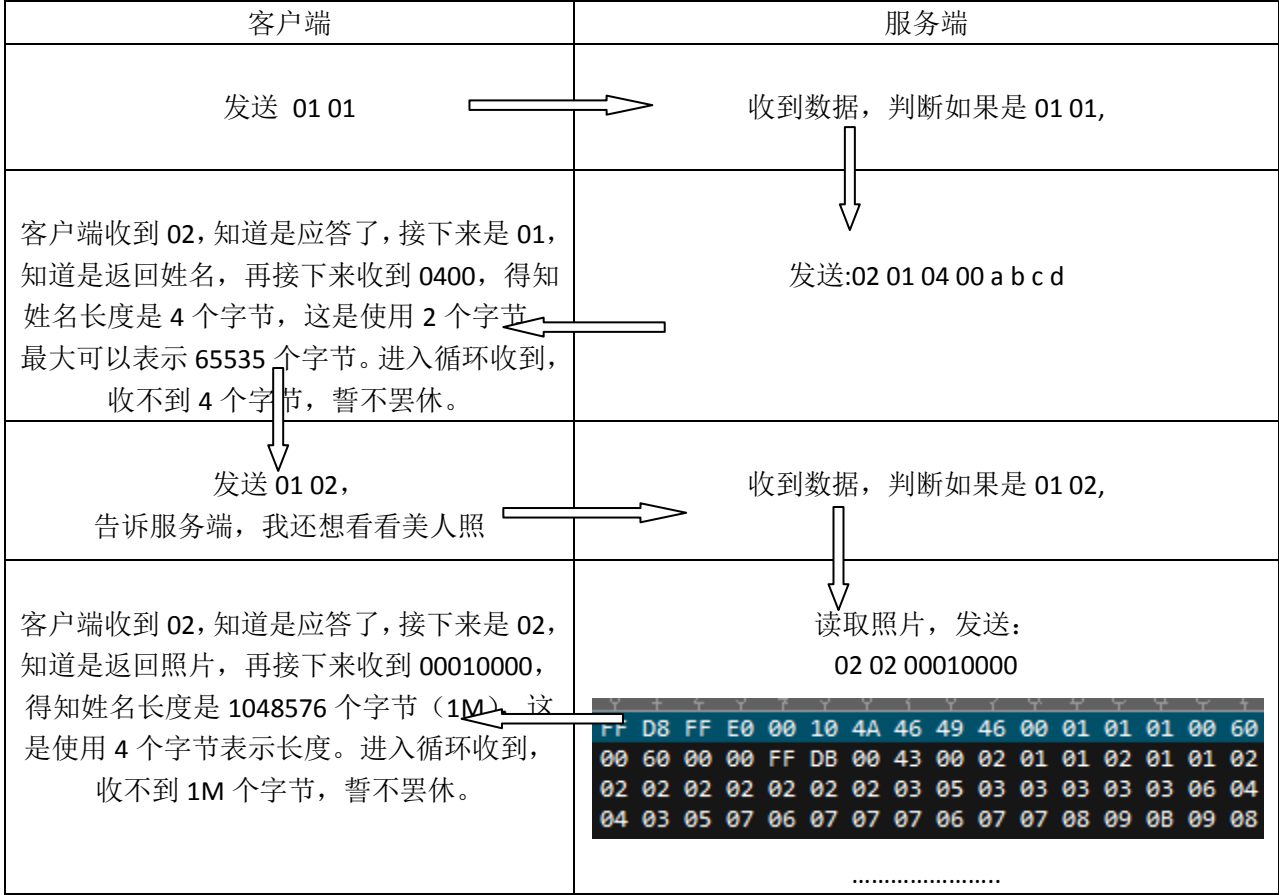
服务端命令：

02 应答

02 01 应答姓名

02 02 应答照片

接下来，我们按照这个协议进行通讯



按照以上步骤，完成所有通讯任务。在整个任务中，通讯本身只是象快递公司一样，按照约定帮你发送内容，至于内容是什么，是表示姓名的字符串，还是可以赏心悦目的大美女，它是不关心的，那是你业务该关心的事情。