**输入**

input(prompt)：读取输入字符串，并去掉末尾的newline标记

**输出**

一、两种基本输出字符串方式：

1.print()函数打印输出

2.file-object对象的write()函数，保存到文本

二、字符串格式化：

1.formatted string literals

f”I love {变量名}”

2.str的format()函数

“I love {参数定位:格式} ”.format(变量名)

3.%表示法

“I love %格式”%变量名

保存到文本

open(filename,mode)：打开文本，返回file项目

filename：文本文件名

mode：文本操作模式。

r -- 只读(默认)

w -- 只写，截断已存在文本

a -- 只写，添加到文末

b -- 二进制模式

t -- 文本模式(默认)

+ -- 添加可读或可写

在文本模式中，文本文件使用依赖于系统环境的换行符(Unix的\n，Windows的\r\n)，编译器统一使用\n

使用with语句结合open()，可以使文件在使用完成后自动close

例：

with open(“vi.txt”) as f：

read\_data = f.read()

file object方法：

read(size)：用于从position处开始读取指定字节数，返回读取字符串内容。size不提供，则读取全部

readline()：从position处开始读取一行字符串，返回读取内容。

readlines()：从position处开始读取所有字符串，以list形式返回读取内容，list的每个item为一行数据。

write(string)：用于写入string，返回写入字节数。

seek(offset[,from\_what])：配置position，offset为相对于from\_what(0-beginning|1-position|2-ending)的偏移，from\_what默认为0，并且1/2仅支持binary模式

tell()：获取当前的position