

Scrivere un programma che legge stringhe da tastiera con la funzione input e le scrive in un file di testo denominato tmp.txt. Il programma deve terminare quando la stringa letta da tastiera e' vuota.

```
fout = open('tmp.txt', 'wt')
l = input('Inserisci una stringa da scrivere nel file:')
while l:
    fout.write(l+'\n')
    l = input('Inserisci una stringa da scrivere nel file:')
fout.close()
```

```
fout = open('tmp.txt', 'wt')
while True:
    l = input('Inserisci una stringa da scrivere nel file:')
    if not l:
        break
    fout.write(l+'\n')
fout.close()
```

Scrivere un programma che legge il file scritto in precedenza e mostra a monitor le righe una per una.

```
fin = open('tmp.txt', 'rt')
l = fin.readline()
while l:
    l = l.strip()
    print(l)
    l = fin.readline()
fin.close()
```

Il file dati_es4_1.txt contiene dati di una serie di esperimenti secondo il formato: id giorno ora x y Dove:

- id e' una stringa che individua in modo univoco un campione
- giorno corrisponde al giorno della settimana (intero tra 0 e 6)
- ora corrisponde ad un intero tra 0 e 23
- x e y sono due float

Le diverse righe corrispondono a diversi campioni.

Scrivere un programma che legge il file e crea una lista con il seguente formato:

[[giorno, ora, x, y], ...]

Ad esempio se il contenuto del file e' il seguente:

m1 0 3 1.2 6.7

f1 1 4 1.4 1.3

f2 2 22 1.7 4.7

La lista deve essere: [['m1', 0, 3, 1.2, 6.7], ['f1', 1, 4, 1.4, 1.3], ['f2', 2, 22, 1.7, 4.7],]

```
L = []
fin = open('dati_es4_1.txt', 'rt')
l = fin.readline()
while l:
    l = l.strip()
    if l:
        ls = l.split()
        sample_id = ls[0]
        day = int(ls[1])
        time = int(ls[2])
        x = float(ls[3])
        y = float(ls[4])
        L.append([sample_id, day, time, x, y])
    l = fin.readline()
fin.close()
print(L)
```

Mostrare a monitor i valori x,y del campione con codice 'f1'

```
for sample in L:
    if sample[0] == 'f1':
        print(sample[3], sample[4])
```

Riscrivere il programma in modo da creare invece della lista un dizionario con chiave il codice del paziente. Ciascun elemento del dizionario deve essere a sua volta un dizionario con chiavi giorno, ora, x, y e rispettivi valori.

Nell'esempio precedente il dizionario deve essere: {'m1':{'giorno':0, 'ora':3, 'x':1.2, 'y':6.7}, 'f1':{'giorno':1, 'ora':4, 'x':1.4, 'y':1.3}, 'f2':{'giorno':2, 'ora':22, 'x':1.7, 'y':4.7}, }

```
d = {}
fin = open('dati_es4_1.txt', 'rt')
l = fin.readline()
while l:
    l = l.strip()
    if l:
        ls = l.split()
        sample_id = ls[0]
        day = int(ls[1])
        time = int(ls[2])
        x = float(ls[3])
        y = float(ls[4])
        d[sample_id] = {'giorno':day, 'ora':time, 'x':x, 'y':y}
    l = fin.readline()
fin.close()
print(d)
```

Mostrare a monitor i valori x,y del campione con codice 'f1'

```
print(d['f1']['x'], d['f1']['y'])
```

Scrivere un programma che:

- leggi i dati dal file dati_es4_1.txt
- calcola la media su tutti i campioni del valore x
- calcola la media su tutti i campioni del valore y

Organizzare il programma in funzioni

```
def read_data(file_name):
    d = {}
    fin = open(file_name, 'rt')
    l = fin.readline()
    while l:
        l = l.strip()
        if l:
            ls = l.split()
            sample_id = ls[0]
            day = int(ls[1])
            time = int(ls[2])
            x = float(ls[3])
            y = float(ls[4])
            d[sample_id] = {'giorno':day, 'ora':time, 'x':x, 'y':y}
        l = fin.readline()
    fin.close()
    return d

def average(data, what):
    m = 0
    for sample in data.keys():
        m += data[sample][what]
    return m/len(data)

d = read_data('dati_es4_1.txt')
mx = average(d, 'x')
my = average(d, 'y')
print('<x> = ',mx)
print('<y> = ',my)
```

Il file dati_es4_2.txt ha lo stesso formato del file analizzato in precedenza, ma alcune righe contengono dati sbagliati. Esempio: giorni della settimana esterni all'intervallo [0,7]. Modificare la funzione scritta in precedenza per leggere i dati in modo che generi un'eccezione ValueError in caso di riga con dati sbagliati. Testare la funzione con il file dati_es4_1.txt e dati_es4_2.txt.

```
def read_data(file_name):
    d = {}
    fin = open(file_name, 'rt')
    l = fin.readline()
    while l:
        l = l.strip()
        if l:
            ls = l.split()
            if len(ls) != 5:
                raise ValueError('ERROR: wrong number of columns')
            sample_id = ls[0]
            if not (ls[1].isnumeric() and ls[2].isnumeric() and ls[3].replace('.', '').isnumeric() and ls[4].replace('.', '').isnumeric()):
                raise ValueError('ERROR: not numeric column')
            day = int(ls[1])
            if day < 0 or day > 7:
                raise ValueError('ERROR: wrong week day')
            time = int(ls[2])
            if time < 0 or time > 23:
                raise ValueError('ERROR: wrong time')
            x = float(ls[3])
            y = float(ls[4])
            d[sample_id] = {'giorno':day, 'ora':time, 'x':x, 'y':y}
        l = fin.readline()
    fin.close()
    return d

d = read_data('dati_es4_1.txt')
```