**BÀI TẬP THỰC HÀNH LẬP TRÌNH PYTHON**

**Bài 1:** Viết chương trình tính biểu thức sau: . Với x, y, z được nhập từ bàn phím.

**Bài 2:** Nhập vào độ dài bán kính r của một hình tròn, in ra màn hình chu vi, diện tích của hình tròn đó. Yêu cầu các giá trị hiển thị trên màn hình có 2 chữ số sau dấu thập phân.

**Bài 3:** Nhập vào độ dài ba cạnh *a, b, c* của một tam giác. Tính diện tích của tam giác bằng công thức *Heron* như sau: 

***Gợi ý:*** Tính nửa chu vi *p=(a+b+c)/2*

**Bài 4:** Nhập bốn số x1, x2, x3, x4, tìm số lớn nhất và số nhỏ nhất của bốn số đó.

**Bài 5:** Nhập các hệ số, giải và biện luận các hệ phương trình sau theo quy tắc *Cramer*:

**Bài 6:** Viết chương trình giải phương trình bậc hai: *ax2+bx+c=0*

**Bài 7:** Nhập chỉ số điện kế tháng trước và tháng này. Hãy tính số kW tiêu thụ và tiền điện cho mỗi hộ dựa vào bảng giá sau (áp dụng từ ngày 20/3/2019):

|  |  |  |
| --- | --- | --- |
| Mức | Số kWh sử dụng | Đơn giá (VNĐ) |
| 1 | kWh từ 0 – 50 | 1,678 |
| 2 | kWh từ 51 – 100 | 1,734 |
| 3 | kWh từ 101 – 200 | 2,014 |
| 4 | kWh từ 201 – 300 | 2,536 |
| 5 | kWh từ 301 – 400 | 2,834 |
| 6 | kWh từ 401 trở lên | 2,927 |

**Bài 8:** Nhập độ dài ba cạnh a, b, c. Cho biết a, b, c có phải là ba cạnh của một tam giác không? Nếu là ba cạnh của một tam giác thì cho biết đó làm tam giác gì: đều, cân hay tam giác thường?

**Bài 9:** Nhập một số nguyên dương n (nếu nhập n≤0 thì chương trình phải yêu cầu nhập lại). Sau đó tính các tổng sau:

**Bài 10:** Nhập n, k nguyên đảm bảo phải dương và k≤n. Tính tổ hợp chập k của n theo công thức:

**Bài 11:** Tìm và in lên màn hình tất cả các số nguyên trong phạm vi từ 10 đến 99 sao cho tích của hai chữ số của nó thì bằng hai lần tổng của hai chữ số của nó. Ví dụ: số N=36 có hai chữ số là 3 và 6, và 3\*6 = 2\*(3+6). Tương tự đối với số 44.

**Bài 12:** Số n là hoàn hảo nếu tổng các ước số nhỏ hơn n bằng chính n, ví dụ 6=1+2+3. Viết chương trình liệt kê các số hoàn hảo nhỏ hơn 1000.

**Bài 13:** Nhập một số nguyên n, cho biết đó là số nguyên tố hay không. Nếu không phải là số nguyên tố thì phân tích số đó thành tích các thừa số nguyên tố. Ví dụ nhập n=1470, khi đó tích các thừa số nguyên tố của n=2.3.5.7.7.

**Bài 14:** Nhập một số nguyên n, in ra màn hình tất cả các số nguyên tố nhỏ hơn hoặc bằng n.

**Bài 15:** Nhập vào 1 số nguyên, in ra màn hình số đảo ngược của số đó. Ví dụ nhập số 34679, khi đó hiển thị lên màn hình số: 97643.

**Bài 16:** Viết một chương trình kiểm tra một số *n* có phải là số đối xứng hay không? Số đối xứng là số khi viết ngược cũng bằng chính nó. Ví dụ: 121, 2332,…

***Gợi ý:***Tìm *m* là số đảo ngược của *n*. Nếu *n=m* thì *n* là số đối xứng.

**Bài 18:** Viết chương trình tìm ước chung lớn nhất và bội chung nhỏ nhất của hai số nguyên a và b.

**Bài 19:** Định nghĩa một hàm có input là 2 chuỗi và in chuỗi có độ dài lớn hơn trong giao diện điều khiển. Nếu 2 chuỗi có chiều dài như nhau thì in tất cả các chuỗi theo dòng.

**Bài 20:** Định nghĩa hàm có thể chấp nhận input là số nguyên và in "Đây là một số chẵn" nếu nó chẵn và in "Đây là một số lẻ" nếu là số lẻ.

**Bài 21:** Định nghĩa các hàm dùng để tính tổng, hiệu, tích , thương của hai số nhập từ bàn phím.

**Bài 22:** Định nghĩa hàm kiểm tra một số có phải là số nguyên tố hay không?

**Bài 23:** Định nghĩa hàm kiểm tra một số có phải là số hoàn hảo hay không?

**Bài 24:** Định nghĩa hàm kiểm tra một số có phải là số đối xứng hay không?

**Bài 25:** Định nghĩa hàm phân tích một số thành tích các thừa số nguyên tố.

**Bài 26:** Định nghĩa một hàm có thể tạo *dictionary*, chứa các *key* là số từ 1 đến 20 (bao gồm cả 1 và 20) và các giá trị bình phương của chúng. Hàm chỉ in các giá trị mà thôi.

***Gợi ý:***

* Sử dụng *dict[key]=value* để nhập mục vào *dictionary*.
* Sử dụng toán từ \*\* để lấy bình phương của một số.
* Sử dụng *range()* cho các vòng lặp.
* Sử dụng *keys()* để di lặp các *key* trong *dictionary*. Có thể sử dụng *item()* để nhận cặp *key/value.*

**Bài 27:** Định nghĩa một hàm có thể tạo ra một *dictionary* chứa *key* là những số từ 1 đến 20 (bao gồm cả 1 và 20) và các giá trị bình phương của *key*. Hàm chỉ cần in các *key.*

**Bài 28:** Định nghĩa một hàm có thể tạo và in *list* chứa các giá trị bình phương của các số từ 1 đến 20 (tính cả 1 và 20).

***Gợi ý:***

* Sử dụng toán tử \*\* để lấy giá trị bình phương.
* Sử dụng *range()* cho vòng lặp.
* Sử dụng *list.append()* để thêm giá trị vào *list.*

**Bài 29:** Định nghĩa một hàm có thể tạo *list* chứa các giá trị bình phương của các số từ 1 đến 20 (bao gồm cả 1 và 20) và in 5 mục đầu tiên trong *list.*

***Gợi ý:***

* Sử dụng toán tử \*\* để lấy giá trị bình phương.
* Sử dụng *range()* cho vòng lặp.
* Sử dụng *list.append()* để thêm giá trị vào *list.*
* Sử dụng *[n1:n2]* để cắt *list*

**Bài 30:** Định nghĩa một hàm có thể tạo ra *list* chứa các giá trị bình phương của các số từ 1 đến 20 (bao gồm cả 1 và 20), rồi in 5 mục cuối cùng trong *list*.

**Bài 31:** Viết chương trình chấp nhận một chuỗi số, phân tách bằng dấu phẩy từ giao diện điều khiển, tạo ra một danh sách và một tuple chứa mọi số.

Ví dụ: Đầu vào được cung cấp là 34,67,55,33,12,98 thì đầu ra là:

['34', '67', '55', '33', '12', '98']

('34', '67', '55', '33', '12', '98')

***Code mẫu:***

values=input("Nhập vào các giá trị:")  
 l=values.split(",")  
 t=tuple(l)  
 print (l)  
 print (t)

**Bài 32:** Với số nguyên n nhất định, hãy viết chương trình để tạo ra một dictionary chứa (i, i\*i) như là số nguyên từ 1 đến n (bao gồm cả 1 và n) sau đó in ra dictionary này. Ví dụ: Giả sử số n là 8 thì đầu ra sẽ là: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}.

***Code mẫu:***

def printDict(n):

d=dict()

for i in range(1,n+1):

d[i]=i\*\*2

print (d)

n=int(input("nhap n= "))

printDict(n)

**Bài 33:** Viết một chương trình để in ra tuple chứa các số chẵn được lấy từ tuple (1,2,3,4,5,6,7,8,9,10).

***Gợi ý:***

* Sử dụng for để lặp tuple
* Sử dụng tuple() để tạo tuple từ list

**Bài 34:** Viết chương trình lọc các số chẵn trong danh sách sử dụng hàm filter. Danh sách là [1,2,3,4,5,6,7,8,9,10].

***Code mẫu:***

li = [1,2,3,4,5,6,7,8,9,10]

evenNumbers = list(filter (lambda x: x% 2 == 0, li))

print (evenNumbers)

**Bài 35:** Viết chương trình tạo và in ra tuple chứa các giá trị bình phương của các số từ 1 đến n. Số nguyên dương n được nhập từ bàn phím.

**Bài 36:** Viết chương trình phân tích số nguyên n thành các thừa số nguyên tố. Ví dụ: 50=2x5x5. (Đưa các thừa số nguyên tố phân tích được vào trong list)

**Bài 37:** Viết chương trình tính tần suất các từ được nhập từ bàn phím. Kết quả được xuất ra sau khi đã sắp xếp theo bảng chữ cái. Giả sử:

Đầu vào: The more danger, the more honor

Đầu ra: The:1 danger:1 honor:1 more:2 the:1

***Code mẫu:***

freq = {} # frequency of words in text

line = input()

for word in line.split():

freq[word] = freq.get(word,0)+1

words = sorted(freq.keys())

for w in words:

print ("%s:%d" % (w,freq[w]))

**Bài 38:** Viết chương trình nhập một chuỗi phân tách nhau bởi dấu phẩy và in từng từ trong chuỗi theo thứ tự bảng chữ cái, phân tách nhau bằng dấu phẩy. Ví dụ:

Đầu vào: mary,peter,alice,jack

Đầu ra: alice,jack,mary,peter

***Code mẫu:***

items=[x for x in input("Nhập một chuỗi: ").split(',')]

items.sort()

print (','.join(items))

**Bài 39:** Viết chương trình chấp nhận đầu vào là chuỗi các số nhị phân, phân tách nhau bởi dấu phẩy, kiểm tra xem chúng có chia hết cho 5 không. Sau đó in các số chia hết cho 5 thành dãy phân tách bởi dấu phẩy.

Ví dụ: Đầu vào: Nhập các số nhị phân: 1011,101,0110,010100,1001

Đầu ra: 101,010100

***Code mẫu:***

value = []

items=[x for x in input("Nhập các số nhị phân: ").split(',')]

for p in items:

k=p

p = int(p, 2)

if not int (p%5):

value.append(k)

print (value)

**Bài 40:** Cho trước một số tự nhiên *n*. Tạo một mảng có *n* phần tử mà các phần tử có chỉ số chẵn (bắt đầu từ *0*) là một cấp số cộng bắt đầu từ *2*, công sai bằng *-0.5*; các phần tử có chỉ số lẻ bằng *-1*.

Ví dụ: Với *n=4*, kết quả trả về là mảng [ 2. -1. 1.5 -1. ]. Với *n=5*, kết quả trả về là mảng [ 2. -1. 1.5 -1. 1. ].

***Code mẫu:***

n=int(input("nhập số phần tử n="))

A[[0]]=2.0

for i in range (1,n):

if (i%2==0):

A[[i]]=(A[[i-2]])-0.5

else: A[[i]]=-1

for i in range (0,n):

print(A[[i]])

**Bài 41:** Cho mảng một chiều A bất kì, hiển thị lên màn hình các yêu cầu sau:

* Đếm các số chính phương có trong mảng
* Đếm số lượng số nguyên tố có trong mảng
* Tính tổng và trung bình cộng các phần tử trong mảng
* Sắp xếp mảng theo thứ tự giảm dần.

***Code mẫu:***

#Hướng dẫn tìm số chính phương trong mảng

import numpy as np

def socp(n):

b=math.sqrt(n)

b=round(b)

if (b\*b==n):

return True

else: return False

A=np.array([2,10,1,16,8,9,20,21,25])

print(A)

d=0

k=A.shape[0]

for i in range(0,k):

if (socp(A[i])):

d=d+1

print (d)

**Bài 42:** Cho ma trận A bất kỳ:

* Viết hàm tính tổng các phần tử trên các cột có chỉ số chẵn của ma trận A
* Viết hàm tính tổng các phần tử có cả chỉ số cột và hàng đều chẵn.
* Trong mỗi hàng, ta định nghĩa độ biến động của nó là sự khác nhau giữa giá trị lớn nhất và nhỏ nhất của các phần tử trong hàng đó. Viết hàm trả về độ biến động của tất cả các hàng trong ma trận.

***Gợi ý:*** Lấy số dòng của ma trận A: *A.shape[0];* Lấy số cột: *A.shape[1].*

**Bài 43:** Hãy tạo ma trận A sau một cách nhanh nhất, không dùng cách thủ công ghi từng phần tử.



***Gợi ý:*** Sử dụng np.arange(); Tìm quy luật của ma trận rồi dùng các phép transpose, reshape thích hợp.

***Code mẫu:***

import numpy as np

A = np.arange(1,13)

print('A =', A)

A=A.reshape(4,-1,order="F")

A=A.reshape(3,-1,order="C")

print(A)

**Bài 44:** Frobenius norm của một ma trận được định nghĩa là căn bậc hai của tổng bình phương các phần tử của ma trận. Frobenius norm của một ma trận A được kí hiệu là .



Viết hàm tính Frobenius norm của một ma trận bất kỳ.

***Code mẫu:***

import numpy as np

import math

def Fnorm():

c=A.shape[0]

r=A.shape[1]

print(c)

print(r)

k=0

for i in (0,c-1):

for j in (0,r-1):

k=k+A[i][j]\*A[i][j]

return (math.sqrt(k))

A = np.array([[1,2],[3,4],[5,6]])

print('A =', A)

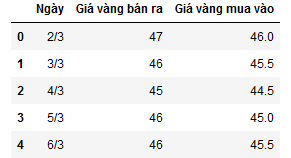
print(Fnorm())

**Bài 45:** Cho bảng dữ liệu về giá vàng trong tháng như sau:

|  |  |  |
| --- | --- | --- |
| Ngày | Giá vàng bán ra | Giá vàng mua vào |
| 1/3 | 46 | 45 |
| 2/3 | 47 | 46 |
| 3/3 | 46 | 45.5 |
| 4/3 | 45 | 44.5 |
| 5/3 | 46 | 45 |
| 6/3 | 46 | 45.5 |
| 7/3 | 45 | 44.5 |

Hãy viết chương trình lưu trữ dữ liệu dưới dạng file CSV.

**Bài 46:** Với dữ liệu trong file CSV đã tạo trong bài 1, viết chương trình đọc và hiển thị dữ liệu từ file CSV này và thêm các tiêu đề đầu vào dữ liệu kết quả như hình sau:



**Bài 47:** Cho bảng dữ liệu về lỗi vi phạm của sinh viên như sau:

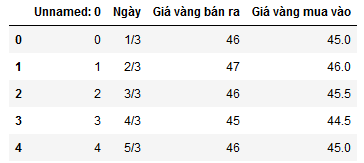
|  |  |  |  |
| --- | --- | --- | --- |
| Tên sinh viên | Vắng lý thuyết | Vắng thực hành | Không nộp bài |
| Trung | 3 | 2 | 0 |
| Cảnh | 5 | 1 | 3 |
| Bảo | 1 | 0 | 0 |
| Quốc | 0 | 0 | 2 |
| Tùng | 2 | 1 | 1 |

Hãy viết chương trình lưu trữ dữ liệu dưới dạng file CSV.

**Bài 48:** Với dữ liệu trong file CSV đã tạo trong bài 3, viết chương trình đọc và hiển thị dữ liệu từ file CSV này và thêm các tiêu đề đầu vào dữ liệu kết quả như hình sau:



**Bài 49:** Với dữ liệu trong file Excel đã tạo trong bài 4.5, viết chương trình đọc và hiển thị dữ liệu từ file Excel này như kết quả hình sau:



**Bài 50:** Cho 2 bảng dữ liệu về giá vàng trong tháng như sau:

Bảng giá vàng tháng 3

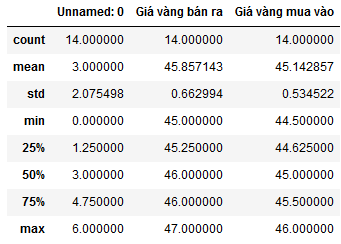
|  |  |  |
| --- | --- | --- |
| Ngày | Giá vàng bán ra | Giá vàng mua vào |
| 1/3 | 46 | 45 |
| 2/3 | 47 | 46 |
| 3/3 | 46 | 45.5 |
| 4/3 | 45 | 44.5 |
| 5/3 | 46 | 45 |
| 6/3 | 46 | 45.5 |
| 7/3 | 45 | 44.5 |

Bảng giá vàng tháng 4

|  |  |  |
| --- | --- | --- |
| Ngày | Giá vàng bán ra | Giá vàng mua vào |
| 4/4 | 48 | 47 |
| 5/4 | 47 | 46.5 |
| 6/4 | 46.5 | 45.5 |
| 7/4 | 46 | 45 |
| 8/4 | 46.5 | 46 |
| 9/4 | 47 | 46 |
| 10/4 | 47.5 | 46.5 |

Hãy viết chương trình lưu trữ dữ liệu của 2 bảng trên vào lần lượt sheet1 và sheet2 của dạng file Excel.

**Bài 51:** Với dữ liệu trong từ 2 file Excel đã tạo trong bài 1 và bài 2, viết chương trình khai thác dữ liệu từ 2 file Excel này với dữ liệu kết quả như hình sau:

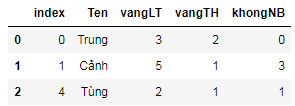


**Bài 52:** Cho bảng dữ liệu về lỗi vi phạm của sinh viên như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên sinh viên | Vắng lý thuyết | Vắng thực hành | Không nộp bài |
| Trung | 3 | 2 | 0 |
| Cảnh | 5 | 1 | 3 |
| Bảo | 1 | 0 | 0 |
| Quốc | 0 | 0 | 2 |
| Tùng | 2 | 1 | 1 |

Hãy viết chương trình lưu trữ dữ liệu dưới dạng file SQL.

**Bài 53:** Với dữ liệu trong file SQL đã tạo trong bài 1, viết chương trình đọc và hiển thị dữ liệu của những sinh viên có số buổi vắng lý thuyết (vangLT) lớn hơn hoặc bằng 2 từ file SQL này như kết quả hình sau:



**Bài 54:** Cho tập tin *dataYear.csv* với dữ liệu về tuổi thọ của mỗi người như sau:

****

Hãy viết chương trình xử lý ngoại lệ trên trường dữ liệu *tuổi* trong khoảng giá trị trung bình của độ lệch chuẩn với dung sai là 1.2 (những giá trị nằm ngoài vùng này sẽ bị loại bỏ). Hiển thị dữ liệu đã được xử lý ngoại lệ với kết quả như hình sau:



**Bài 55:** Với dữ liệu trong file CSV đã cho trong bài 1, hãy viết chương trình tìm và xóa những phần tử khác biệt với những phần tử nằm trong 20% đến 80% của trường tuổi với độ lệch chuẩn và dung sai là 0.9. Xuất file dữ liệu CSV đã được xử lý ngoại lệ với kết quả như hình sau:

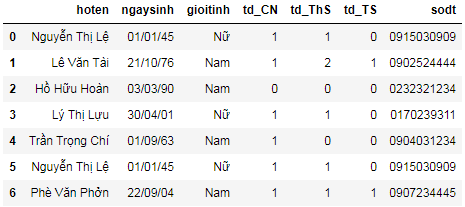


**Bài 56:** Viết chương trình tạo tập tin dữ liệu CSV như hình sau:

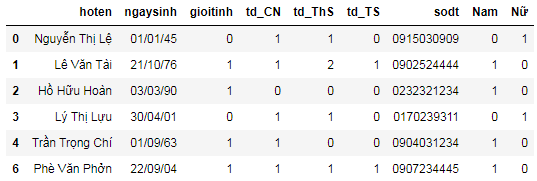


Viết chương trình xóa các ký tự trắng thừa của trường Họ tên, loại bỏ các ký tự “-” và “.” của trường số ĐT.

**Bài 4:** Với dữ liệu trong file CSV đã tạo trong bài 3, viết chương trình chuẩn hóa lại trường dữ liệu Ngày sinh thành định dạng dd/mm/yy, hiển thị dữ liệu kết quả như hình sau:



**Bài 57:** Với tập tin dữ liệu CSV đã được chuẩn hóa như trên bài 3 và bài 4, viết chương trình chia trường Giới tính có 2 giá trị thành 2 trường: “Nam” và “Nữ”, nhận giá trị 0 hoặc 1. Chuyển đổi dữ liệu giá trị cho trường Giới tính: giá trị “Nam” là: 1 và “Nữ” là: 0. Xóa các bản ghi trùng lặp, chỉ giữ lại 1 bản ghi. Lưu và hiển thị dữ liệu kết quả như hình sau:



**Bài 58:** Với tập tin dữ liệu lưu trữ trong bài 5 ở trên, viết chương trình thực hiện các yêu cầu sau:

- Xóa cột Nam và cột Nữ.

- Thêm cột Hưu trí với dãy giá trị cho trước là: [1,0,0,1,0,1].

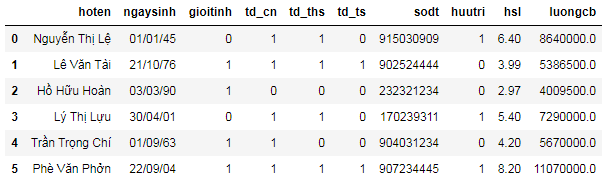
- Thêm cột hệ số lương với giá trị cho trước là: [6.4,3.99,2.97,5.4,4.2,8.2]

- Thêm cột lương cơ bản với giá trị bằng Hệ số lượng\*1350000

- Đổi tất cả tên tiêu đề sang chữ thường.

- Sửa đổi giá trị của trường Trình độ thạc sỹ, nếu giá trị lớn hơn bằng 1 thì kết quả là 1.

- Lưu trữ dưới dạng tập tin CSV và hiển thị dữ liệu như hình sau:



**Bài 59:** Với tập tin dữ liệu CSV lưu trữ trong bài trên, viết chương trình thực hiện các yêu cầu sau:

- Lọc 3 trường dữ liệu: hoten, ngaysinh, gioitinh.

- Lọc 3 dòng đầu tiên của tập tin dữ liệu.

- Hiển thị những bản ghi thỏa điều kiện có trường hoten xuất hiện chữ “Thị”.

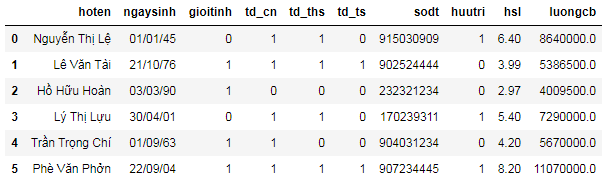
- Lọc những bản ghi có giới tính 1.

- Lọc những bản ghi có giới tính là 0 và hưu trí là 1

- Lọc dữ liệu những bản ghi có giới tính là 0 hoặc trình độ tiến sỹ là 1

- Lấy 3 dòng dữ liệu một cách ngẫu nhiên

**Bài 60:**Với tập tin dữ liệu CSV lưu trữ như trong hình dưới đây:



Viết chương trình thực hiện các yêu cầu sau:

- Đếm số nhân viên có giới tính là Nữ (giá trị 0), số nhân viên có giới tính là Nam (giá trị 1).

- Đếm số nhân viên Nữ và có trình độ Thạc sỹ

- Đếm số nhân viên có giới tính là Nữ hoặc nhân viên có trình độ Tiến sỹ

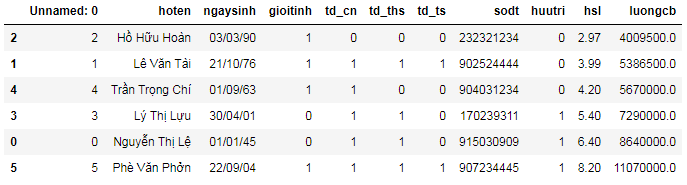
- Tính trung bình trường dữ liệu Lương cơ bản của những người có trình độ là cử nhân.

- Xác định mức lương cơ bản cao nhất của nhân viên có trình độ cử nhân.

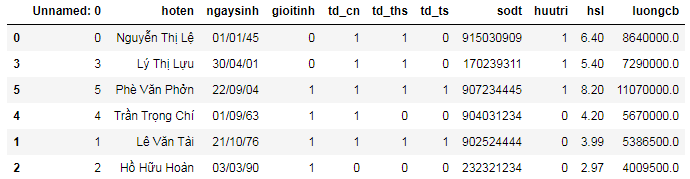
- Xác định mức lương cơ bản thấp nhất của nhân viên nữ có trình độ cử nhân.

**Bài 61:** Với tập tin dữ liệu CSV như trong bài 1 ở trên,

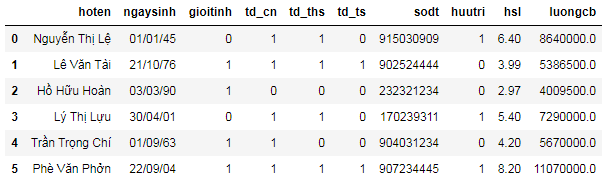
- Viết chương trình thực hiện hiển thị và sắp xếp dữ liệu tăng dần theo cột Lương cơ bản, như hình dưới:



- Viết chương trình thực hiện hiển thị và sắp xếp dữ liệu tăng dần theo cột Giới tính, giảm dần theo cột Lương cơ bản, như hình dưới:

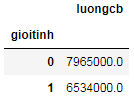


**Bài 62:** Với tập tin dữ liệu CSV lưu trữ như trong hình dưới đây:

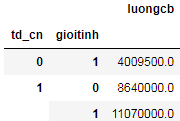


Viết chương trình thực hiện các yêu cầu sau:

- Tạo bảng thống kê tính trung bình của trường Lương cơ bản theo gộp nhóm trường Giới tính, như hình sau:

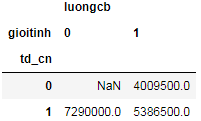


- Tạo bảng thống kê lương cao nhất của trường Lương cơ bản theo gộp nhóm trường trình độ cử nhân và Giới tính, như hình sau:

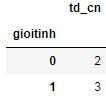


**Bài 63:** Với tập tin dữ liệu CSV như trong bài 1 ở hình trên, viết chương trình thực hiện các yêu cầu sau:

- Tạo bảng thống kê lương thấp nhất của trường Lương cơ bản theo gộp nhóm trường trình độ cử nhân và Giới tính, như hình sau:



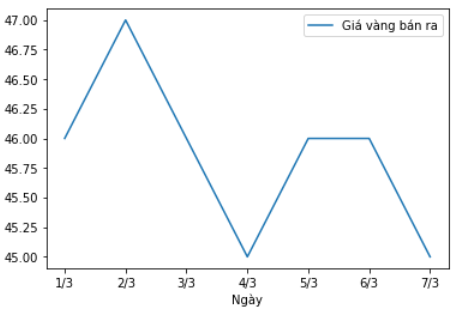
- Tạo bảng thống kê số nhân viên nam, nữ có bằng Cử nhân, như hình sau:



**Bài 63:**Cho bảng dữ liệu về giá vàng trong tháng như sau:

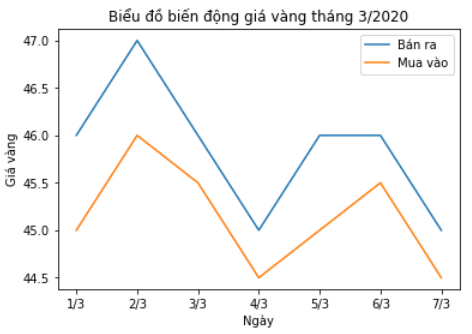
|  |  |  |
| --- | --- | --- |
| Ngày | Giá vàng bán ra | Giá vàng mua vào |
| 1/3 | 46 | 45 |
| 2/3 | 47 | 46 |
| 3/3 | 46 | 45.5 |
| 4/3 | 45 | 44.5 |
| 5/3 | 46 | 45 |
| 6/3 | 46 | 45.5 |
| 7/3 | 45 | 44.5 |

Hãy vẽ biểu đồ đường (line plot) để thể hiện sự biến thiên của giá vàng bán ra, với trục tung là giá vàng, trục hoành là ngày.



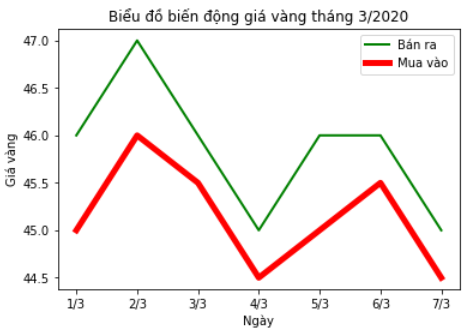
**Bài 64:** Với dữ liệu giá vàng cho ở bài 1, hãy vẽ biểu đồ đường (line plot) để thể hiện sự biến thiên của giá vàng bán ra và mua vào trên cùng một biểu đồ với các yêu cầu sau:

* Nhãn trục tung là: Giá vàng
* Nhãn trục hoành là: Ngày
* Tên biểu đồ là: Biểu đồ biến động giá vàng tháng 3/2020



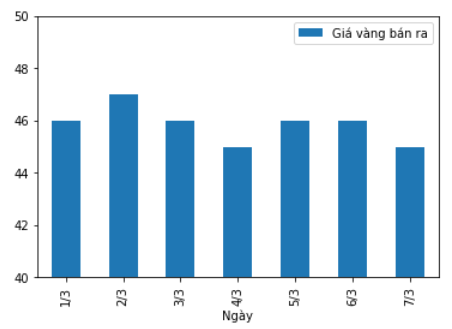
**Bài 65:** Tương tự như bài 2, nhưng với yêu cầu bổ sung sau:

* Đường thể hiện giá vàng bán ra: được vẽ với màu xanh, độ dày 2.
* Đường thể hiện giá vàng mua vào: được vẽ với màu đỏ, độ dày 5.



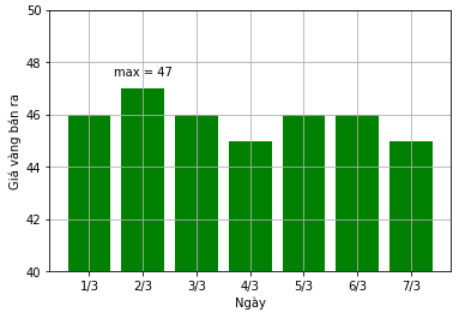
**Bài 66:** Cho dữ liệu giá vàng như bài Line Plot, thể hiện giá vàng bán ra ở dạng biểu đồ thanh (bar plot)

Điều chỉnh trục tung hiển thị khung giá vàng từ 40 đến 50.



**Bài 67:**Vẽ biểu đồ biến động giá vàng bán ra, với các yêu cầu sau:

* Thanh có màu xanh lá cây.
* Nhãn trục tung là: Giá vàng bán ra
* Nhãn trục hoành là: Ngày
* Tên biểu đồ là: Biểu đồ biến động giá vàng tháng 3/2020
* Gán nhãn chú thích cho ngày có giá vàng bán ra cao nhất (max = 47)
* Hiển thị lưới (grid)



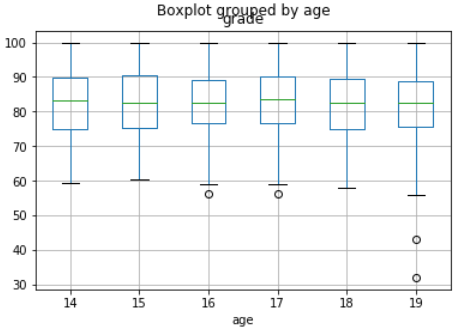
**Bài 68:** Cho bảng dữ liệu về giá vàng trong tháng như sau:

|  |  |  |
| --- | --- | --- |
| Ngày | Giá vàng bán ra | Giá vàng mua vào |
| 1/3 | 46 | 45 |
| 2/3 | 47 | 46 |
| 3/3 | 46 | 45.5 |
| 4/3 | 45 | 44.5 |
| 5/3 | 46 | 45 |
| 6/3 | 46 | 45.5 |
| 7/3 | 45 | 44.5 |

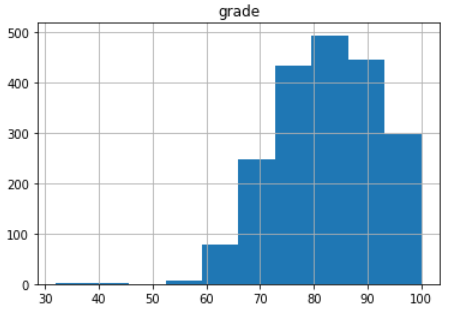
Vẽ biểu đồ biến động giá vàng bán ra với các thanh hiển thị theo chiều ngang (như hình bên dưới):



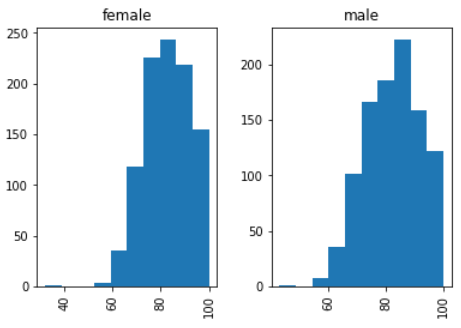
**Bài 69:** Sử dụng biểu đồ hộp (box plot), thể hiện điểm số theo độ tuổi trong bộ dữ liệu datasets/gradedata.csv.



**Bài 70:** Sử dụng dữ liệu datasets/gradedata.csv, vẽ biểu đồ histogram về mức điểm.



**Bài 71:** Sử dụng dữ liệu datasets/gradedata.csv, vẽ biểu đồ histogram về mức điểm theo giới tính.



**Bài 72:** Sử dụng dữ liệu datasets/axisdata.csv, vẽ biểu đồ histogram về số lượng xe bán được (trường Car Sold)

