

**Proiect Pachete Software**  
**Analiza firmei “S.C. Targu Ocna Mare S.A.”**

Stefanescu Mircea

Grupa 1151 ID

## **Introducere**

Firma aleasa este una de dimensiuni mici, ce activeaza in domeniul vanzarii de produse alimentare, in orasul Craiova.

S-au obtinut date referitoare la produsele disponibile si informatiile despre acestea, angajatii firmei si furnizorii aferenti. De asemenea, am putut obtine o statistica a numarului de clienti ce au frecventat magazinul in decursul primelor patru luni de la deschidere si cantitatile din fiecare produs ce au fost cumparate.

Firma a fost infiintata la data de 26-11-2020 si si-a inceput activitatea la data de 13-12-2020.

Fiind aflata la inceput, afacerea nu are dimensiuni foarte mari, produsele oferite fiind orientative, deoarece in functie de cumparaturile realizate de clienti si cererea acestora, magazinul va furniza si alte produse, in alte cantitati. Aprovizionarea se face zilnic pentru produsele de panificatie, legume, fructe si lactate si saptamanal pentru restul produselor, stocul fiind stabilit in functie de cerere.

Dorim sa aflam ce imbunatatiri ar trebui sa realizeze proprietarii afacerii pentru a isi creste profitul si numarul de clienti ce ii frecventeaza, si eventual, pe viitor, sa se extinda in mai multe zone ale orasului.

## **1. Prelucrari in Excel**

In tabela produse, am primit informatii referitoare la:

- produsele disponibile;
- categoria din care fac parte;
- pretul de achizitie;
- stocul;
- furnizorul.

### **Vom calcula:**

- valoare stoc
- pret vanzare
- calculare pret cu tva
- determinare cantitate stoc
- determinare discount
- raportul stocurilor si al preturilor in functie de furnizori

#### **A. Valoare stoc**

Pentru calcularea valorii stocului pentru fiecare tip de produs vom folosi functia matriciala MMULT:

AutoSave On

Firma.xlsx

Search (Alt+Q)

miccea stefanescu

File Home Insert Page Layout Formulas Data Review View Help

Clipboard Font Alignment Number Styles Cells Editing Analysis

G2 =MMULT(D2:E2)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	CRT	Produse Disponibile	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc												
2	1	Mere	Fructe	12	500	Carrefour	=MMULT(D2:E2)												
3	2	Pere	Fructe	11	600	Carrefour													
4	3	Gutui	Fructe	13	1200	Carrefour													
5	4	Ciocolata	Dulciuri	21	30	Carrefour													
6	5	Paine	Fainoase	2	45	Carrefour													
7	6	Seminte	Cereale	3	66	Megaimage													
8	7	Lapte	Lactate	10	75	Megaimage													
9	8	Miere	Dulciuri	33	45	Megaimage													
10	9	Varza	Legume	3	56	Megaimage													
11	10	Cartofi	Legume	2	333	Megaimage													
12	11	Ceapa	Legume	4	230	Megaimage													
13	12	Conopida	Legume	12	121	Megaimage													
14	13	Oua	Lactate	5	420	Megaimage													

Firma

Enter Accessibility: Good to go

Display Settings

100%

## B.Pret vanzare

Stiind adausul comercial de 20 % vom calcula pretul de vanzare astfel: valoare achizitie + adaus comercial

AutoSave On

Firma.xlsx - Saved

Search (Alt+Q)

miccea stefanescu

File Home Insert Page Layout Formulas Data Review View Help

Clipboard Font Alignment Number Styles Cells Editing Analysis

FV =D2+H2

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	CRT	Produse Disponibile	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc	Adaus Comercial	Pret Vanzare								
2	1	Mere	Fructe	12	500	Carrefour	6000	20%	=D2+H2								
3	2	Pere	Fructe	11	600	Carrefour	6600	20%	11,20								
4	3	Gutui	Fructe	13	1200	Carrefour	15600	20%	13,20								
5	4	Ciocolata	Dulciuri	21	30	Carrefour	630	20%	21,20								
6	5	Paine	Fainoase	2	45	Carrefour	90	20%	2,20								
7	6	Seminte	Cereale	3	66	Megaimage	198	20%	3,20								
8	7	Lapte	Lactate	10	75	Megaimage	750	20%	10,20								
9	8	Miere	Dulciuri	33	45	Megaimage	1485	20%	33,20								
10	9	Varza	Legume	3	56	Megaimage	168	20%	3,20								
11	10	Cartofi	Legume	2	333	Megaimage	666	20%	2,20								
12	11	Ceapa	Legume	4	230	Megaimage	920	20%	4,20								
13	12	Conopida	Legume	12	121	Megaimage	1452	20%	12,20								
14	13	Oua	Lactate	5	420	Megaimage	2100	20%	5,20								

Firma

Edit Accessibility: Good to go

Display Settings

100%

## C. Calculare pret cu TVA

	B	C	D	E	F	G	H	I	J	K
	Produce Disponibilă	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc	Adaus Comercial	Pret Vanzare	Unitati Vandute	Pret TVA
2	Mere	Fructe	12	500	Carrefour	6000	20%	12,20	300	=D2*1,19
3	Pere	Fructe	11	600	Carrefour	6600	20%	11,20	450	13,09
4	Gutui	Fructe	13	1200	Carrefour	15600	20%	13,20	845	15,47
5	Ciocolata	Dulciuri	21	30	Carrefour	630	20%	21,20	25	24,99
6	Paine	Fainoase	2	45	Carrefour	90	20%	2,20	36	2,38
7	Seminte	Cereale	3	66	Megaimage	198	20%	3,20	42	3,57
8	Lapte	Lactate	10	75	Megaimage	750	20%	10,20	30	11,9
9	Miere	Dulciuri	33	45	Megaimage	1485	20%	33,20	25	39,27
10	Varza	Legume	3	56	Megaimage	168	20%	3,20	34	3,57
11	Cartofi	Legume	2	333	Megaimage	666	20%	2,20	142	2,38
12	Ceapa	Legume	4	230	Megaimage	920	20%	4,20	132	4,76
13	Conopida	Legume	12	121	Megaimage	1452	20%	12,20	100	14,28
14	Oua	Lactate	5	420	Megaimage	2100	20%	5,20	235	5,95

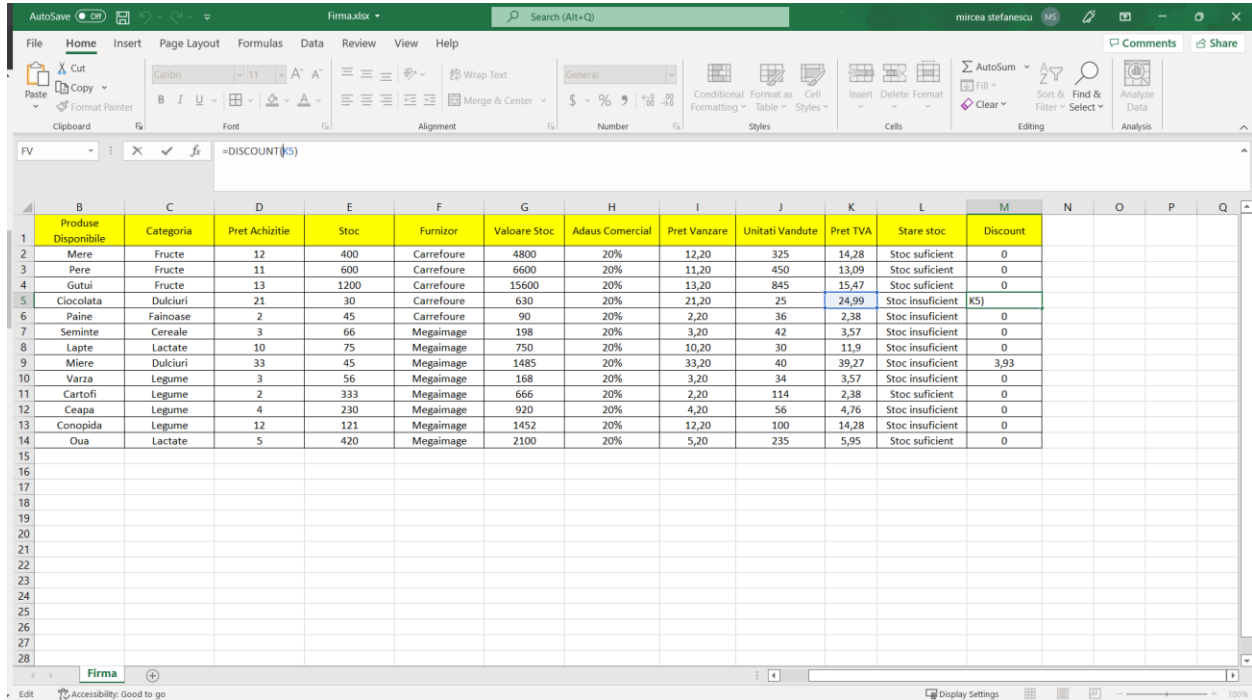
## D. Cantitate stoc

In coloana “L” vom aplica formula “IF” pentru a determina daca stocul de produse de pe fiecare rand este suficient;

	B	C	D	E	F	G	H	I	J	K	L
	Produce Disponibilă	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc	Adaus Comercial	Pret Vanzare	Unitati Vandute	Pret TVA	Stare stoc
2	Mere	Fructe	12	400	Carrefour	4800	20%	12,20	325	14,28	=IF(E2>=300;"Stoc suficient";"Stoc insuficient")
3	Pere	Fructe	11	600	Carrefour	6600	20%	11,20	450	13,09	Stoc suficient
4	Gutui	Fructe	13	1200	Carrefour	15600	20%	13,20	845	15,47	Stoc suficient
5	Ciocolata	Dulciuri	21	30	Carrefour	630	20%	21,20	25	24,99	Stoc insuficient
6	Paine	Fainoase	2	45	Carrefour	90	20%	2,20	36	2,38	Stoc insuficient
7	Seminte	Cereale	3	66	Megaimage	198	20%	3,20	42	3,57	Stoc insuficient
8	Lapte	Lactate	10	75	Megaimage	750	20%	10,20	30	11,9	Stoc insuficient
9	Miere	Dulciuri	33	45	Megaimage	1485	20%	33,20	40	39,27	Stoc insuficient
10	Varza	Legume	3	56	Megaimage	168	20%	3,20	34	3,57	Stoc insuficient
11	Cartofi	Legume	2	333	Megaimage	666	20%	2,20	114	2,38	Stoc suficient
12	Ceapa	Legume	4	230	Megaimage	920	20%	4,20	56	4,76	Stoc insuficient
13	Conopida	Legume	12	121	Megaimage	1452	20%	12,20	100	14,28	Stoc insuficient
14	Oua	Lactate	5	420	Megaimage	2100	20%	5,20	235	5,95	Stoc suficient

## E.Determinare discount

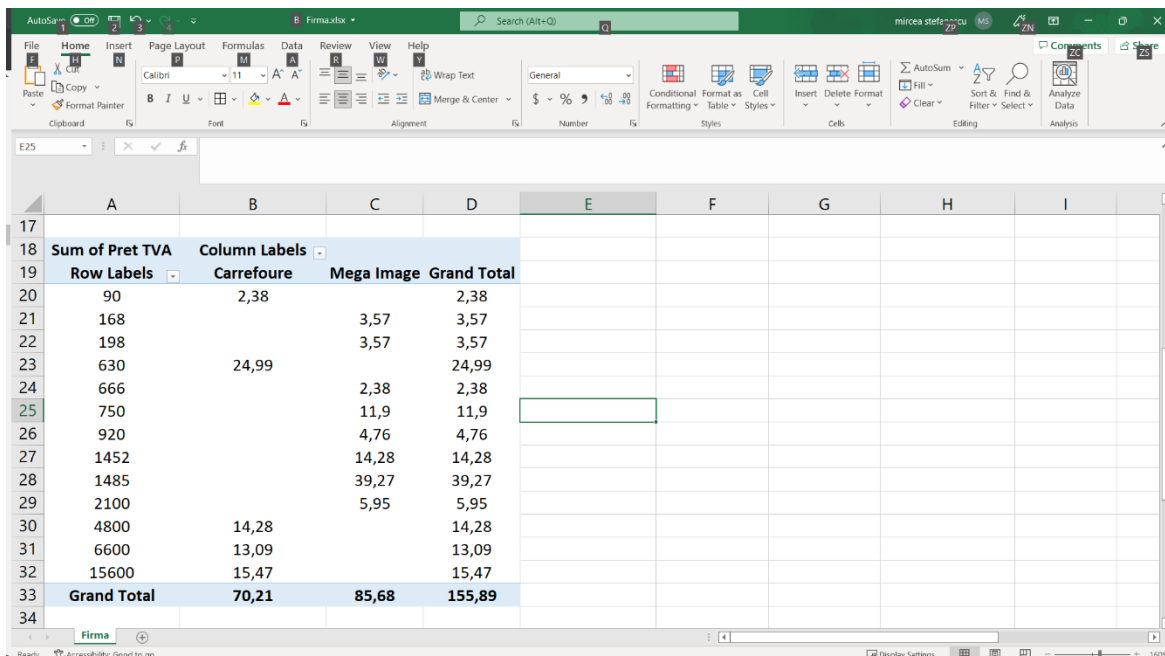
Valoarea este determinata de o functie creata de utilizator; astfel daca pretul cu TVA este mai mare de 20, se aplica un discount de 10 %.



	B	C	D	E	F	G	H	I	J	K	L	M
	Produce Disponibil	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc	Adaus Comercial	Pret Vanzare	Unitati Vandute	Pret TVA	Stare stoc	Discount
1	Miere	Fructe	12	400	Carrefour	4800	20%	12,20	325	14,28	Stoc suficient	0
2	Pere	Fructe	11	600	Carrefour	6600	20%	11,20	450	13,09	Stoc suficient	0
3	Gutui	Fructe	13	1200	Carrefour	15600	20%	13,20	845	15,47	Stoc suficient	0
4	Ciocolata	Dulciuri	21	30	Carrefour	630	20%	21,20	25	24,99	Stoc insuficient	K5
5	Paine	Fainoase	2	45	Carrefour	90	20%	2,20	36	2,38	Stoc insuficient	0
6	Seminte	Cereale	3	66	Megaimage	198	20%	3,20	42	3,57	Stoc insuficient	0
7	Lapte	Lactate	10	75	Megaimage	750	20%	10,20	30	11,9	Stoc insuficient	0
8	Miere	Dulciuri	33	45	Megaimage	1485	20%	33,20	40	39,27	Stoc insuficient	3,93
9	Varza	Legume	3	56	Megaimage	168	20%	3,20	34	3,57	Stoc insuficient	0
10	Cartofi	Legume	2	333	Megaimage	666	20%	2,20	114	2,38	Stoc suficient	0
11	Ceapa	Legume	4	230	Megaimage	920	20%	4,20	56	4,76	Stoc insuficient	0
12	Conopida	Legume	12	121	Megaimage	1452	20%	12,20	100	14,28	Stoc insuficient	0
13	Oua	Lactate	5	420	Megaimage	2100	20%	5,20	235	5,95	Stoc suficient	0

## F.Raportul stocurilor

In tabelul de mai jos se regaseste un raport ce contine atat valoarea stocurilor pentru fiecare produs dar si pretul de vanzare cat si furnizorii acelor produse



	A	B	C	D	E	F	G	H	I
17									
18	Sum of Pret TVA	Column Labels							
19	Row Labels	Carrefour	Mega Image	Grand Total					
20	90	2,38		2,38					
21	168		3,57	3,57					
22	198		3,57	3,57					
23	630	24,99		24,99					
24	666		2,38	2,38					
25	750		11,9	11,9					
26	920		4,76	4,76					
27	1452		14,28	14,28					
28	1485		39,27	39,27					
29	2100		5,95	5,95					
30	4800	14,28		14,28					
31	6600	13,09		13,09					
32	15600	15,47		15,47					
33	Grand Total	70,21	85,68	155,89					

## 2.Prelucrari SAS Enterprise Guide:

- importul unui fisier non-SAS;
- interogări;
- joncțiune;
- folosirea parametrilor;
- rapoarte;
- grafice;
- prelucrări statistice;
- formate definite de utilizator;
- crearea unui document compus;
- stiluri create de utilizator.

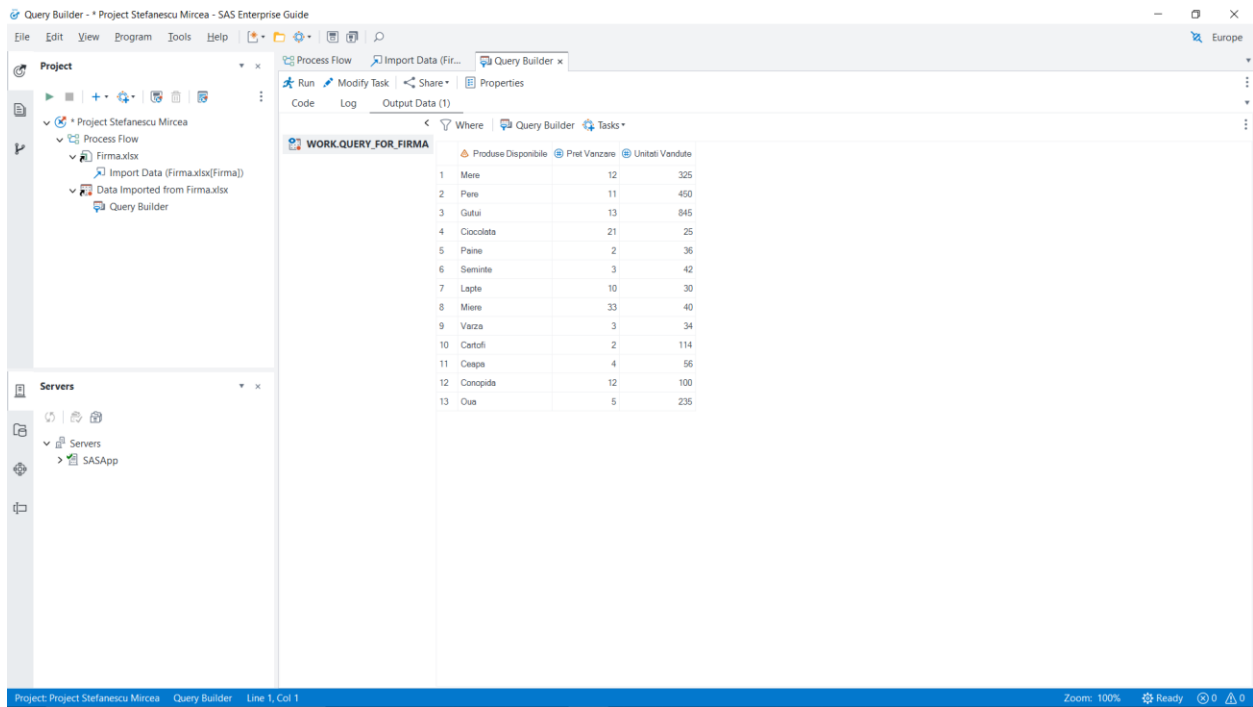
### ● Importul fisierului “Firma.xml”

Project: Project Stefanescu Mircea - Import Data (Firma.xlsx[Firma])

Zoom: 100% Ready

	Categoria	Pret Achizitie	Stoc	Furnizor	Valoare Stoc	Adus Comercial	Pret Vanzare	Unitati Vandute	Pret TVA	Stare stoc	Discount
1	Fructe	12	400	Carefour	4800	20%	12	325	14.28	Stoc suficient	
2	Fructe	11	600	Carefour	6600	20%	11	450	13.09	Stoc suficient	
3	Fructe	13	1200	Carefour	15600	20%	13	845	15.47	Stoc suficient	
4	Dulciuri	21	30	Carefour	630	20%	21	25	24.99	Stoc insuficient	
5	Fainoase	2	45	Carefour	90	20%	2	36	2.38	Stoc insuficient	
6	Cereale	3	66	Megamimage	198	20%	3	42	3.57	Stoc insuficient	
7	Lactate	10	75	Megamimage	750	20%	10	30	11.9	Stoc insuficient	
8	Dulciuri	33	45	Megamimage	1485	20%	33	40	39.27	Stoc insuficient	
9	Legume	3	56	Megamimage	168	20%	3	34	3.57	Stoc insuficient	
10	Legume	2	333	Megamimage	666	20%	2	114	2.38	Stoc suficient	
11	Legume	4	230	Megamimage	920	20%	4	56	4.76	Stoc insuficient	
12	Legume	12	121	Megamimage	1452	20%	12	100	14.28	Stoc insuficient	
13	Lactate	5	420	Megamimage	2100	20%	5	235	5.95	Stoc suficient	

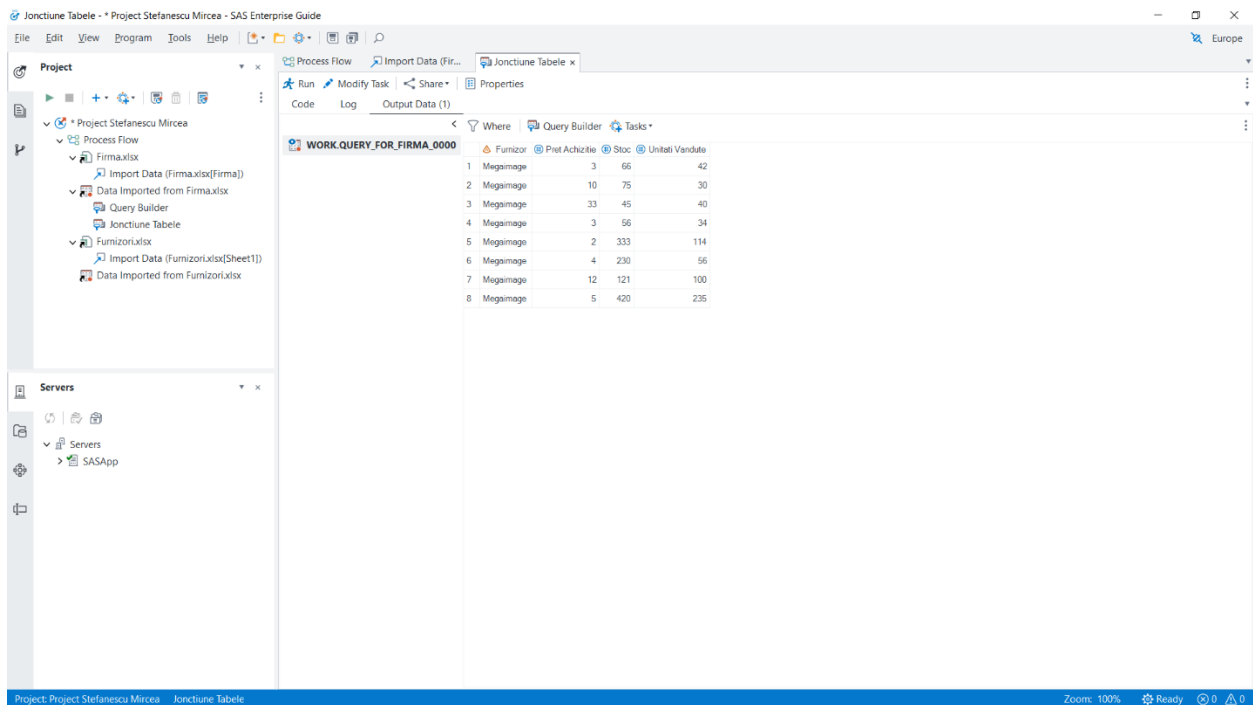
- **Interogari:** O prima interogare folosind Query Builder a grupat produsele disponibile, pretul de vanzare si numarul de unitati vandute si am obtinut tabelul:



The screenshot shows the SAS Enterprise Guide interface with the Query Builder window active. The query is named 'WORK.QUERY\_FOR\_FIRMA'. The result table displays 13 rows of data with columns for product name, price, and units sold.

	Produse Disponibile	Pret Vanzare	Unitati Vandute
1	Mere	12	325
2	Pere	11	450
3	Gutui	13	845
4	Ciocolata	21	25
5	Paine	2	36
6	Semințe	3	42
7	Lapte	10	30
8	Miere	33	40
9	Varza	3	34
10	Cartofi	2	114
11	Ceapa	4	56
12	Conopida	12	100
13	Oua	5	235

- **Jonctiune:** am realizat jonctiunea interna folosind primul tabel “Firma” si un nou tabel introdus si anume lista furnizorilor firmei.



The screenshot shows the SAS Enterprise Guide interface with the Query Builder window active. The query is named 'WORK.QUERY\_FOR\_FIRMA\_0000'. The result table displays 8 rows of data, representing an internal join between the 'Firma' table and a new table 'Furnizori'.

	Furnizor	Pret Achizitie	Stoc	Unitati Vandute
1	Megaimage	3	66	42
2	Megaimage	10	75	30
3	Megaimage	33	45	40
4	Megaimage	3	56	34
5	Megaimage	2	333	114
6	Megaimage	4	230	56
7	Megaimage	12	121	100
8	Megaimage	5	420	235



- **Folosirea parametrilor:** in acesta intergoare am creat o tabela care sa afiseze furnizorul, pretul de achizitie, stocul si unitatile vandute doar pentru produsele care au inregistrat vanzari de peste 50 unitati.

Furnizor	Pret Achizitie	Stoc	Unitati Vandute
Megaimage	2	333	114
Megaimage	4	230	56
Megaimage	12	121	100
Megaimage	5	420	235

Generated by SAS (SASApp, Linux) on April 14, 2022 at 01:17:09 PM

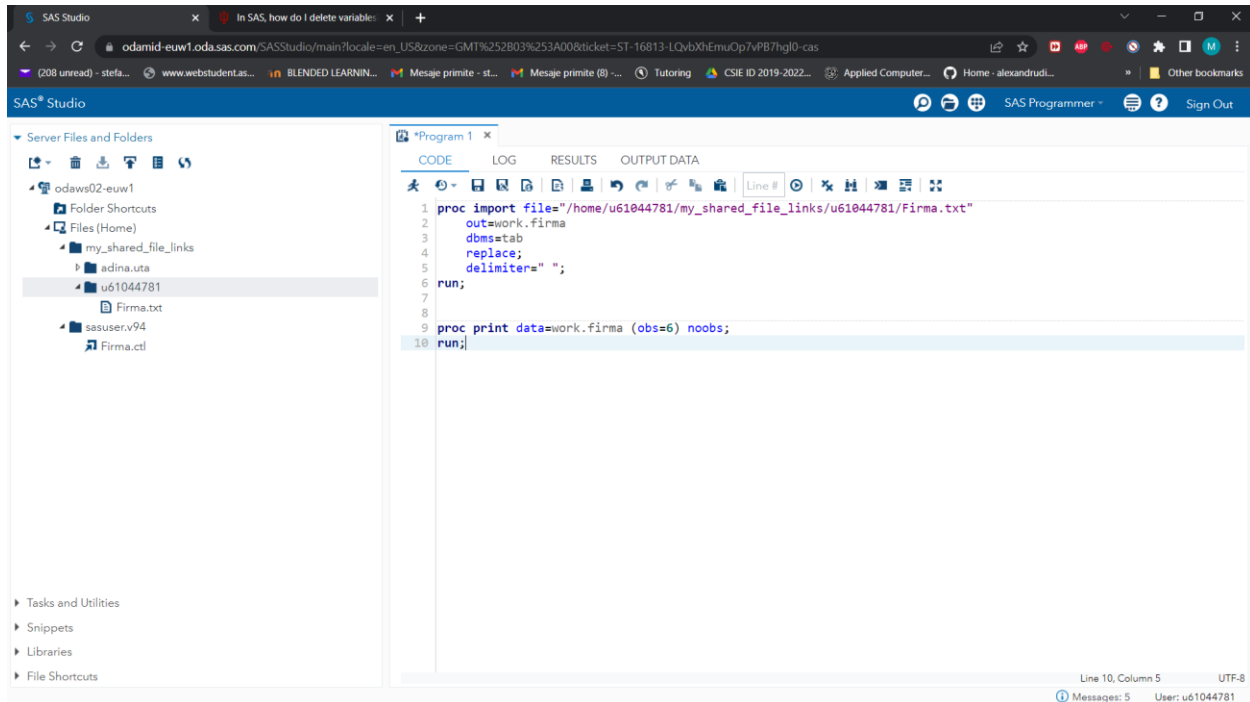
- **Rapoarte:** acest raport cuprinde informatii legate de produsele disponibile, stocul, adausul comercial si pretul cu tva

Produse Disponibile	Stoc	Adaus Comercial	Pret TVA
Mere	400	20%	14.28
Pere	600	20%	13.09
Gutui	1200	20%	15.47
Ciocolata	30	20%	24.99
Paine	45	20%	2.38
Seminte	66	20%	3.57
Lapte	75	20%	11.9
Miere	45	20%	39.27
Varza	56	20%	3.57
Cartofi	333	20%	2.38
Ceaapa	230	20%	4.76
Conopida	121	20%	14.28
Oua	420	20%	5.95

Generated by SAS (SASApp, Linux) on April 14, 2022 at 01:23:29 PM

### 3. Programare SAS:

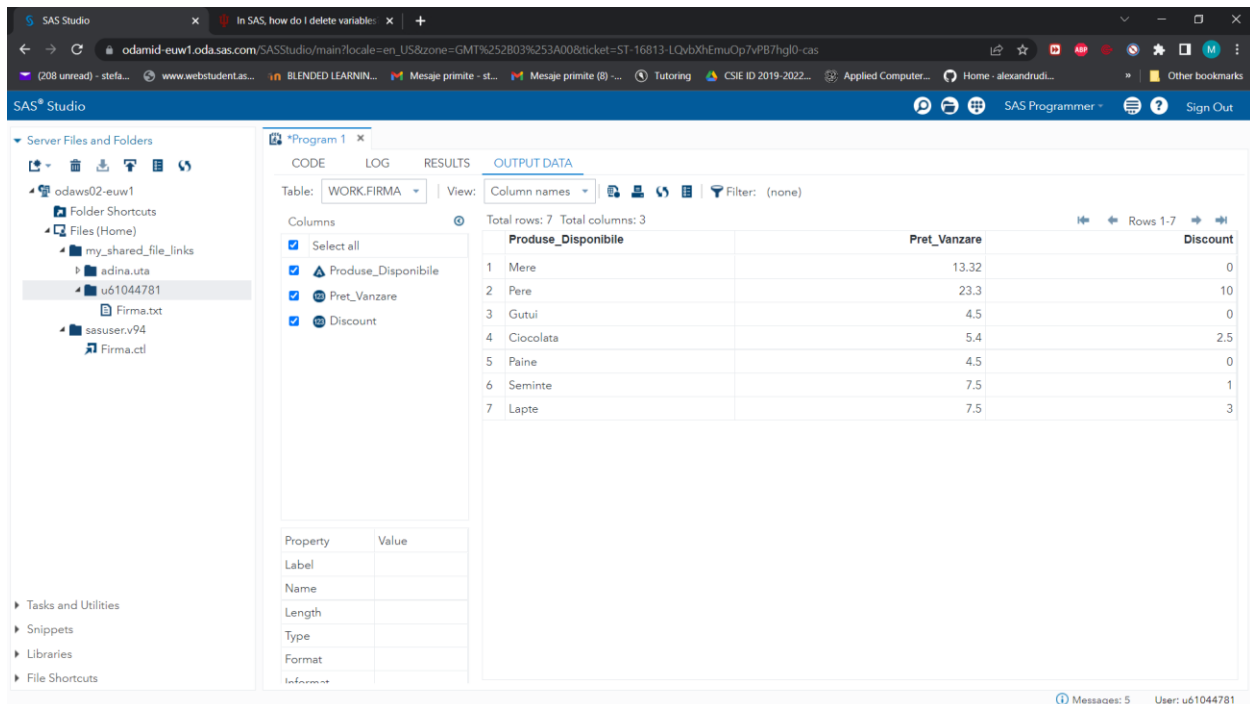
a) **crearea unui set de date SAS din fișiere externe:** am importat fisierul Firma.txt in spatiul de foldere din SAS si am rulat run pentru a datele ce se afla in fisier.



The screenshot shows the SAS Studio interface. On the left, the 'Server Files and Folders' pane displays a tree structure with folders like 'odaws02-euw1', 'Folder Shortcuts', 'Files (Home)', 'my\_shared\_file\_links', 'adina.uta', 'u61044781', 'Firma.txt', 'sasuser.v94', and 'Firma.ctf'. The main editor area shows the 'CODE' tab with the following SAS code:

```
1 proc import file="/home/u61044781/my_shared_file_links/u61044781/Firma.txt"
2 out=work.firma
3 dbms=tab
4 replace;
5 delimiter=" ";
6 run;
7
8
9
10 proc print data=work.firma (obs=6) noobs;
11 run;
```

The status bar at the bottom indicates 'Line 10, Column 5' and 'UTF-8'.



The screenshot shows the SAS Studio interface with the 'OUTPUT DATA' tab selected. The table 'WORK.FIRMA' is displayed with 7 rows and 3 columns. The columns are 'Produce\_Disponibile', 'Pret\_Vanzare', and 'Discount'. The data is as follows:

Produce_Disponibile	Pret_Vanzare	Discount
1 Mere	13.32	0
2 Pere	23.3	10
3 Gutui	4.5	0
4 Ciocolata	5.4	2.5
5 Paine	4.5	0
6 Seminte	7.5	1
7 Lapte	7.5	3

The status bar at the bottom indicates 'Messages: 5' and 'User: u61044781'.

b) **procesarea iterativă și condițională a datelor:** in cadrul procesarii conditionate afisez doar produsele ale caror preturi de vanzare este mai mare de 8;

The screenshot shows the SAS Studio interface with the 'RESULTS' tab selected. The title of the results window is 'Produse cu pretul de vanzare peste 8'. The table contains the following data:

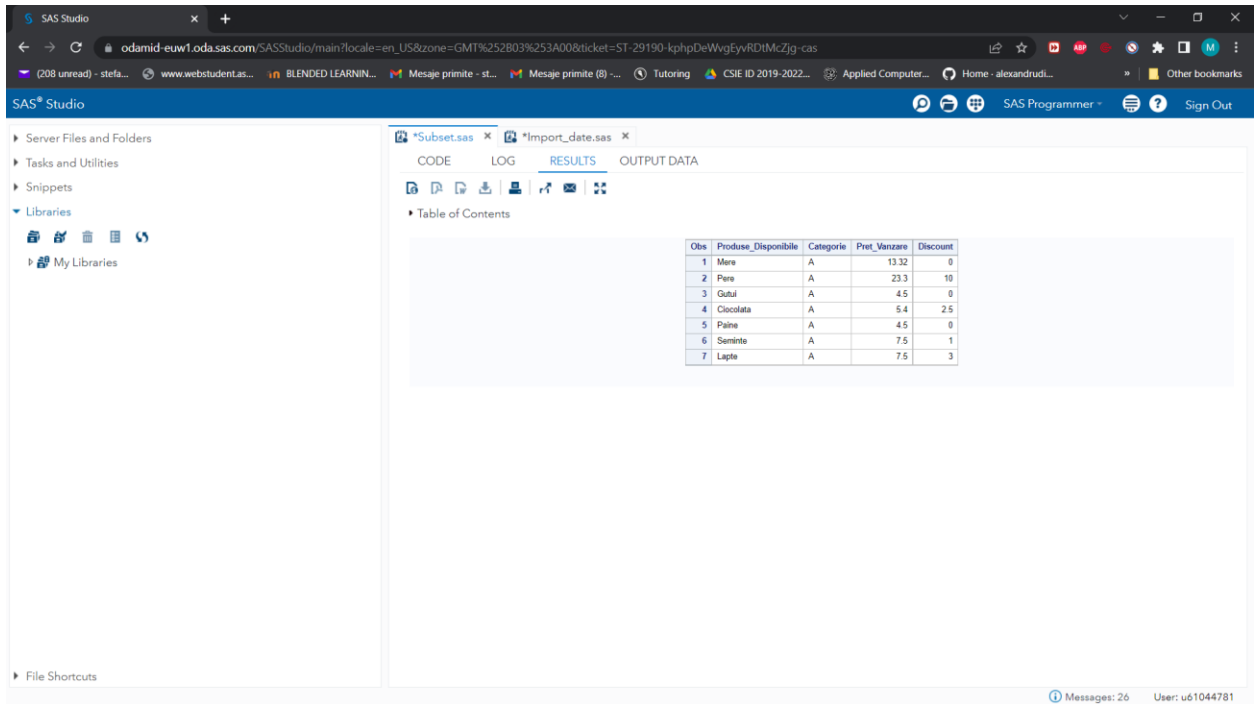
Obs	Produse_Disponibile	Pret_Vanzare	Discount
1	Mere	13.32	0
2	Pere	23.3	10

In cazul iterarii am folosit clauza DO in care am marit valoarea discountului cu 45 % daca pretul de vanzare este mai mic de 5 altfel dicountul a fost marit cu 15 % daca pretul de vanzare este mai mare de 5;

The screenshot shows the SAS Studio interface with the 'OUTPUT DATA' tab selected. The table contains the following data:

	Produse_Disponibile	Pret_Vanzare	Discount
1	Mere	13.32	0
2	Pere	23.3	11.5
3	Gutui	4.5	0
4	Ciocolata	5.4	2.875
5	Paine	4.5	0
6	Seminte	7.5	1.15
7	Lapte	7.5	3.45

c) **crearea de subseturi de date:** afisarea tuturor produselor ce contin la colona 'Categorie' litera 'A';

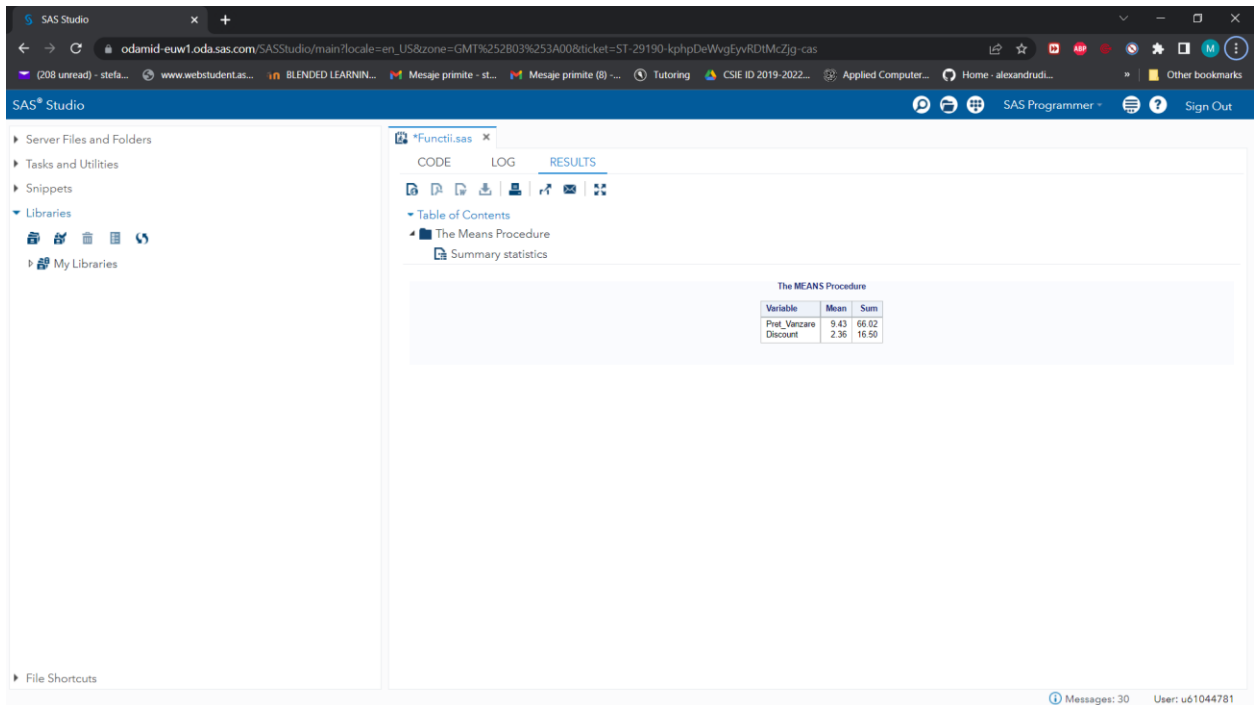


The screenshot shows the SAS Studio interface. The left sidebar contains 'Server Files and Folders', 'Tasks and Utilities', 'Snippets', 'Libraries', and 'File Shortcuts'. The main workspace has tabs for '\*Subset.sas' and '\*Import\_date.sas'. The 'RESULTS' tab is selected, showing a 'Table of Contents' with 'Table of Contents' and 'Table of Contents'. Below this is a table of results:

Obs	Produse_Disponibile	Categorie	Pret_Vanzare	Discount
1	Mere	A	13.32	0
2	Pere	A	23.3	10
3	Gutui	A	4.5	0
4	Ciocolata	A	5.4	2.5
5	Paine	A	4.5	0
6	Seminte	A	7.5	1
7	Lapte	A	7.5	3

The bottom status bar shows 'Messages: 26' and 'User: u61044781'.

d) **utilizarea de functii SAS:** mean si sum



The screenshot shows the SAS Studio interface. The left sidebar is the same as in the previous screenshot. The main workspace has a tab for '\*Funcții.sas'. The 'RESULTS' tab is selected, showing a 'Table of Contents' with 'Table of Contents', 'The Means Procedure', and 'Summary statistics'. Below this is a table titled 'The MEANS Procedure':

Variable	Mean	Sum
Pret_Vanzare	9.43	66.02
Discount	2.36	16.50

The bottom status bar shows 'Messages: 30' and 'User: u61044781'.

- e) **utilizarea de masive:** in exemplu de mai jos am definit o matrice de 2x3 in care am calculat suma elemntelor de pe randuri, media si valoarea minima;

SAS Studio interface showing the results of a matrix calculation. The 'RESULTS' tab is active, displaying a table with columns: Obs, A1, A2, A3, A4, A\_SUM, A\_MEAN, A\_MIN. The data is as follows:

Obs	A1	A2	A3	A4	A_SUM	A_MEAN	A_MIN
1	21	4	52	11	88	22.00	4
2	96	25	42	6	169	42.25	6

- f) **utilizarea de proceduri pentru raportare:** afisez intr-un raport toate coloanele din tablele "Firma2.txt".

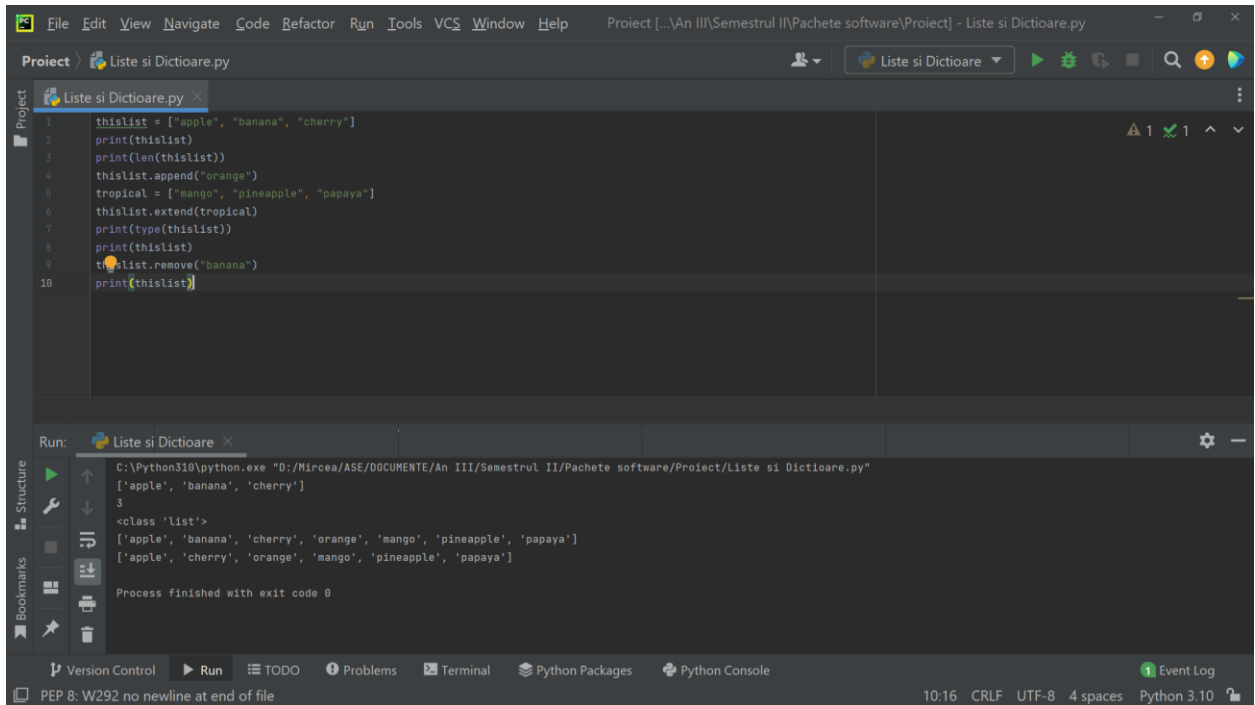
SAS Studio interface showing the results of a report generation. The 'RESULTS' tab is active, displaying a table titled 'Raport Preturi' with columns: Obs, Produse\_Disponibile, Categorie, Pret\_Vanzare, Discount. The data is as follows:

Obs	Produse_Disponibile	Categorie	Pret_Vanzare	Discount
1	Mere	A	13.32	0
2	Pere	A	23.3	10
3	Gutui	A	4.5	0
4	Ciocolata	A	5.4	2.5
5	Paine	A	4.5	0
6	Seminte	A	7.5	1
7	Lapte	A	7.5	3
8	Hartie	N	4.5	2
9	Bicicleta	N	25.45	4
10	Pantofi	N	3.45	4
11	.	.	.	.

## 4. Programare Python

### a) utilizarea listelor și a dicționarilor, incluzând metode specifice acestora;

Am declarat o lista in care am inserat mai multe valori de tip string dupa care am efectuat mai multe operatiuni asupra sa: functia 'len()', 'append()', 'extend()' si 'remove()'

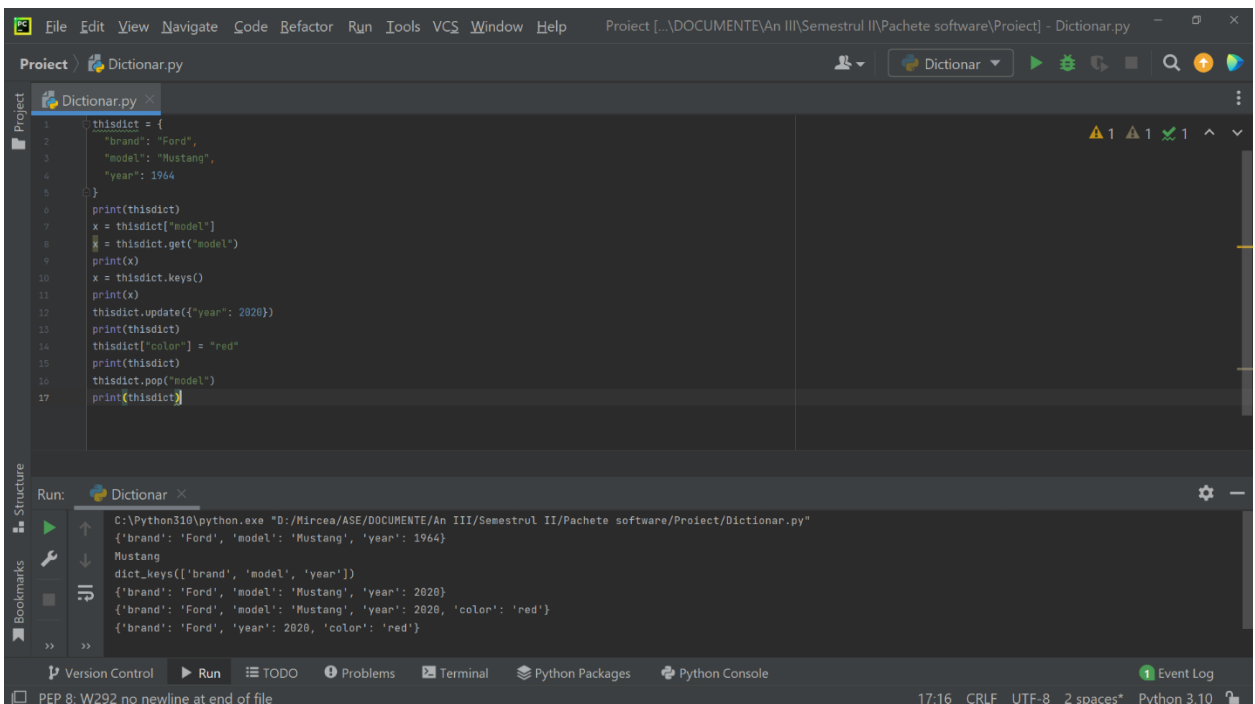


```
1 thislist = ["apple", "banana", "cherry"]
2 print(thislist)
3 print(len(thislist))
4 thislist.append("orange")
5 tropical = ["mango", "pineapple", "papaya"]
6 thislist.extend(tropical)
7 print(type(thislist))
8 print(thislist)
9 thislist.remove("banana")
10 print(thislist)
```

Run: Liste si Dictionare

```
C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Proiect/Liste si Dictionare.py"
['apple', 'banana', 'cherry']
3
<class 'list'>
['apple', 'banana', 'cherry', 'orange', 'mango', 'pineapple', 'papaya']
['apple', 'cherry', 'orange', 'mango', 'pineapple', 'papaya']
Process finished with exit code 0
```

Am declarat un dictionar de tip cheie-valoare; Am folost 'get()', 'keys()', 'update()', 'pop()' si am atribuit o valoare noua.



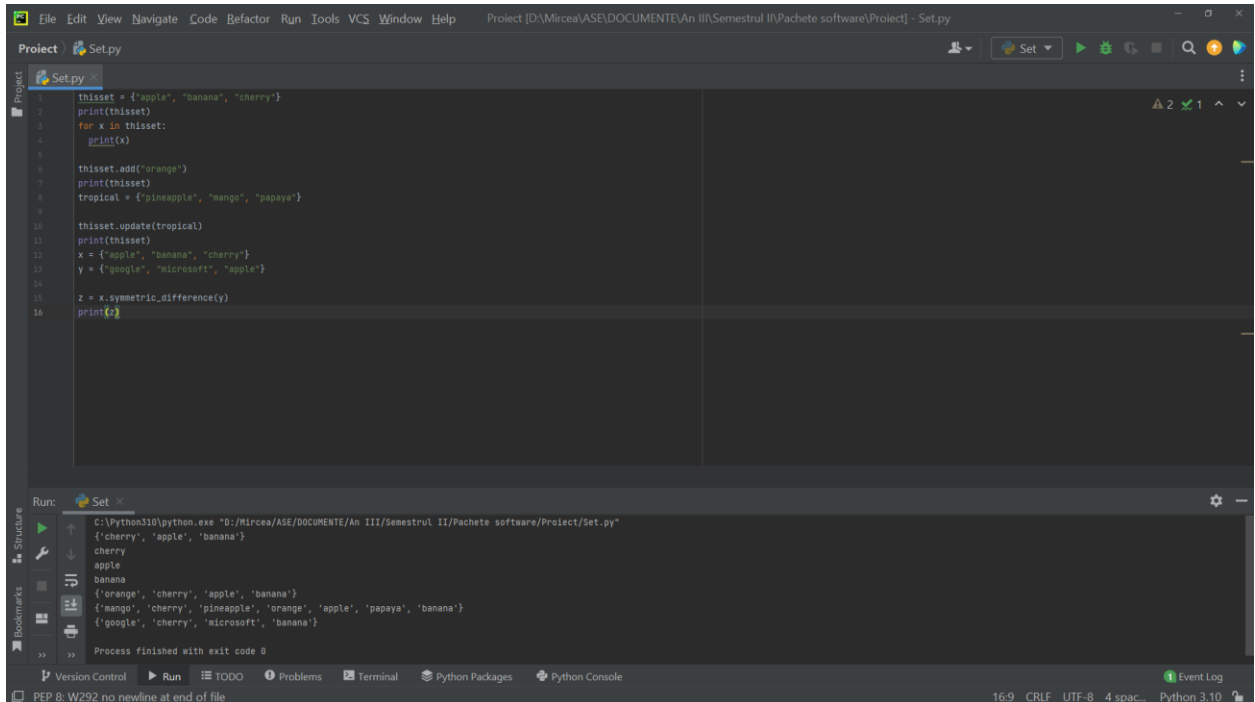
```
1 thisdict = {
2     "brand": "Ford",
3     "model": "Mustang",
4     "year": 1964
5 }
6 print(thisdict)
7 x = thisdict["model"]
8 y = thisdict.get("model")
9 print(x)
10 x = thisdict.keys()
11 print(x)
12 thisdict.update({'year': 2020})
13 print(thisdict)
14 thisdict["color"] = "red"
15 print(thisdict)
16 thisdict.pop("model")
17 print(thisdict)
```

Run: Dictionar

```
C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Proiect/Dictionar.py"
{'brand': 'Ford', 'model': 'Mustang', 'year': 1964}
Mustang
dict_keys(['brand', 'model', 'year'])
{'brand': 'Ford', 'model': 'Mustang', 'year': 2020}
{'brand': 'Ford', 'model': 'Mustang', 'year': 2020, 'color': 'red'}
{'brand': 'Ford', 'year': 2020, 'color': 'red'}
```

## b) utilizarea seturilor și a tuplurilor, incluzând metode specifice acestora:

Pentru set am printat folosind metoda ‘print()’ dar și iterând print fiecare element; De asemenea am folosit metoda ‘add()’, ‘update()’ și ‘symmetric\_difference()’



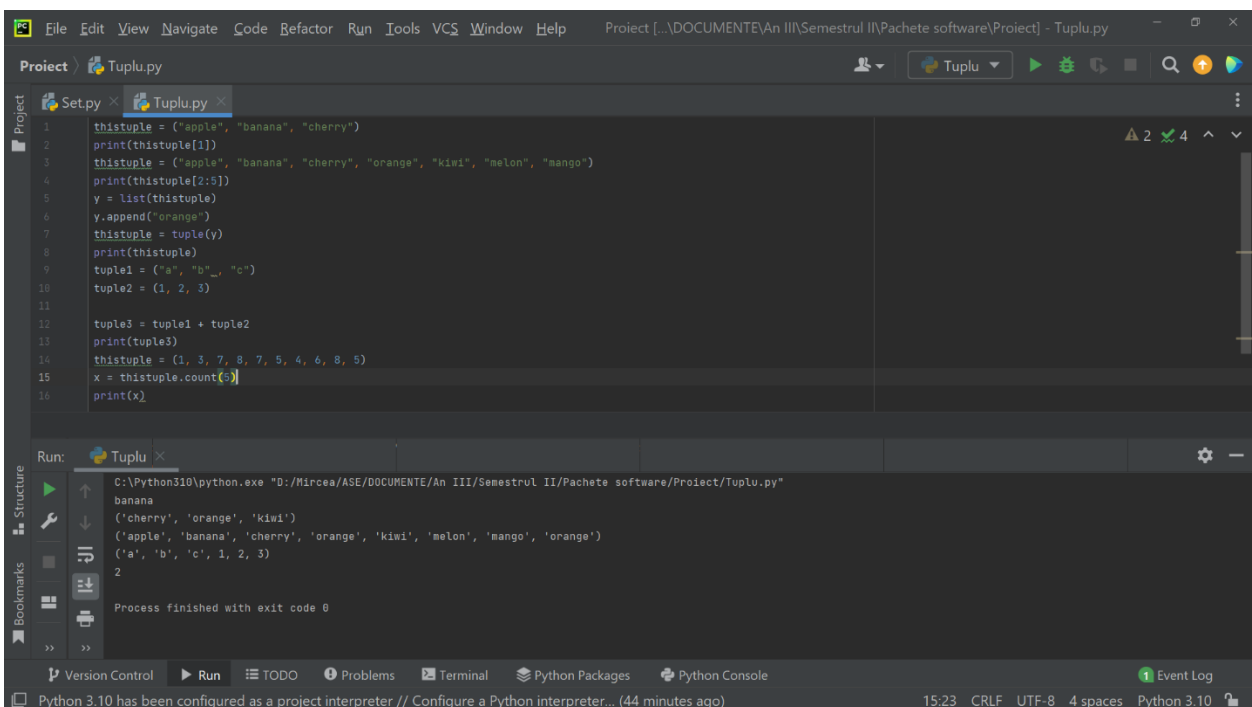
```
1 thisset = {'apple', 'banana', 'cherry'}
2 print(thisset)
3 for x in thisset:
4     print(x)
5
6 thisset.add('orange')
7 print(thisset)
8 tropical = {'pineapple', 'mango', 'papaya'}
9
10 thisset.update(tropical)
11 print(thisset)
12 x = {'apple', 'banana', 'cherry'}
13 y = {'google', 'microsoft', 'apple'}
14
15 z = x.symmetric_difference(y)
16 print(z)
```

Run: Set

```
C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Project/Set.py"
{'cherry', 'apple', 'banana'}
cherry
apple
banana
{'orange', 'cherry', 'apple', 'banana'}
{'mango', 'cherry', 'pineapple', 'orange', 'apple', 'papaya', 'banana'}
{'google', 'cherry', 'microsoft', 'banana'}
```

Process finished with exit code 0

Pentru tupluri am folosit metoda ‘append()’, ‘count()’, am cocatenat 2 tupluri, am accesat un element pe baza indexului sau mai multe elemente pe baza unui interval.



```
1 thistuple = ('apple', 'banana', 'cherry')
2 print(thistuple[1])
3 thistuple = ('apple', 'banana', 'cherry', 'orange', 'kiwi', 'melon', 'mango')
4 print(thistuple[2:5])
5 y = list(thistuple)
6 y.append('orange')
7 thistuple = tuple(y)
8 print(thistuple)
9 tuple1 = ('a', 'b', 'c')
10 tuple2 = (1, 2, 3)
11
12 tuple3 = tuple1 + tuple2
13 print(tuple3)
14 thistuple = (1, 3, 7, 8, 7, 5, 4, 6, 8, 9)
15 x = thistuple.count(7)
16 print(x)
```

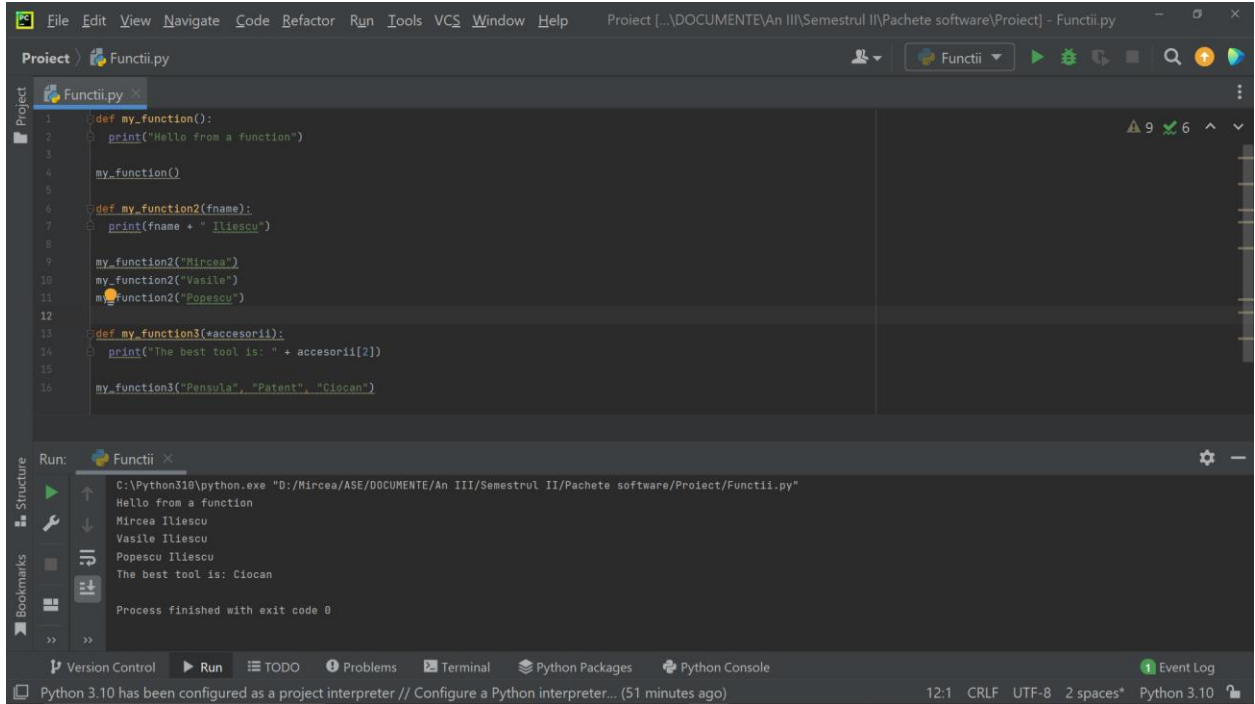
Run: Tuplu

```
C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Project/Tuplu.py"
banana
('cherry', 'orange', 'kiwi')
('apple', 'banana', 'cherry', 'orange', 'kiwi', 'melon', 'mango', 'orange')
('a', 'b', 'c', 1, 2, 3)
2
```

Process finished with exit code 0

### c) definirea și apelarea unor funcții:

Am definit si apelat 3 functii: prima fara parametrii, a doua cu un parametru iar a 3 a cu numar variat de argumente.



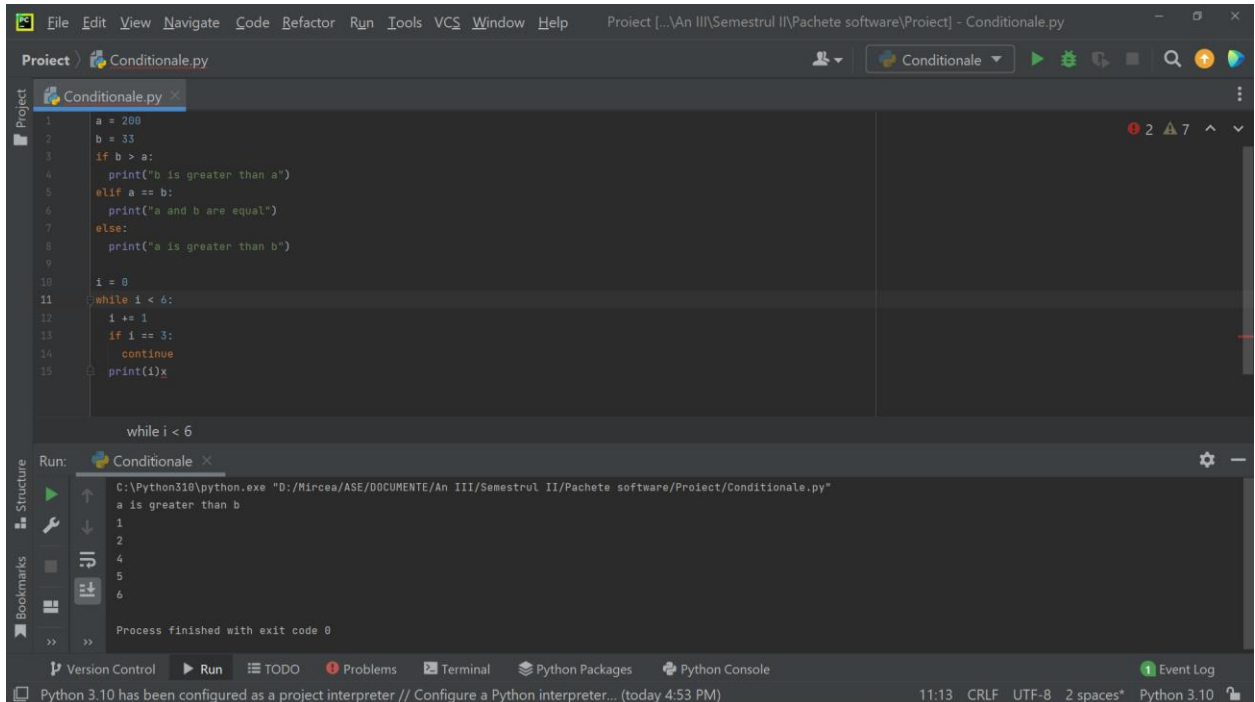
```
1 def my_function():
2     print("Hello from a function")
3
4     my_function()
5
6 def my_function2(fname):
7     print(fname + " Iliescu")
8
9     my_function2("Mircea")
10    my_function2("Vasile")
11    my_function2("Popescu")
12
13 def my_function3(*accesorii):
14     print("The best tool is: " + accesorii[2])
15
16     my_function3("Pensula", "Patent", "Ciocan")
```

Run: Funcții

```
C:\Python310\python.exe "D:\Mircea\ASE\DOCUMENTE\An III\Semestrul II\Pachete software\Project\Functii.py"
Hello from a function
Mircea Iliescu
Vasile Iliescu
Popescu Iliescu
The best tool is: Ciocan

Process finished with exit code 0
```

### d) utilizarea structurilor condiționale: if-else si while



```
1 a = 200
2 b = 33
3 if b > a:
4     print("b is greater than a")
5 elif a == b:
6     print("a and b are equal")
7 else:
8     print("a is greater than b")
9
10 i = 0
11 while i < 6:
12     i += 1
13     if i == 3:
14         continue
15     print(i)
```

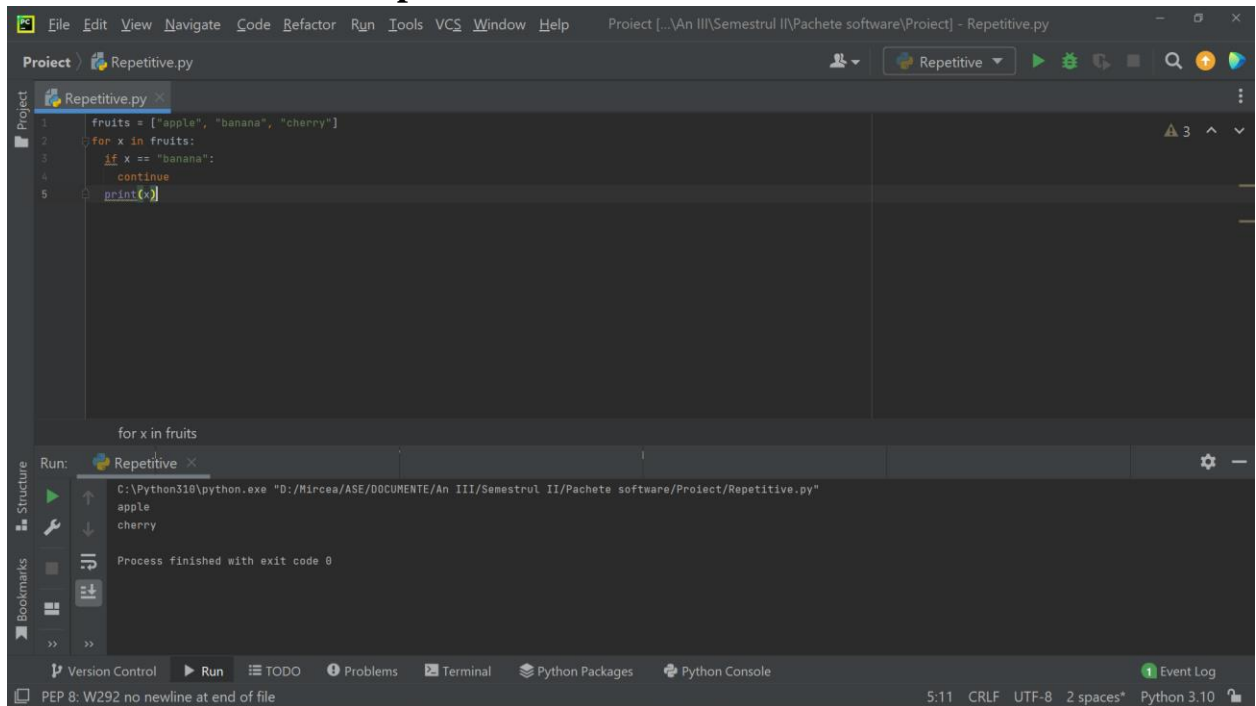
Run: Conditionale

```
C:\Python310\python.exe "D:\Mircea\ASE\DOCUMENTE\An III\Semestrul II\Pachete software\Project\Conditionale.py"
a is greater than b
1
2
4
5
6

Process finished with exit code 0
```



**e) utilizarea structurilor repetitive: for**



```
1 fruits = ["apple", "banana", "cherry"]
2 for x in fruits:
3     if x == "banana":
4         continue
5     print(x)
```

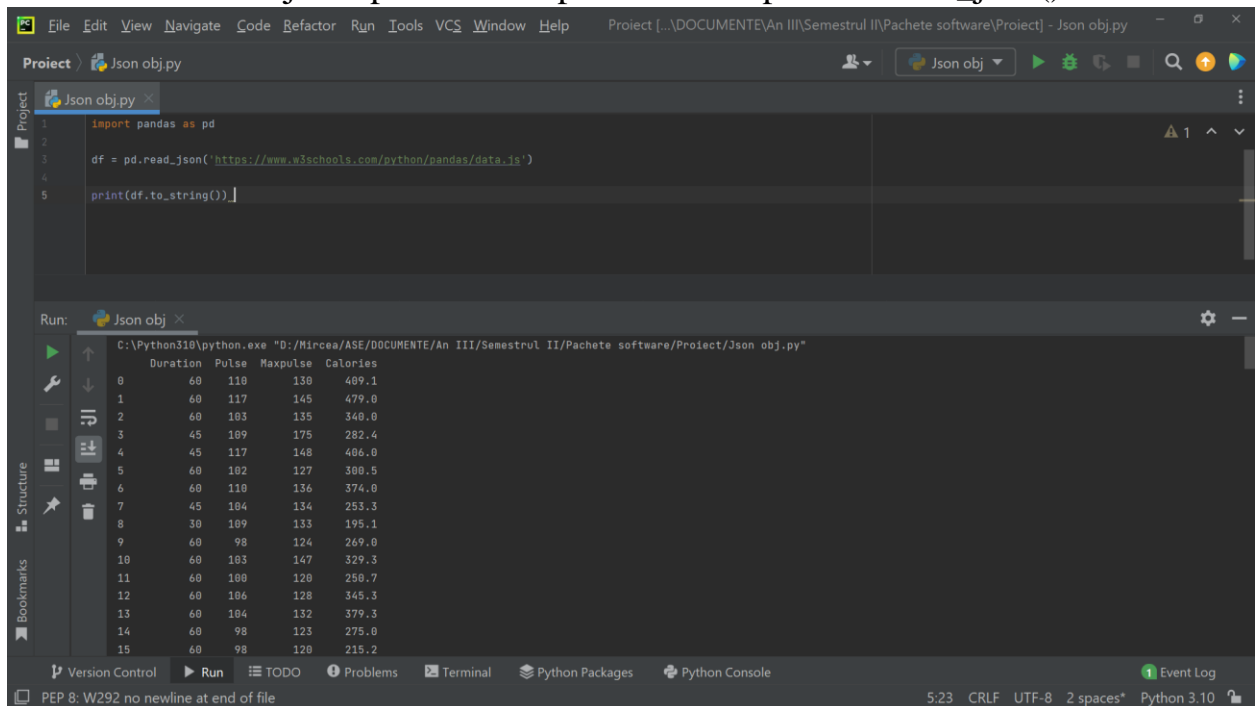
Run: Repetitive

C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Project/Repetitive.py"

apple  
cherry

Process finished with exit code 0

**f) importul unei fisier csv sau json în pachetul pandas: am folosit un fisier JSON numit 'data.json' pe care am aplicat functia pandas "read\_json()"**



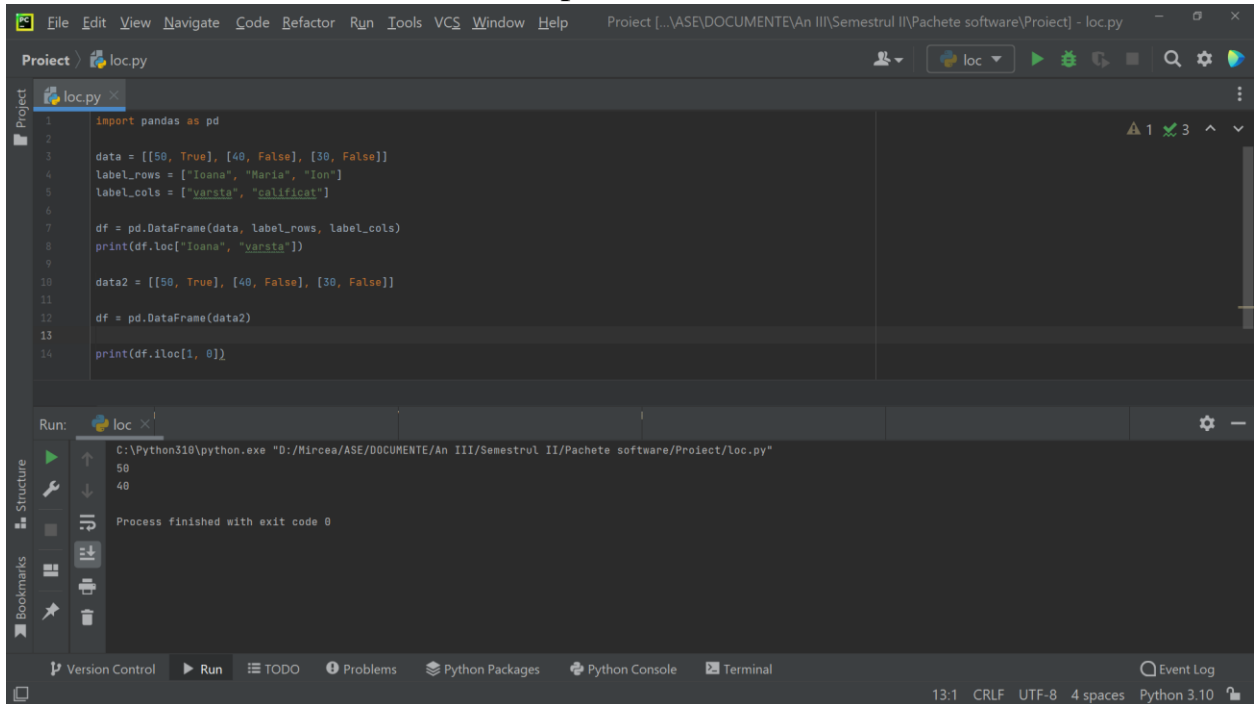
```
1 import pandas as pd
2
3 df = pd.read_json('https://www.w3schools.com/python/pandas/data.js')
4
5 print(df.to_string())
```

Run: Json obj

C:\Python310\python.exe "D:/Mircea/ASE/DOCUMENTE/An III/Semestrul II/Pachete software/Project/Json obj.py"

	Duration	Pulse	Maxpulse	Calories
0	60	110	130	409.1
1	60	117	145	479.0
2	60	103	135	340.0
3	45	109	175	282.4
4	45	117	148	406.0
5	60	102	127	300.5
6	60	110	136	374.0
7	45	104	134	253.3
8	30	109	133	195.1
9	60	98	124	269.0
10	60	103	147	329.3
11	60	100	120	250.7
12	60	106	128	345.3
13	60	104	132	379.3
14	60	98	123	275.0
15	60	98	120	215.2

g) **accesarea datelor cu loc și iloc:** am folosit functiile ‘loc()’ si ‘iloc’ pentru a returna continutul unei variabile de tip lista de lista



The screenshot shows an IDE window titled 'loc.py' with the following Python code:

```
1 import pandas as pd
2
3 data = [[50, True], [40, False], [30, False]]
4 label_rows = ["Ioana", "Maria", "Ion"]
5 label_cols = ["varsta", "calificat"]
6
7 df = pd.DataFrame(data, label_rows, label_cols)
8 print(df.loc["Ioana", "varsta"])
9
10 data2 = [[50, True], [40, False], [30, False]]
11
12 df = pd.DataFrame(data2)
13
14 print(df.iloc[1, 0])
```

Below the code editor, the 'Run' panel shows the command: `C:\Python310\python.exe "D:\Mircea\ASE\DOCUMENTE\An III\Semestrul II\Pachete software\Proiect\loc.py"`. The output displays the values `50` and `40`, followed by the message 'Process finished with exit code 0'.

The bottom status bar indicates the file encoding is UTF-8, uses 4 spaces for indentation, and is running Python 3.10.