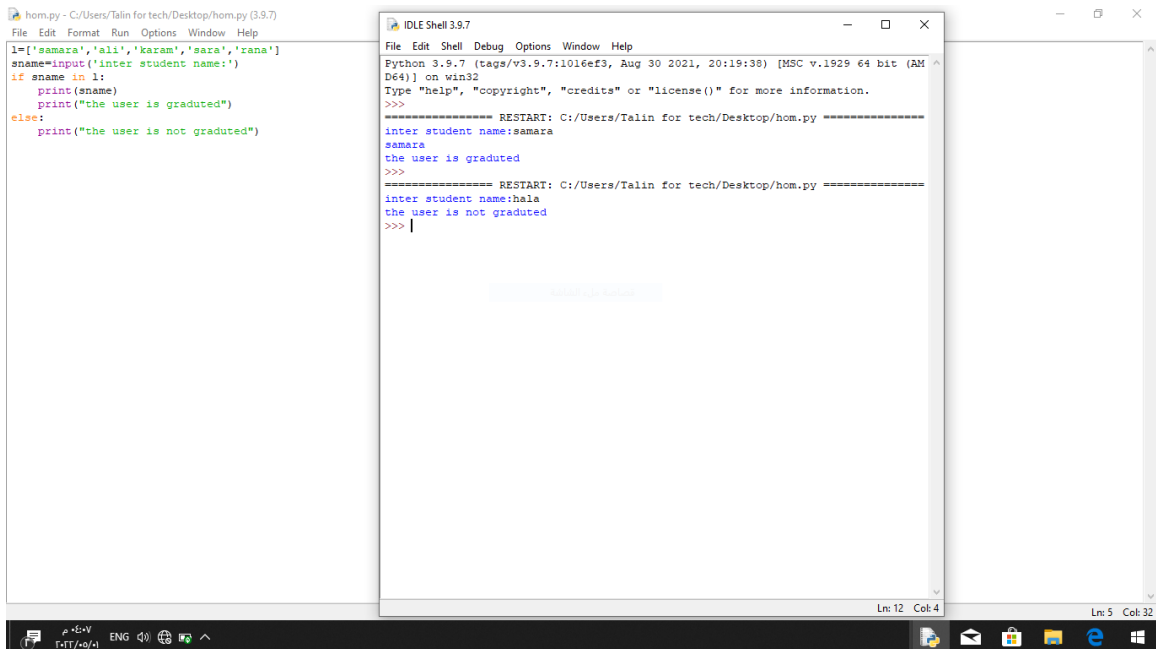


Name: samara Mansour Suleiman 2438

## First network programming homework

### Question 1:

A-



The screenshot displays a Python IDE with two windows. The left window, titled 'hom.py - C:/Users/Talin for tech/Desktop/hom.py (3.9.7)', contains the following Python code:

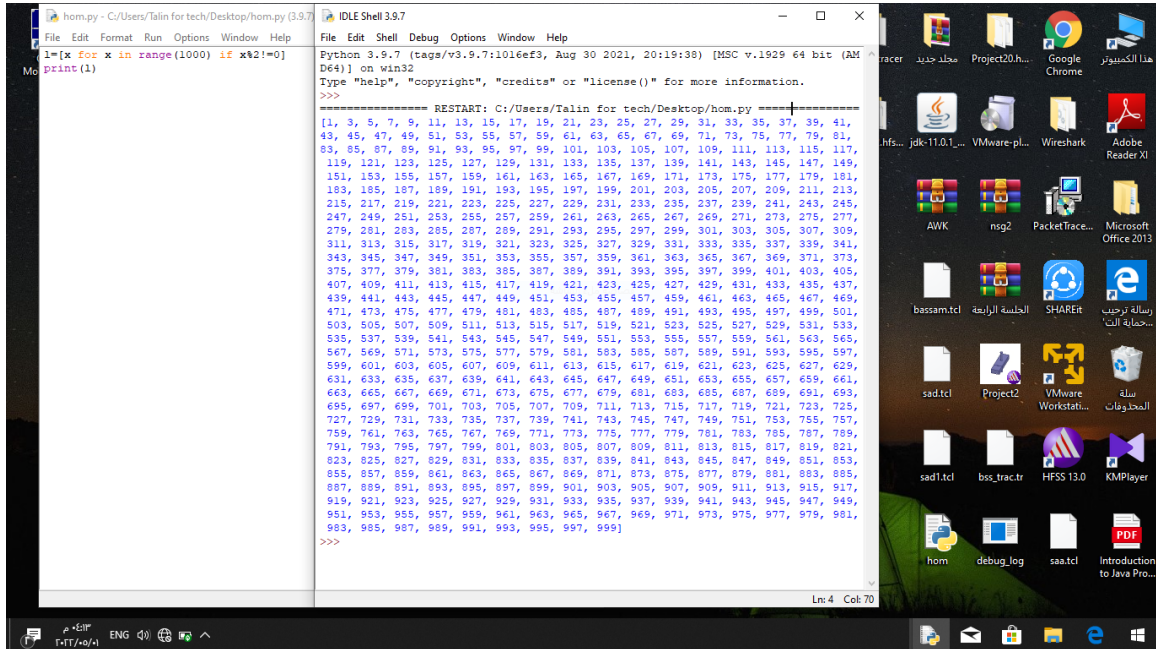
```
l=['samara','ali','karam','sara','rana']
sname=input('inter student name:')
if sname in l:
    print(sname)
    print("the user is graduted")
else:
    print("the user is not graduted")
```

The right window, titled 'IDLE Shell 3.9.7', shows the execution output:

```
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Talin for tech/Desktop/hom.py =====
inter student name:samara
samara
the user is graduted
>>>
===== RESTART: C:/Users/Talin for tech/Desktop/hom.py =====
inter student name:hala
the user is not graduted
>>> |
```

The taskbar at the bottom shows the Windows operating system with various icons and a system clock.

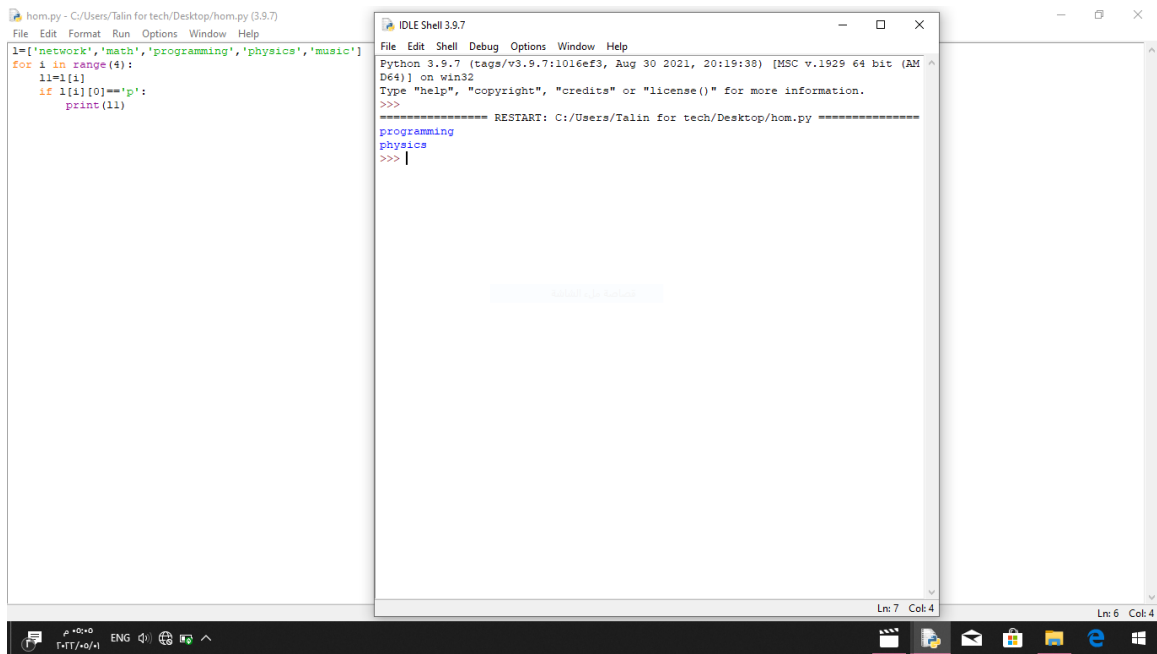
B-



The screenshot shows a Windows desktop environment. In the foreground, an IDE Shell window titled "IDLE Shell 3.9.7" is open, displaying a Python script and its output. The script is a simple loop that prints the number 1. The output shows a long list of numbers from 1 to 999, followed by a final line indicating the end of the output. The desktop background is a dark image. The taskbar at the bottom contains several icons, including the Start button, a search icon, and various application icons like Google Chrome, VMware Workstation, and Microsoft Office 2013. The system tray at the bottom right shows the date and time as 2021/8/30 20:19:38.

```
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Talin for tech/Desktop/hom.py =====
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 185, 187, 189, 191, 193, 195, 197, 199, 201, 203, 205, 207, 209, 211, 213, 215, 217, 219, 221, 223, 225, 227, 229, 231, 233, 235, 237, 239, 241, 243, 245, 247, 249, 251, 253, 255, 257, 259, 261, 263, 265, 267, 269, 271, 273, 275, 277, 279, 281, 283, 285, 287, 289, 291, 293, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325, 327, 329, 331, 333, 335, 337, 339, 341, 343, 345, 347, 349, 351, 353, 355, 357, 359, 361, 363, 365, 367, 369, 371, 373, 375, 377, 379, 381, 383, 385, 387, 389, 391, 393, 395, 397, 399, 401, 403, 405, 407, 409, 411, 413, 415, 417, 419, 421, 423, 425, 427, 429, 431, 433, 435, 437, 439, 441, 443, 445, 447, 449, 451, 453, 455, 457, 459, 461, 463, 465, 467, 469, 471, 473, 475, 477, 479, 481, 483, 485, 487, 489, 491, 493, 495, 497, 499, 501, 503, 505, 507, 509, 511, 513, 515, 517, 519, 521, 523, 525, 527, 529, 531, 533, 535, 537, 539, 541, 543, 545, 547, 549, 551, 553, 555, 557, 559, 561, 563, 565, 567, 569, 571, 573, 575, 577, 579, 581, 583, 585, 587, 589, 591, 593, 595, 597, 599, 601, 603, 605, 607, 609, 611, 613, 615, 617, 619, 621, 623, 625, 627, 629, 631, 633, 635, 637, 639, 641, 643, 645, 647, 649, 651, 653, 655, 657, 659, 661, 663, 665, 667, 669, 671, 673, 675, 677, 679, 681, 683, 685, 687, 689, 691, 693, 695, 697, 699, 701, 703, 705, 707, 709, 711, 713, 715, 717, 719, 721, 723, 725, 727, 729, 731, 733, 735, 737, 739, 741, 743, 745, 747, 749, 751, 753, 755, 757, 759, 761, 763, 765, 767, 769, 771, 773, 775, 777, 779, 781, 783, 785, 787, 789, 791, 793, 795, 797, 799, 801, 803, 805, 807, 809, 811, 813, 815, 817, 819, 821, 823, 825, 827, 829, 831, 833, 835, 837, 839, 841, 843, 845, 847, 849, 851, 853, 855, 857, 859, 861, 863, 865, 867, 869, 871, 873, 875, 877, 879, 881, 883, 885, 887, 889, 891, 893, 895, 897, 899, 901, 903, 905, 907, 909, 911, 913, 915, 917, 919, 921, 923, 925, 927, 929, 931, 933, 935, 937, 939, 941, 943, 945, 947, 949, 951, 953, 955, 957, 959, 961, 963, 965, 967, 969, 971, 973, 975, 977, 979, 981, 983, 985, 987, 989, 991, 993, 995, 997, 999]
>>>
```

C-



The image shows a screenshot of a Python IDE with two windows. The left window, titled 'hom.py - C:/Users/Talin for tech/Desktop/hom.py (3.9.7)', contains the following Python code:

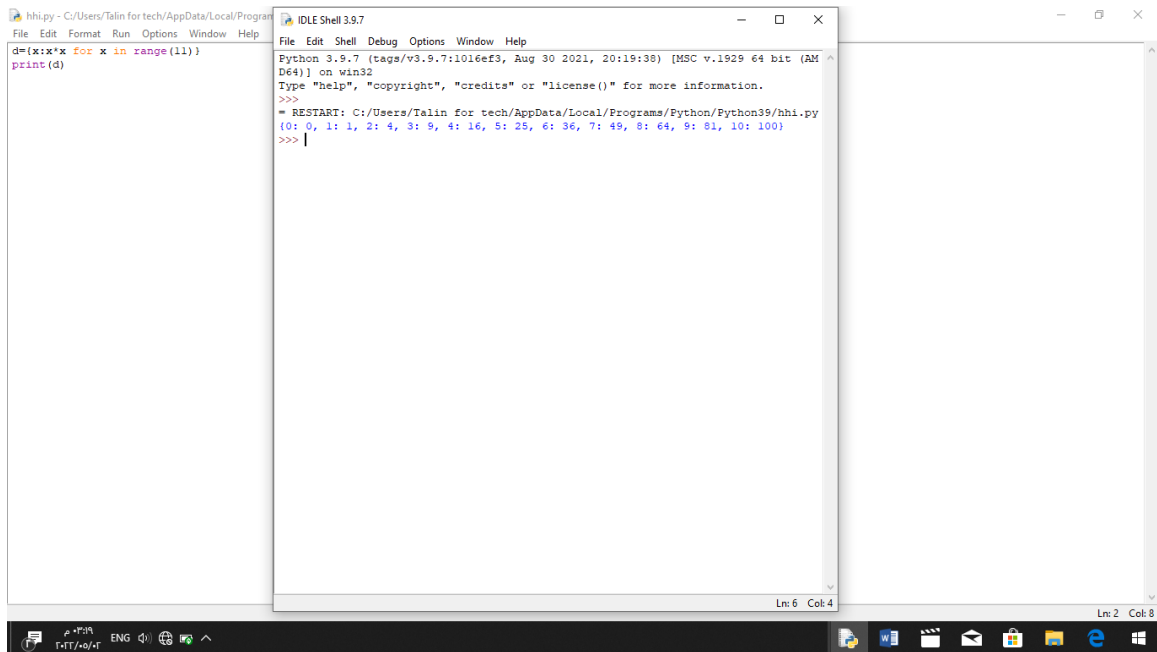
```
l=['network','math','programming','physics','music']
for i in range(4):
    ll=l[i]
    if l[i][0]=='p':
        print(ll)
```

The right window, titled 'IDLE Shell 3.9.7', shows the output of the script after execution. It displays the following text:

```
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Talin for tech/Desktop/hom.py =====
programming
physics
>>>
```

The taskbar at the bottom shows the Windows operating system with various icons and the system clock.

# D-

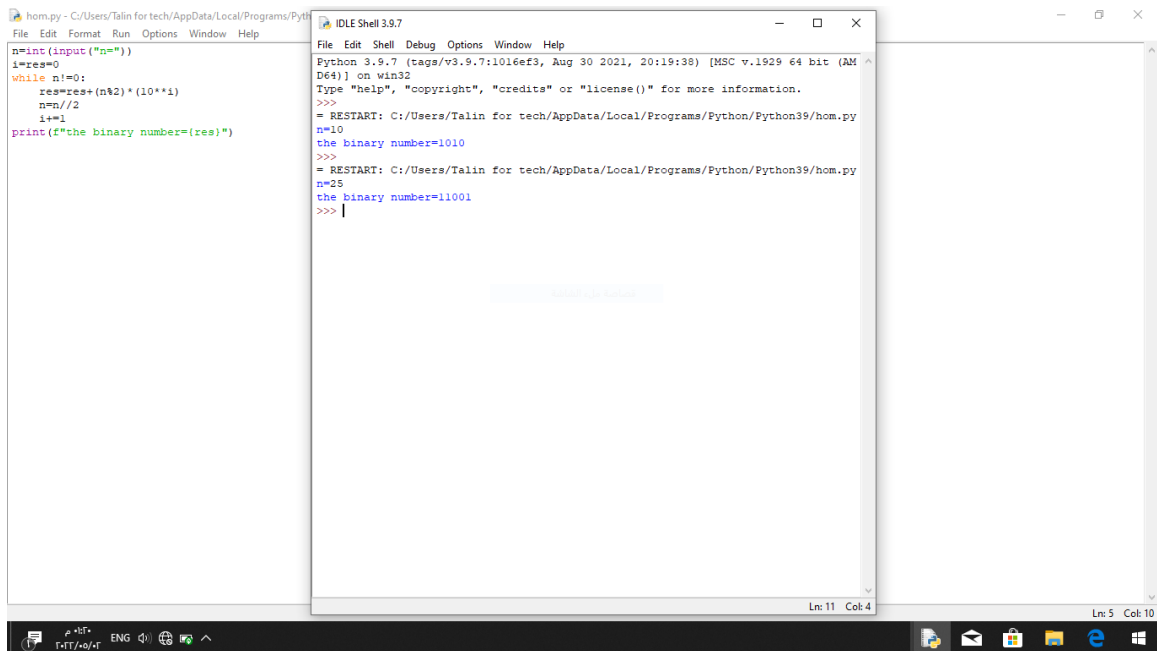


The screenshot shows the Python IDLE Shell 3.9.7 interface. The left pane displays a Python script: `d={x:x*x for x in range(11)}` followed by `print(d)`. The right pane shows the execution output, which includes the standard Python 3.9.7 startup message and a `RESTART` line indicating the execution of `hhi.py`. The output of the script is a dictionary: `{0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100}`. The status bar at the bottom indicates the cursor is at line 6, column 4.

```
hhi.py - C:/Users/Talin for tech/AppData/Local/Programs/Python/Python39/hhi.py
File Edit Format Run Options Window Help
d={x:x*x for x in range(11)}
print(d)

IDLE Shell 3.9.7
File Edit Shell Debug Options Window Help
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Talin for tech/AppData/Local/Programs/Python/Python39/hhi.py
{0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100}
>>> |
```

## Question 2:

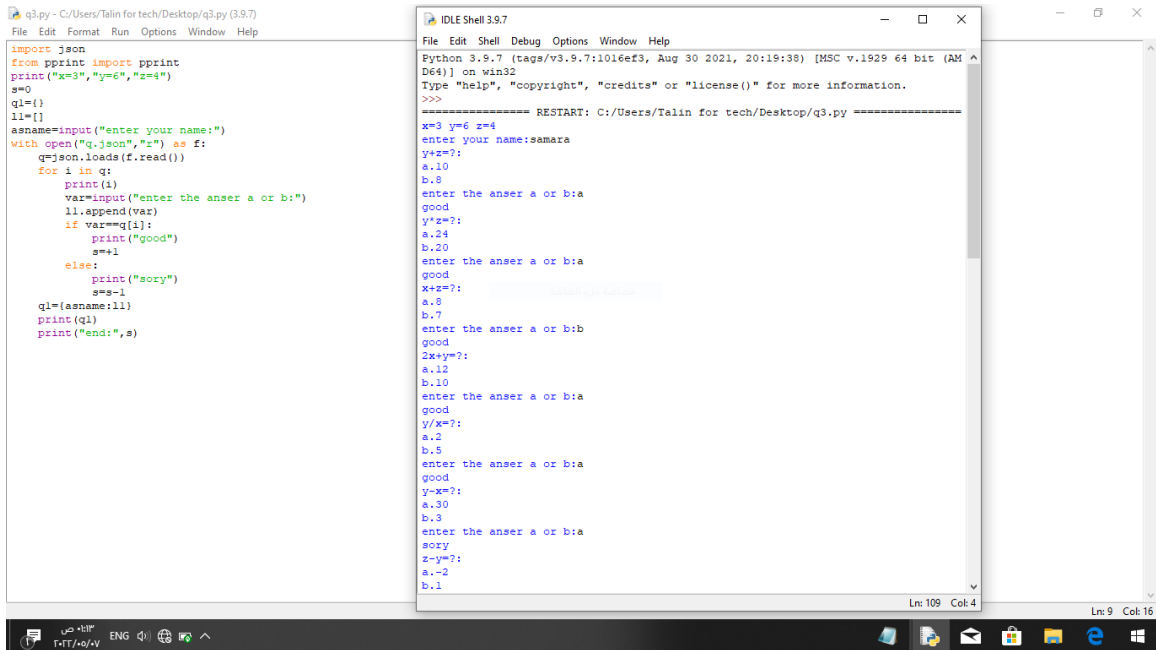


The screenshot shows the Python IDLE Shell 3.9.7 interface. The left pane displays a Python script that takes an input `n` and prints its binary representation using a loop. The right pane shows the execution output for two inputs: `n=10` resulting in `the binary number=1010` and `n=25` resulting in `the binary number=11001`. The status bar at the bottom indicates the cursor is at line 11, column 4.

```
hom.py - C:/Users/Talin for tech/AppData/Local/Programs/Python/Python39/hom.py
File Edit Format Run Options Window Help
n=int(input("n="))
i=0
while n!=0:
    res=res+(n%2)*(10**i)
    n=n//2
    i+=1
print(f"the binary number={res}")

IDLE Shell 3.9.7
File Edit Shell Debug Options Window Help
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Talin for tech/AppData/Local/Programs/Python/Python39/hom.py
n=10
the binary number=1010
>>>
= RESTART: C:/Users/Talin for tech/AppData/Local/Programs/Python/Python39/hom.py
n=25
the binary number=11001
>>> |
```

## Question 3:

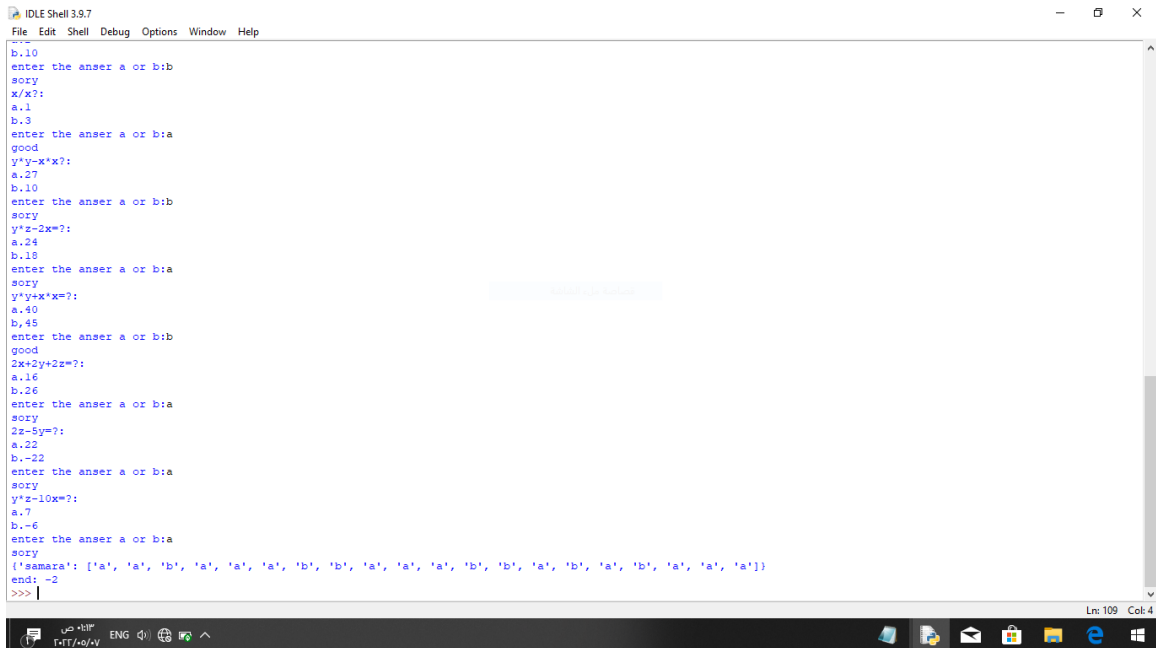


The screenshot shows a Python IDE with two windows. The left window displays a script named `q3.py` located at `C:/Users/Talin for tech/Desktop/q3.py`. The script imports `json`, `pprint`, and `pprint`. It prints initial values `x=3`, `y=6`, and `z=4`. It then prompts the user for their name, which is stored in `aname`. A loop starts, and for each iteration, it prompts the user for an answer (a or b). If the answer is correct, it prints "good" and increments a counter `s`. If the answer is incorrect, it prints "sorry" and decrements `s`. The loop continues until `s` reaches -2. The script then prints the final value of `s` and the list of answers.

```
import json
from pprint import pprint
print("x=3","y=6","z=4")
s=0
ql=[]
lname=input("enter your name:")
with open("q.json","r") as f:
    q=json.loads(f.read())
    for i in q:
        print(i)
        var=input("enter the anser a or b:")
        ll.append(var)
        if var==q[i]:
            print("good")
            s+=1
        else:
            print("sorry")
            s-=1
    ql=(aname:ll)
    print(ql)
    print("end:",s)
```

The right window shows the execution output. It displays the initial values, the user's name "samara", and the results of the loop. The final output is a dictionary containing the name and the list of answers, and the value of `s`.

```
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:18:38) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Talin for tech/Desktop/q3.py =====
x=3 y=6 z=4
enter your name:samara
y+z=?
a.10
b.8
enter the anser a or b:a
good
y+z=?
a.24
b.20
enter the anser a or b:a
good
x+z=?
a.8
b.7
enter the anser a or b:b
good
2x+y=?
a.12
b.10
enter the anser a or b:a
good
y/x=?
a.2
b.5
enter the anser a or b:a
good
y-x=?
a.30
b.3
enter the anser a or b:a
sorry
z-y=?
a.-2
b.1
```



The screenshot shows the execution output of the script in the IDE. The output displays the results of the loop, including the user's name, the list of answers, and the value of `s`.

```
b.10
enter the anser a or b:b
sorry
x/x?:
a.1
b.3
enter the anser a or b:a
good
y*y-x*x?:
a.27
b.10
enter the anser a or b:b
sorry
y*z-2x=?
a.24
b.18
enter the anser a or b:a
sorry
y*y+x*x=?
a.40
b.45
enter the anser a or b:b
good
2x+2y+2z=?
a.16
b.26
enter the anser a or b:a
sorry
2x-5y=?
a.22
b.-22
enter the anser a or b:a
sorry
y*z-10x=?
a.7
b.-6
enter the anser a or b:a
sorry
{'samara': ['a', 'a', 'b', 'a', 'a', 'a', 'b', 'b', 'a', 'a', 'b', 'b', 'a', 'b', 'a', 'b', 'a', 'a', 'a']}
end: -2
>>>
```

```
bhh.py - C:/Users/Talin for tech/Desktop/bhh.py (3.9.7)
File Edit Format Run Options Window Help
q10=""y/z/x=:
a.10
b.8====
q11=""5x+z=:
a.19
b.10====
q12=""6x+2y+z=:
a.30
b.34====
q13=""y-z=:
a.2
b.10====
q14=""x/x?:
a.1
b.3====
q15=""y-y-x*x?:
a.27
b.10====
q16=""y+z-2x=:
a.24
b.18====
q17=""y+y+x*x=:
a.40
b.45====
q18=""2x+2y+2z=:
a.16
b.26====
q19=""2z-5y=:
a.22
b.-22====
q20=""y+z-10x=:
a.7
b.-6====
dic={q1:"a",q2:"a",q3:"b",q4:"a",q5:"a",q6:"b",q7:"a",q8:"a",q9:"a",q10:"b",
      q11:"a",q12:"b",q13:"a",q14:"a",q15:"a",q16:"b",q17:"b",q18:"b",q19:"b",q20:""}
q=json.dumps(dic)
with open("q.json","w") as f:
    f.write(q)
```