

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 41

FP Practical 6

Institute of Computer Technology
B. Tech Computer Science and Engineering

Sub: (2CSE403) FUNCTIONAL PROGRAMMING

Practical 6

1. Shyam is a zookeeper at XYZ Zoo. Zoo has finite no. of animals and birds and infinite cages to accommodate them. One fine day, few amount of animals are transferred from ABC to XYZ. The animals group consists of: – A lion (being king) --Rest all animal families consisting of K members per group. $K \neq 1$ ($1 < K < 100$) The lion has to be given separate cage, rest were given one cage per group .Mr. Shyam has an unordered list of randomly arranged cage entries. The list consists of the cage numbers for all of the animals. The cage numbers will appear times per group except for the lion's room. Mr. Shyam needs you to help him find the Captain's room number. You do not know the total number of tourists or the total number of groups of families. You only know the value of and the room number list.

Code:

```
import random

cages = []

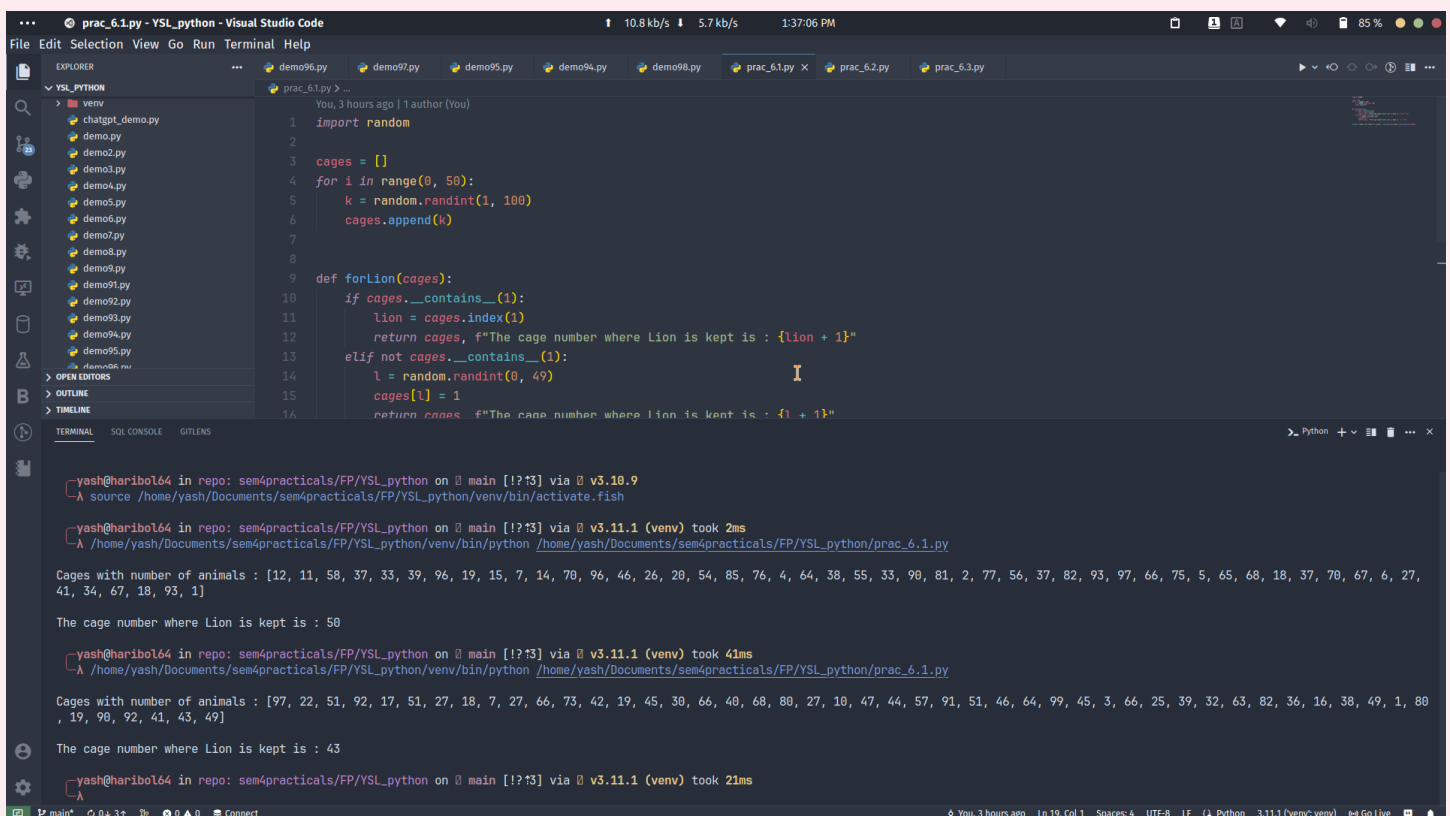
for i in range(0, 50):
    k = random.randint(1, 100)
    cages.append(k)

def forLion(cages):
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 41
FP Practical 6

```
if cages.__contains__(1):  
    lion = cages.index(1)  
    return cages, f"The cage number where Lion is kept is : {lion + 1}"  
elif not cages.__contains__(1):  
    l = random.randint(0, 49)  
    cages[l] = 1  
    return cages, f"The cage number where Lion is kept is : {l + 1}"  
  
print(f"\nCages with number of animals :  
{forLion(cages)[0]}\n\n{forLion(cages)[1]}")
```

Output :



The screenshot shows a Visual Studio Code window with a Python file named `prac_6.1.py` open. The file contains a function `forLion(cages)` that checks if the number 1 is in a list `cages`. If it is, it returns the index of 1 plus 1. If not, it randomly selects a cage number between 0 and 49, sets it to 1, and returns that number plus 1. The main function calls `forLion(cages)` and prints the result.

```
prc_6.1.py > ...  
You, 3 hours ago | 1 author (You)  
1 import random  
2  
3 cages = []  
4 for i in range(0, 50):  
5     k = random.randint(1, 100)  
6     cages.append(k)  
7  
8  
9 def forLion(cages):  
10     if cages.__contains__(1):  
11         lion = cages.index(1)  
12         return cages, f"The cage number where Lion is kept is : {lion + 1}"  
13     elif not cages.__contains__(1):  
14         l = random.randint(0, 49)  
15         cages[l] = 1  
16         return cages, f"The cage number where Lion is kept is : {l + 1}"
```

The terminal output shows the execution of the script. It displays the list of cages with the number of animals in each cage, and the cage number where the lion is kept.

```
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.10.9  
-A source /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/activate.fish  
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 2ms  
-A /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/python /home/yash/Documents/sem4practicals/FP/YSL_python/prac_6.1.py  
Cages with number of animals : [12, 11, 58, 37, 33, 39, 96, 19, 15, 7, 14, 70, 96, 46, 26, 20, 54, 85, 76, 4, 64, 38, 55, 33, 90, 81, 2, 77, 56, 37, 82, 93, 97, 66, 75, 5, 65, 68, 18, 37, 70, 67, 6, 27, 41, 34, 67, 18, 93, 1]  
The cage number where Lion is kept is : 50  
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 41ms  
-A /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/python /home/yash/Documents/sem4practicals/FP/YSL_python/prac_6.1.py  
Cages with number of animals : [97, 22, 51, 92, 17, 51, 27, 18, 7, 27, 66, 73, 42, 19, 45, 30, 66, 40, 68, 80, 27, 10, 47, 44, 57, 91, 51, 46, 64, 99, 45, 3, 66, 25, 39, 32, 63, 82, 36, 16, 38, 49, 1, 80, 19, 90, 92, 41, 43, 49]  
The cage number where Lion is kept is : 43  
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 21ms  
-A
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 41

FP Practical 6

2. A device generates 100 random characters at an instance. The input to the machine will be randomly generated characters (combination of lowercase + uppercase characters). The machine needs to return the occurrence of each character considering both cases (lowercase and uppercase) same. Kindly build a machine, which meet the requirements mentioned above. Sample: The lowercase letters are: E Y L S R I B K J V J H A B Z N W B T V s c c k r d w a m p w v u n q a m p l o A Z G D E G F I N D x m z o u l o z j v h w i w N T G X W c d o t x h y v z y z q e a m f w p g u q T R E N N W F c r f
The occurrences of each letter are: 5 a 3 b 4 c 4 d 4 e 4 f 4 g 3 h 3 i 3 j 2 k 3 l 4 m 6 n 4 o 3 p 3 q 4 r 2 s 4 t 3 u 5 v 8 w 3 x 3 y 6 z

Code:

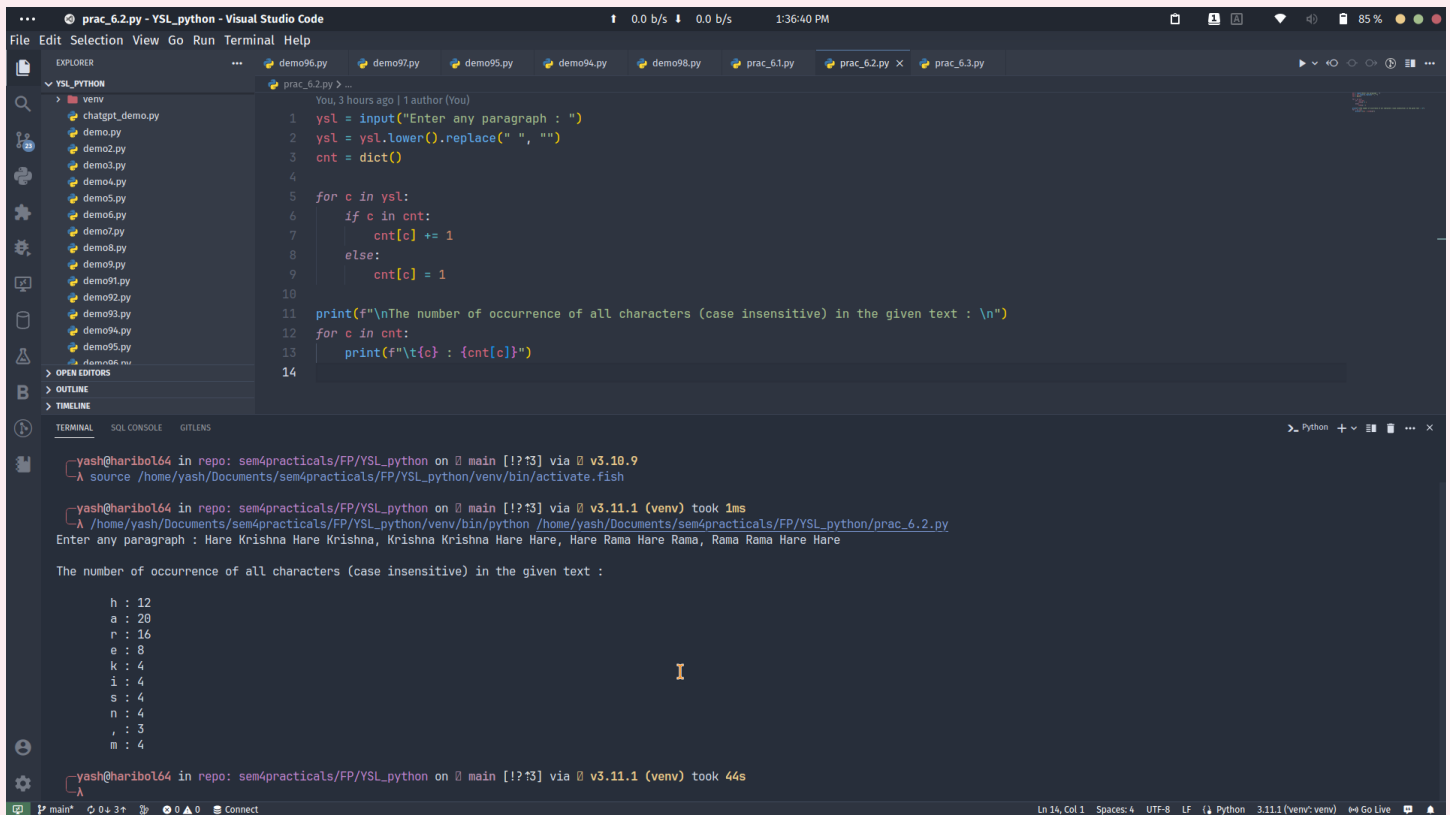
```
ysl = input("Enter any paragraph : ")
ysl = ysl.lower().replace(" ", "")
cnt = dict()

for c in ysl:
    if c in cnt:
        cnt[c] += 1
    else:
        cnt[c] = 1

print(f"\nThe number of occurrence of all characters (case insensitive) in the given text : \n")
for c in cnt:
    print(f"\t{c} : {cnt[c]}")
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 41
FP Practical 6

Output :



The screenshot shows a Visual Studio Code window with a Python file named `prac_6.2.py` open. The code is a script that takes a paragraph of text as input and counts the frequency of each character (case-insensitive). The terminal output shows the execution of the script, where the user enters a paragraph of text, and the program prints the frequency of each character.

```
prc_6.2.py > ...
You, 3 hours ago | 1 author (You)
1 ysl = input("Enter any paragraph : ")
2 ysl = ysl.lower().replace(" ", "")
3 cnt = dict()
4
5 for c in ysl:
6     if c in cnt:
7         cnt[c] += 1
8     else:
9         cnt[c] = 1
10
11 print(f"\nThe number of occurrence of all characters (case insensitive) in the given text : \n")
12 for c in cnt:
13     print(f"{c} : {cnt[c]}")
14
```

Terminal Output:

```
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.10.9
-A source /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/activate.fish

yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 1ms
-A /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/python /home/yash/Documents/sem4practicals/FP/YSL_python/prac_6.2.py
Enter any paragraph : Hare Krishna Hare Krishna, Krishna Krishna Hare Hare, Hare Rama Hare Rama, Rama Rama Hare Hare

The number of occurrence of all characters (case insensitive) in the given text :

h : 12
a : 20
r : 16
e : 8
k : 4
i : 4
s : 4
n : 4
, : 3
m : 4

yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 44s
-A
```

3. A game is being developed by a toy company, which helps the kids to learn maths table easily. Given a positive integer number n , kid has to simply enter the power value corresponding to that number. Kindly help the kid to play this game in a smooth way.

Code :

```
import random

base = random.randint(1, 10)

power = random.randint(1, 5)

n = base ** power
```

Name - Yash Lakhtariya

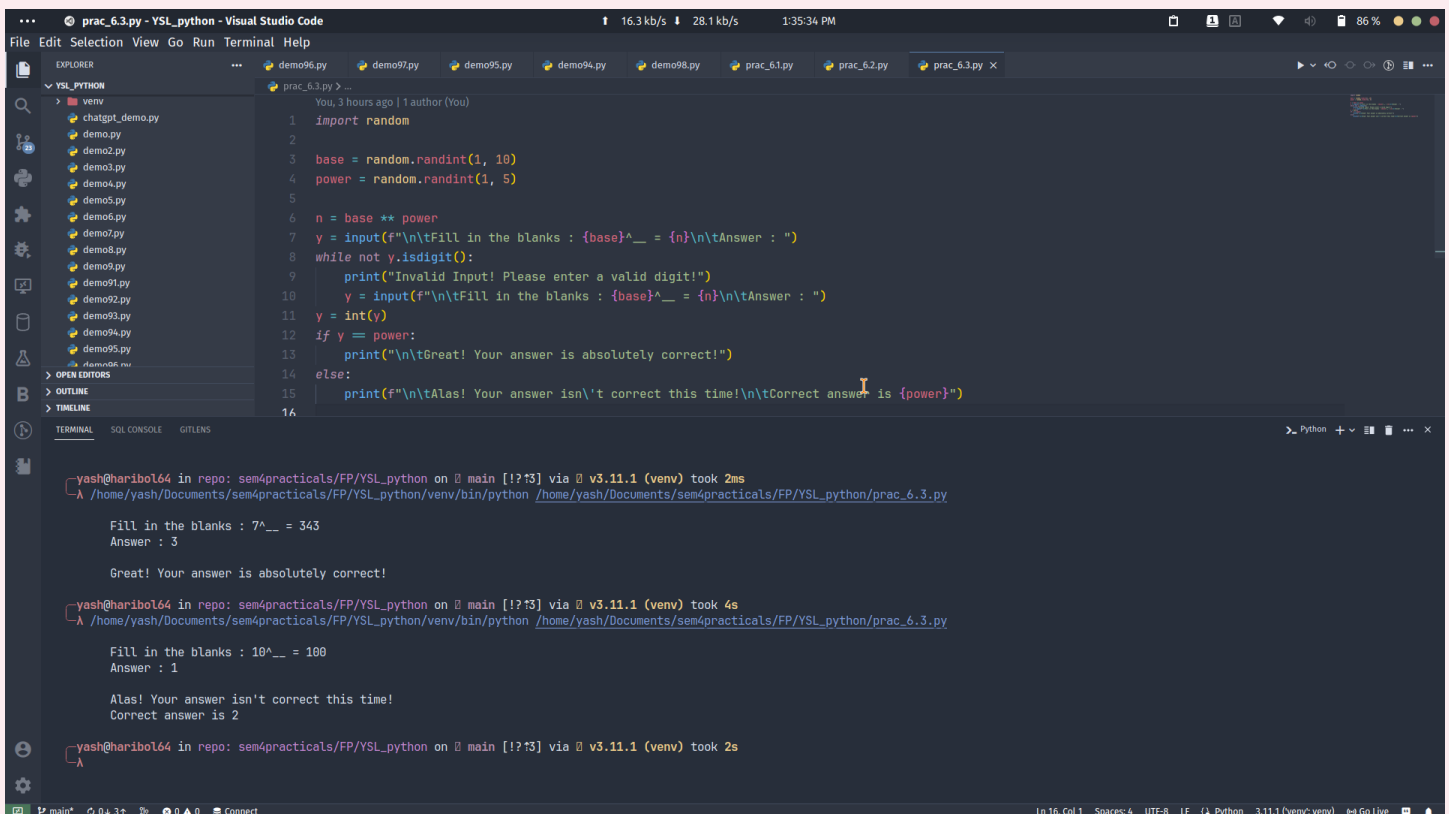
Enrollment number - 21162101012

Branch - CBA Batch - 41

FP Practical 6

```
y = input(f"\n\tFill in the blanks : {base}^__ = {n}\n\tAnswer : ")
while not y.isdigit():
    print("Invalid Input! Please enter a valid digit!")
    y = input(f"\n\tFill in the blanks : {base}^__ = {n}\n\tAnswer : ")
y = int(y)
if y == power:
    print("\n\tGreat! Your answer is absolutely correct!")
else:
    print(f"\n\tAlas! Your answer isn't correct this time!\n\tCorrect answer is {power}")
```

Output :



The screenshot shows the Visual Studio Code editor with a Python file named `prac_6.3.py` open. The code is a program that generates a random base and power, then asks the user to fill in the blanks for the equation `base^__ = n`. The program includes a loop to ensure the user enters a valid digit and provides feedback based on whether the answer is correct or incorrect.

The terminal output shows three test cases:

```
yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 2ms
A /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/python /home/yash/Documents/sem4practicals/FP/YSL_python/prac_6.3.py

Fill in the blanks : 7^__ = 343
Answer : 3

Great! Your answer is absolutely correct!

yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 4s
A /home/yash/Documents/sem4practicals/FP/YSL_python/venv/bin/python /home/yash/Documents/sem4practicals/FP/YSL_python/prac_6.3.py

Fill in the blanks : 10^__ = 100
Answer : 1

Alas! Your answer isn't correct this time!
Correct answer is 2

yash@haribol64 in repo: sem4practicals/FP/YSL_python on main [!?t3] via v3.11.1 (venv) took 2s
A
```