

| Experiment No. 3 |
| --- |
| To explore basic data types of Python like strings, lists, dictionaries and tuples |
| Date of Performance: 31/01/2024 |
| Date of Submission: 14/02/2024 |













































**Code:**

# string

s = "Sarwadeep Vijay Dhaval"

a = "Hello"

print(a+" "+s)

# list

l = []

l.append(1)

l.append(8)

print(l)

l.remove(1)

print(l)

# tuple

tup = (1,"hello",3)

print(tup)

# set

Deep = set()

Deep.add(7)

Deep.add(4)

Deep.add(2)

Deep.pop()

Deep.remove(7)

print(Deep)

Deep.clear()

print(Deep)

# Dictionary

d = {}

d[1]= "one"

d[3]= "three"

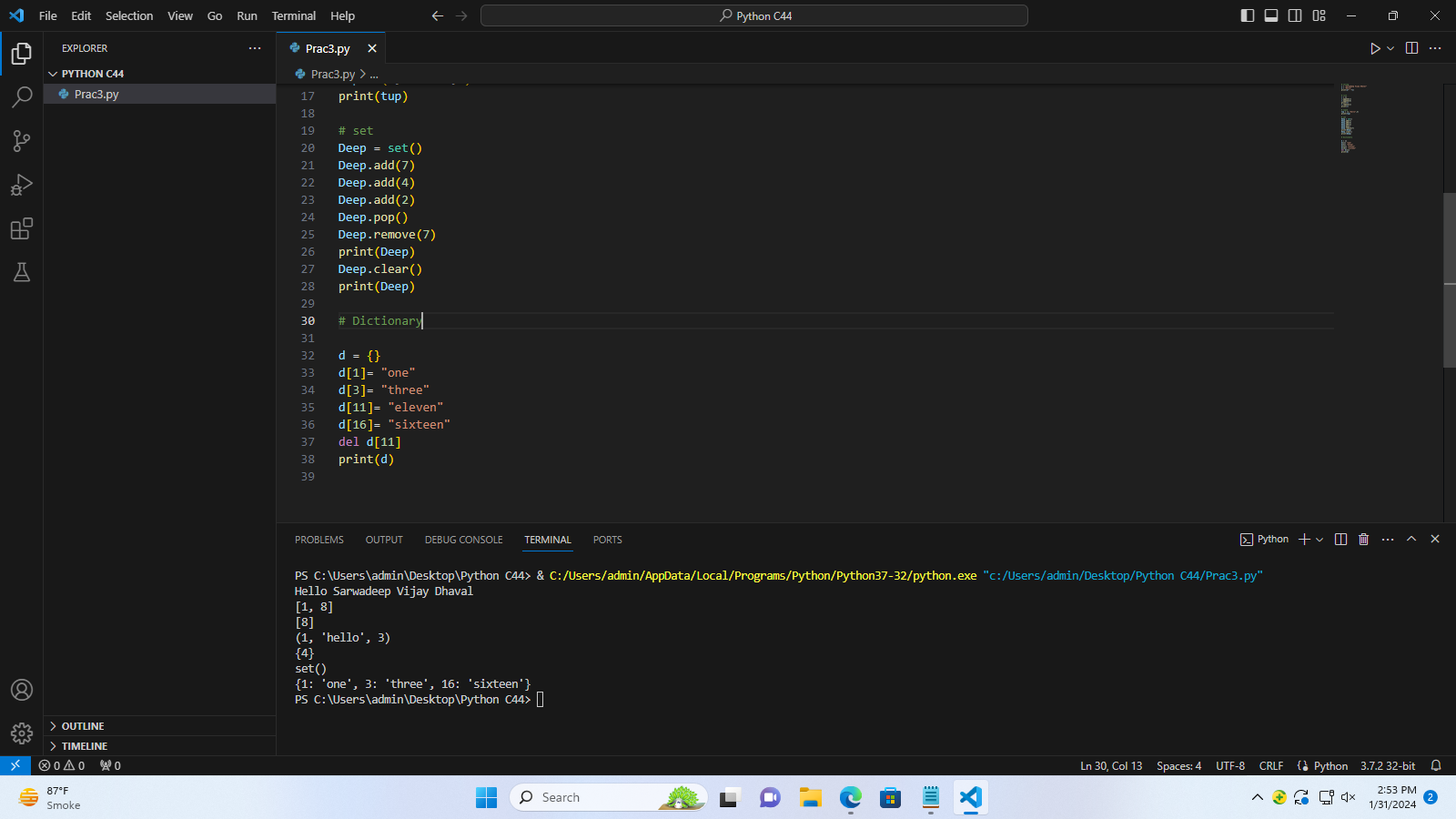
d[11]= "eleven"

d[16]= "sixteen"

del d[11]

print(d)

**Output:**



**Conclusion:**

Python offers a diverse range of built-in data types, including integers, floats, strings, booleans, lists, tuples, dictionaries, sets, and the NoneType. It is dynamically typed, meaning variables' types are inferred at runtime. Some types, like lists, dictionaries, and sets, are mutable, allowing for changes after creation, while others, like integers, floats, tuples, and strings, are immutable. Type inference enables checking the type of an object using type(), and type casting allows conversion between types using constructors. Each data type comes with its own set of methods and functions for efficient manipulation and operation. Understanding Python's data types is crucial for effective programming and data handling.