

## ASSIGNMENT-1

### Task-1:

#### Code Snippet 1: Variable Name Typo

```
python

number_of_apples = 5
print(number_of_apple)
```

**Error:** apple in line-2

**Corrected code:**

```
number_of_apples=5
print(number_of_apples)
```

**Output:**

5

**Error explanation:**

As the first line of the code is mentioned as “number\_of\_apples=5” and the second line is having the error as “print(number\_of\_apple)”. It implies name error. It can be rectified by adding ‘s’ to the apple in the second line of the code.

#### Code Snippet 2: Accessing List Elements Out of Range

```
python

fruits = ["apple", "banana", "cherry"]
print(fruits[3])
```

**Error:** index in line-2

**Corrected code:**

```
fruits=["apple","banana","cherry"]
print(fruits[2])
```

**Output:**

cherry

### Error explanation:

As the first line of the code is mentioned number of fruits given = 3 and their indices are “0,1,2”. But in the given code it is mentioned that fruits index is 3 i.e., fruits[3] . Then it implies error. It is called index error. This can be rectified by using the correct index for the code. For accessing the third fruit use the index as “2” . Hence we get the output as cherry.

### Debugging Exercise 3: Function Not Behaving as Expected

```
python

def find_average(numbers):
    sum = 0
    for number in numbers:
        sum += number
    average = sum / len(numbers)
    return average

numbers = [1, 2, 3, 4, 5, "6"]
average = find_average(numbers)
print(f"The average is: {average}")
```

**Error :** using inverted commas for “6” in the numbers (line-8).

### Corrected code :

```
def find_average(numbers):
    sum=0
    for number in numbers:
        sum+=number
    average=sum/len(numbers)
    return average

numbers=[1,2,3,4,5,6]
average=find_average(numbers)
print(f"The average is:{average}")
```

### Output :

```
The average is:3.5
```

### Error explanation :

The given code has the only error in the 8<sup>th</sup> line . This code is used to find the average of the ‘1,2,3,4,5,6’. But it consists an error called “unsupported operand”. It can be resolved by removing the unused operand i.e.,inverted commas for the number-6 in numbers. By this the error got rectified.

#### Exercise 4: Incorrect Dictionary Usage

```
python

def update_record(records, name, score):
    if name in records:
        records[name].append(score)
    else:
        records[name] = score

student_records = {"Alice": [88, 92], "Bob": [70, 85]}
update_record(student_records, "Charlie", 91)
update_record(student_records, "Alice", 95)

print(student_records)
```

#### Error:

Attribute error. It attempts to score a value that is not a list.

#### Corrected code:

```
def update_record(records, name, score):
    if name in records:
        records[name].append(score)
    else:
        records[name]=[score]

student_records={"alice": [88, 92], "bob": [70, 85]}
update_record(student_records, "charlie", 91)
update_record(student_records, "alice", 95)
print(student_records)
```

#### Output:

```
{'alice': [88, 92, 95], 'bob': [70, 85], 'charlie': [91]}
```

#### Error explanation:

As 'Charlie' is not in the student\_records and we added some values to the existed records. Changed the else block in the code to [name]=[score]. Hence the error is rectified.