

Name : Asghar Hussain

Id : 13461

Subject : Data sciences

Degree : BS(SE)

Module : 7th semester

Ans no.1:

There are different types of errors in Python.

TypeError is occur when a function is applied to an object of an inappropriate type.

```
>>> '5'+5
```

Traceback (most recent call last):

File "<pyshell#23>", line 1, in <module>

```
'5'+5
```

TypeError: must be str, not int

ValueError is occur when a function's argument is of an inappropriate type.

```
>>> int('abc')
```

Traceback (most recent call last):

File "<pyshell#14>", line 1, in <module>

```
int('abc')
```

ValueError: invalid literal for int() with base 10: 'abc'

NameError is occur when an object could not be found.

```
>>> gender
```

Traceback (most recent call last):

File "<pyshell#6>", line 1, in <module>

gender

NameError: name 'gender' is not defined

ZeroDivisionError is occur when the second operator in the division is zero.

```
>>> y=10/0
```

Traceback (most recent call last):

File "<pyshell#8>", line 1, in <module>

```
y=10/0
```

ZeroDivisionError: division by zero

IndexError is occur when trying to access an item at an invalid index.

```
>>> M1=[1,2,3]
```

```
>>> M1[3]
```

Traceback (most recent call last):

File "<pyshell#18>", line 1, in <module>

```
M1[3]
```

IndexError: list index out of range

Ans no.2:

A string in Python can be Boolean tested for truth value.

The return type will be in Boolean value (True or False)

Example:

```
my_string="I am happy"
```

```
print my_string.isalnum()
```

```
#False
```

```
print my_string.isalpha()
```

```
#False
```

```
print my_string.isdigit()
```

```
#False
```

```
print my_string.istitle()
```

```
#True
```

```
print my_string.isupper()
```

```
#False
```

```
print my_string.islower()
```

```
#False
```

```
print my_string.isspace()
```

```
#False
```

```
print my_string.endswith('d')
```

```
#True
```

```
print my_string.startswith('H')
```

```
#True
```

Ans no:3

Formatted string inputs are used to place variables inside a string. Usually strings are concatenated using the “+” sign. But in formatted strings, we use placeholders for different variables for different occasions.

Example:1

```
#CODE in Python 3.6
#not using formatted string

name = "zeeshan"

Age = 21

print("My name is " + name + "and my age is " + str(age))
```

Example:2

```
#CODE in Python 3.6
#using formatted strings

name = "zeeshan"

Age = 21

print(f"My name is {name} and my age is {age}")
```