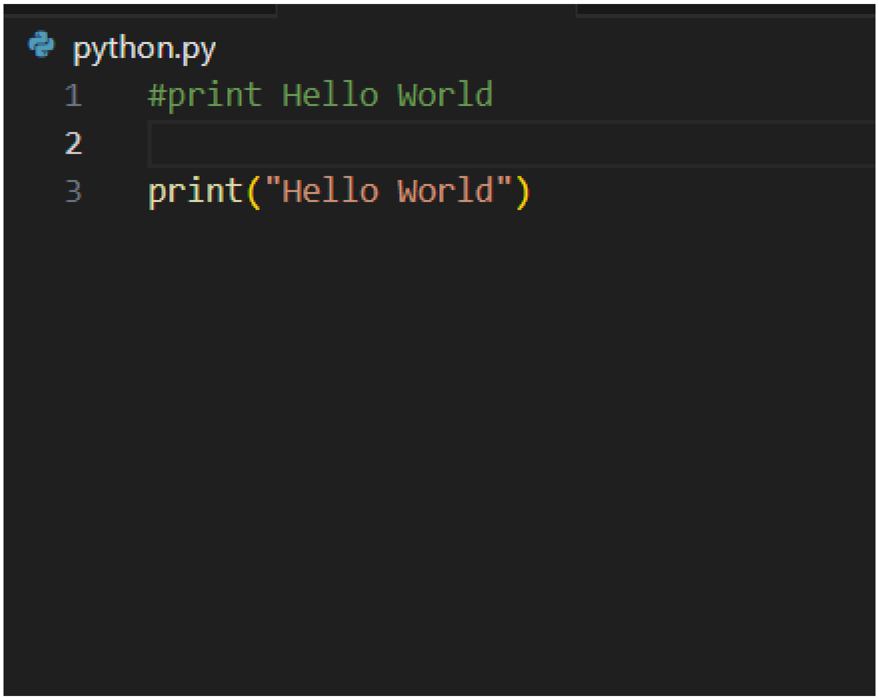
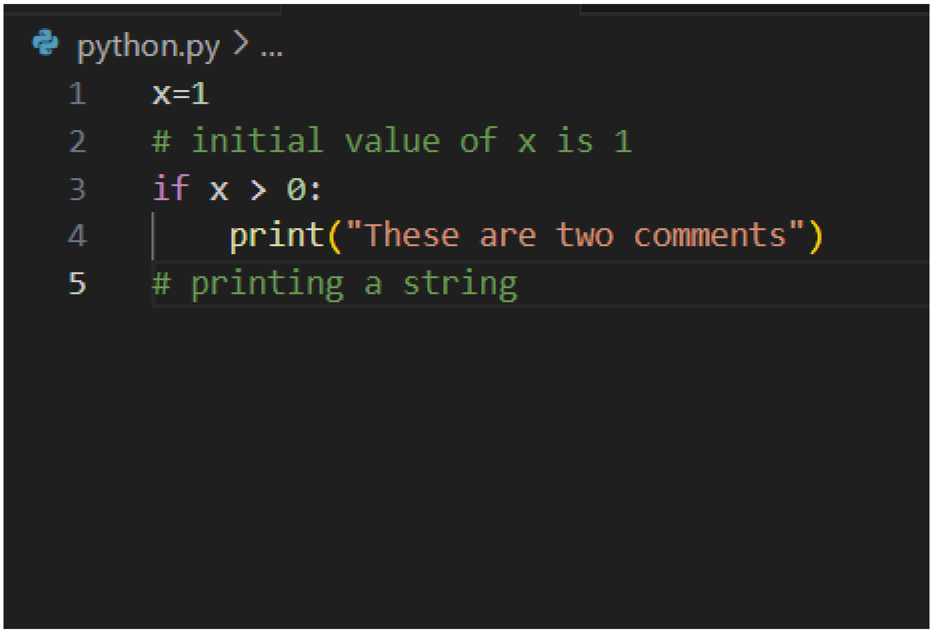
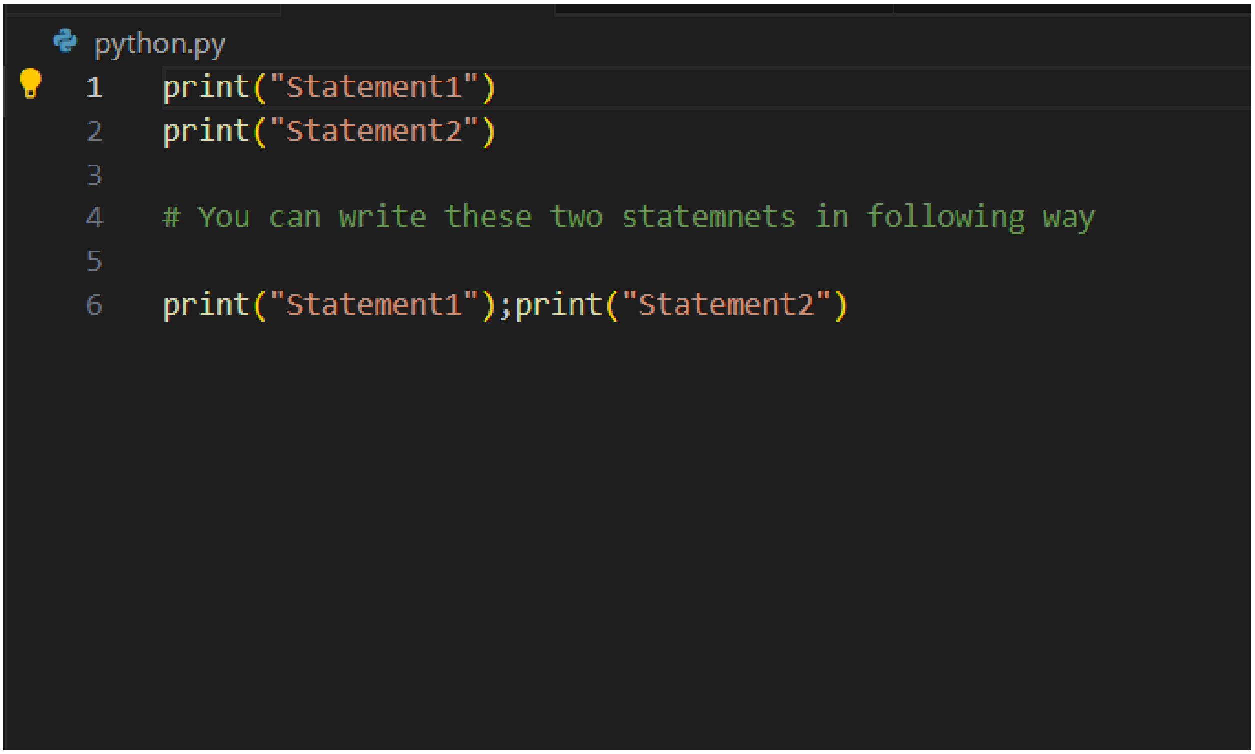


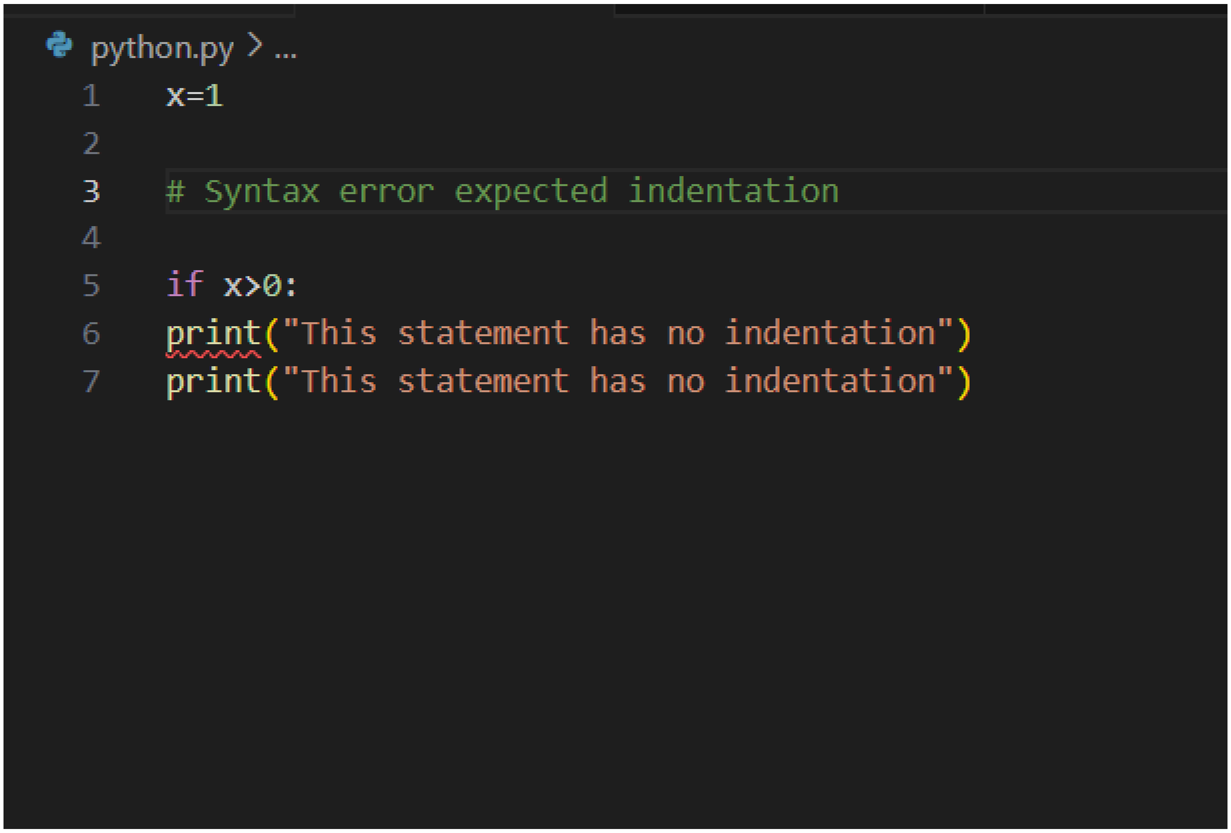
```python
#print Hello World

print("Hello World")
```



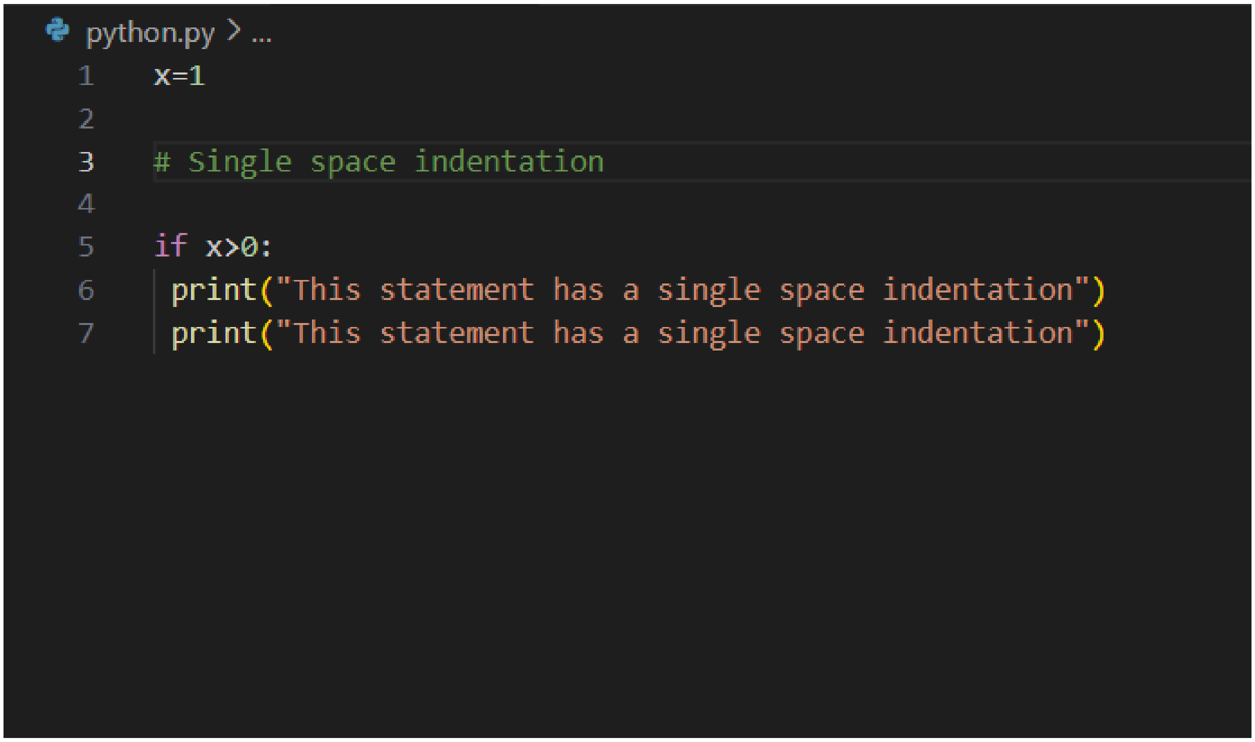
```python
x=1
# initial value of x is 1
if x > 0:
    print("These are two comments")
# printing a string
```



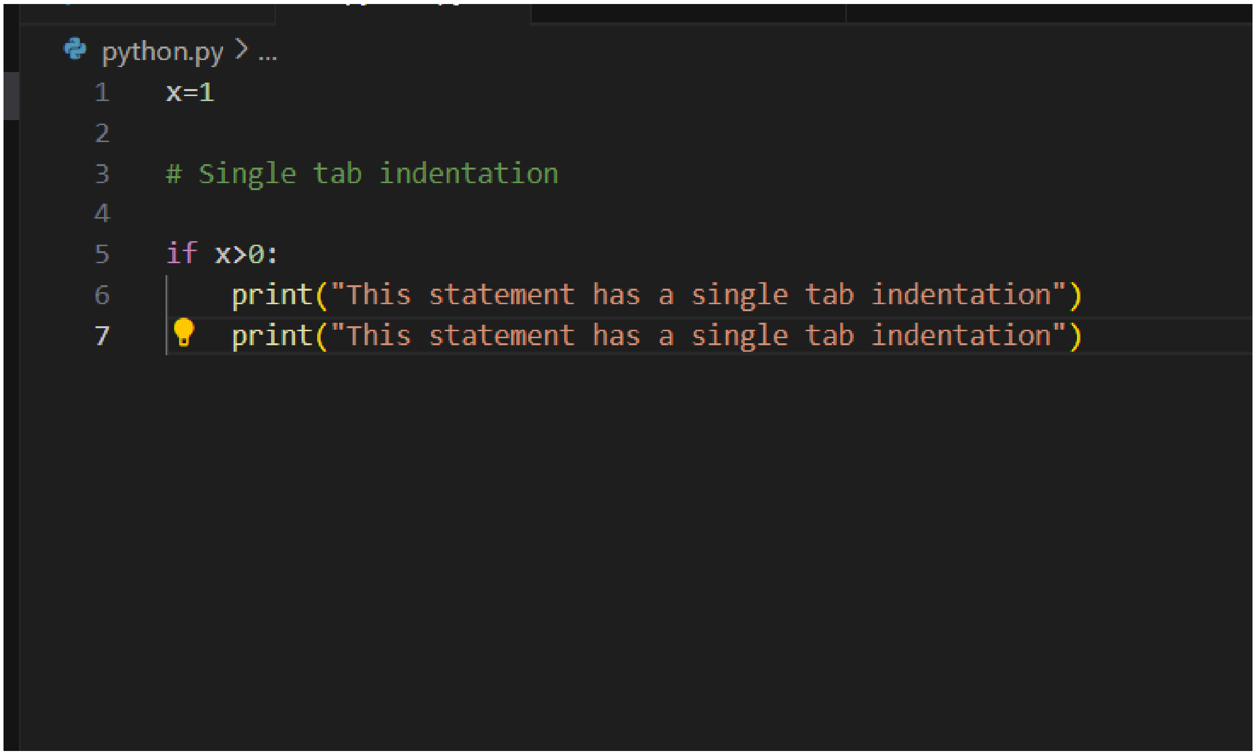
```python
print("Statement1")
print("Statement2")

# You can write these two statemnets in following way

print("Statement1");print("Statement2")
```



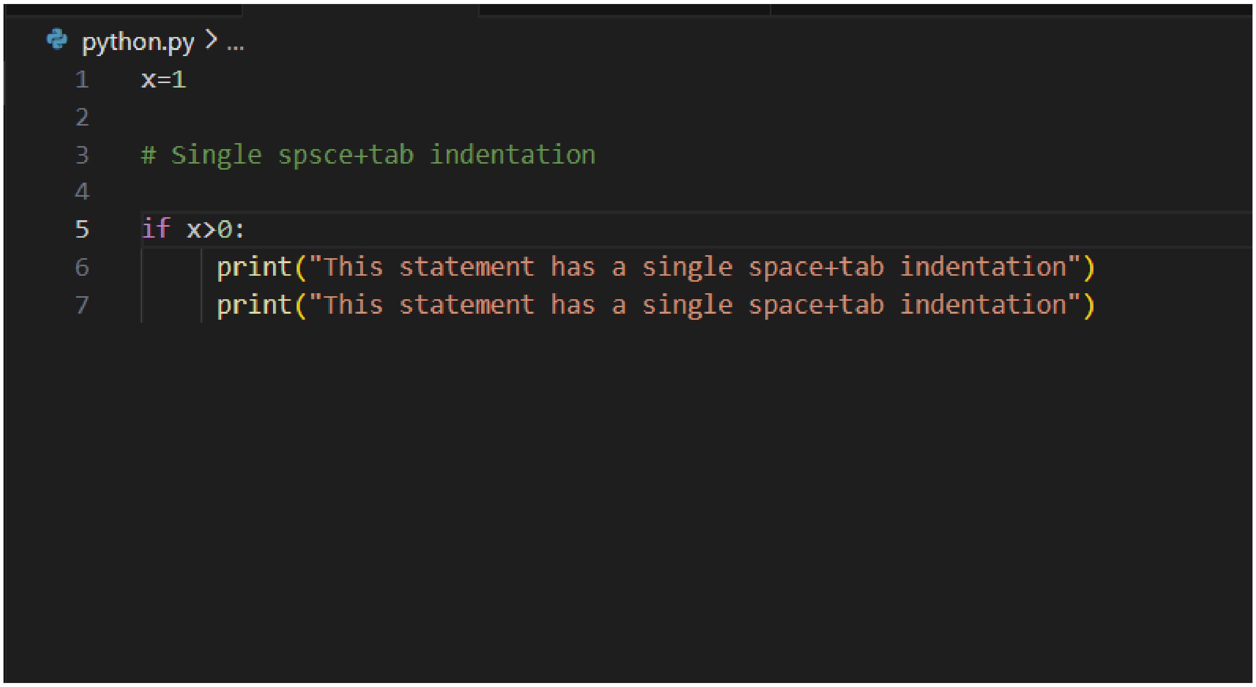
```python
x=1

# Syntax error expected indentation

if x>0:
print("This statement has no indentation")
print("This statement has no indentation")
```



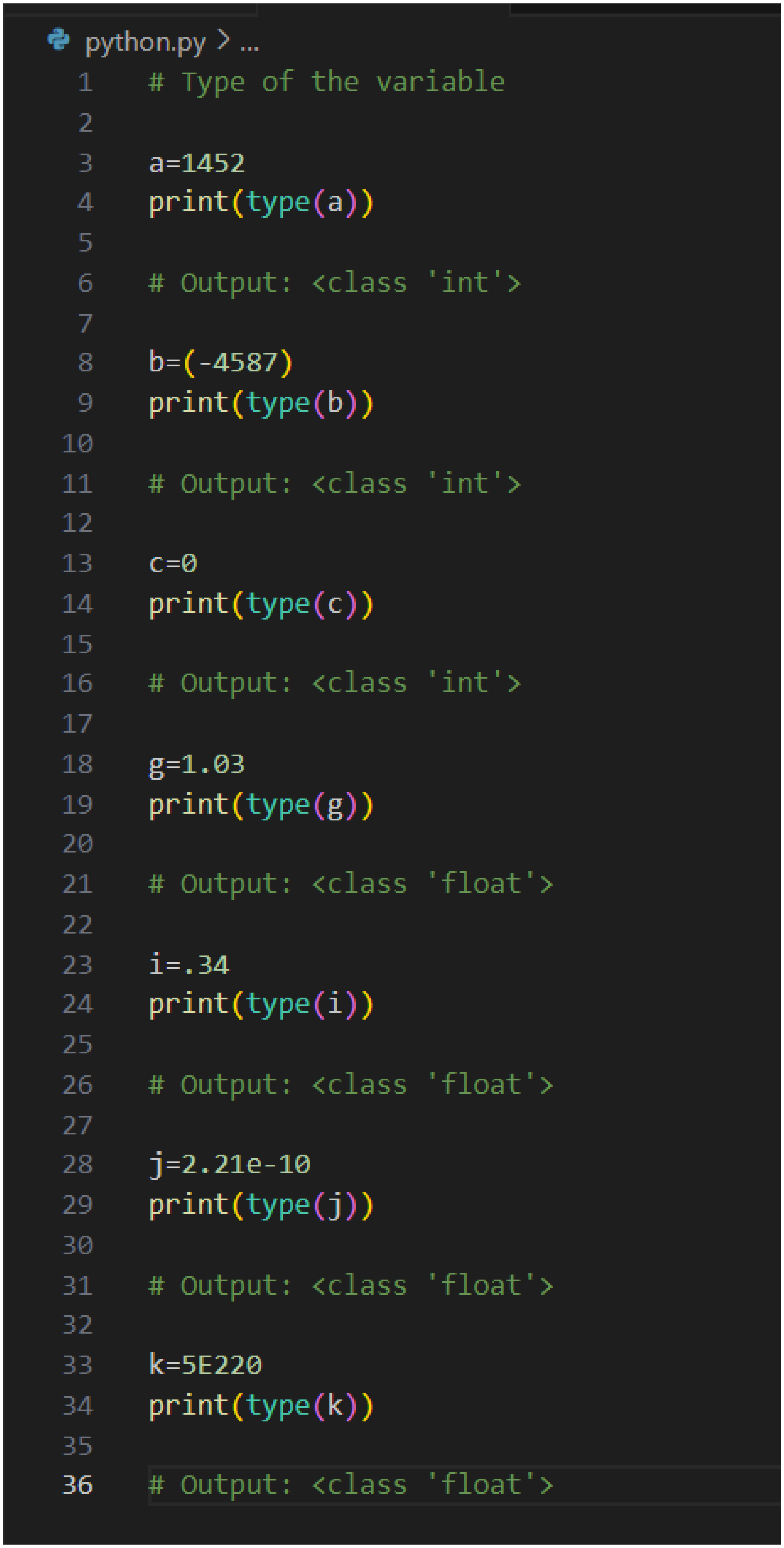
```python
x=1

# Single space indentation

if x>0:
 print("This statement has a single space indentation")
 print("This statement has a single space indentation")
```



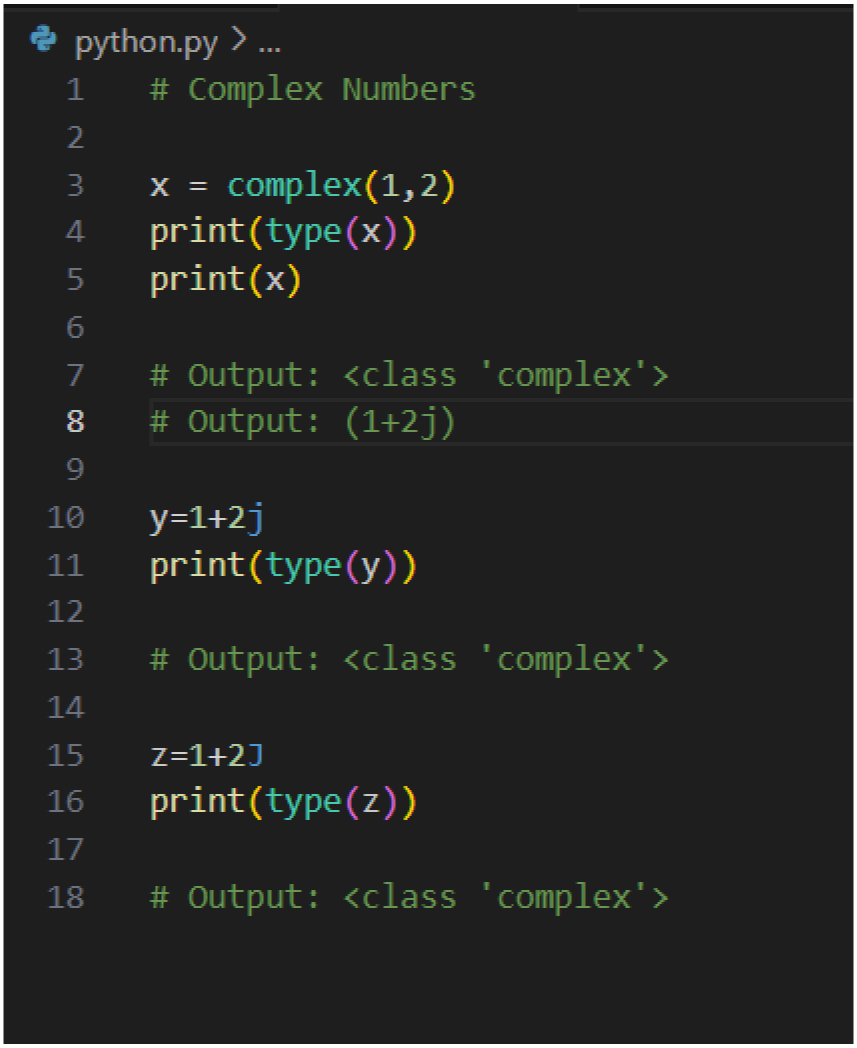
```python
x=1

# Single tab indentation

if x>0:
    print("This statement has a single tab indentation")
    print("This statement has a single tab indentation")
```



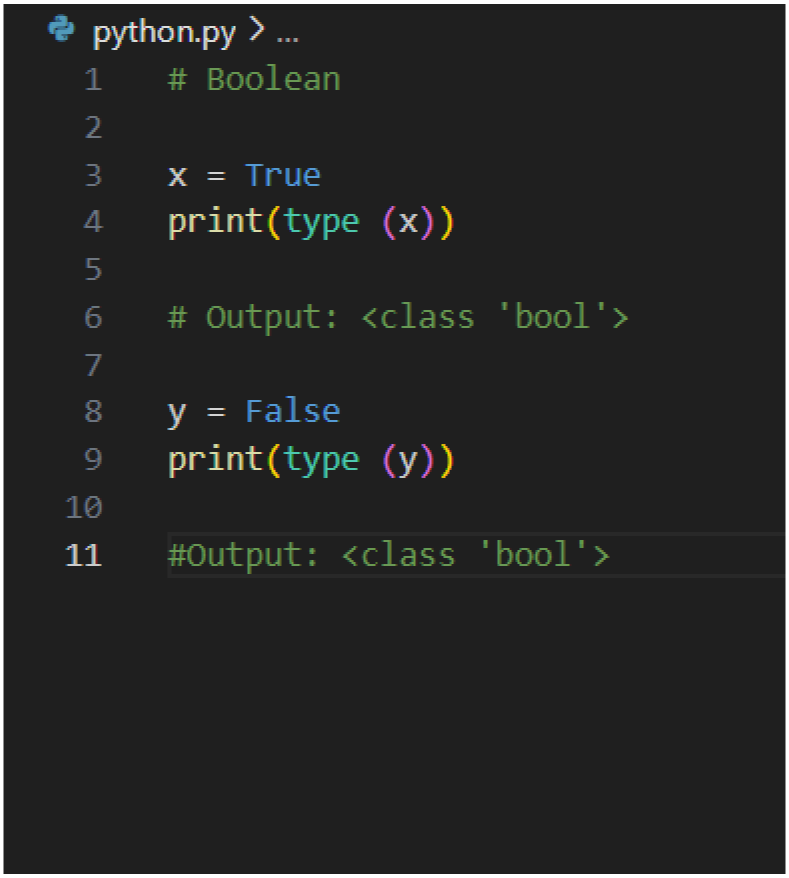
```python
x=1

# Single spsce+tab indentation

if x>0:
     print("This statement has a single space+tab indentation")
     print("This statement has a single space+tab indentation")
```



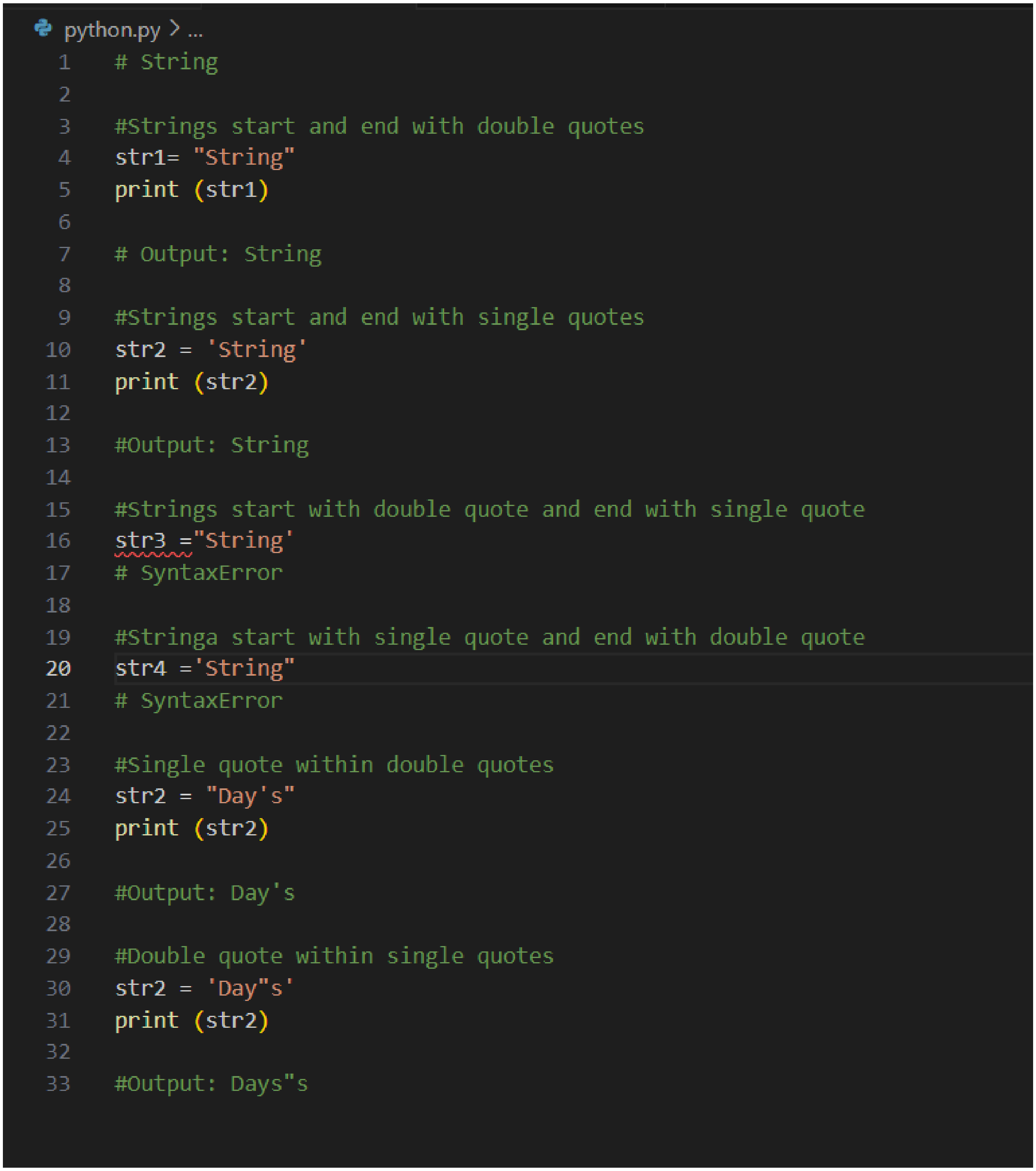
```python
# Type of the variable

a=1452
print(type(a))

# Output: <class 'int'>

b=(-4587)
print(type(b))

# Output: <class 'int'>

c=0
print(type(c))

# Output: <class 'int'>

g=1.03
print(type(g))

# Output: <class 'float'>

i=.34
print(type(i))

# Output: <class 'float'>

j=2.21e-10
print(type(j))

# Output: <class 'float'>

k=5E220
print(type(k))

# Output: <class 'float'>
```



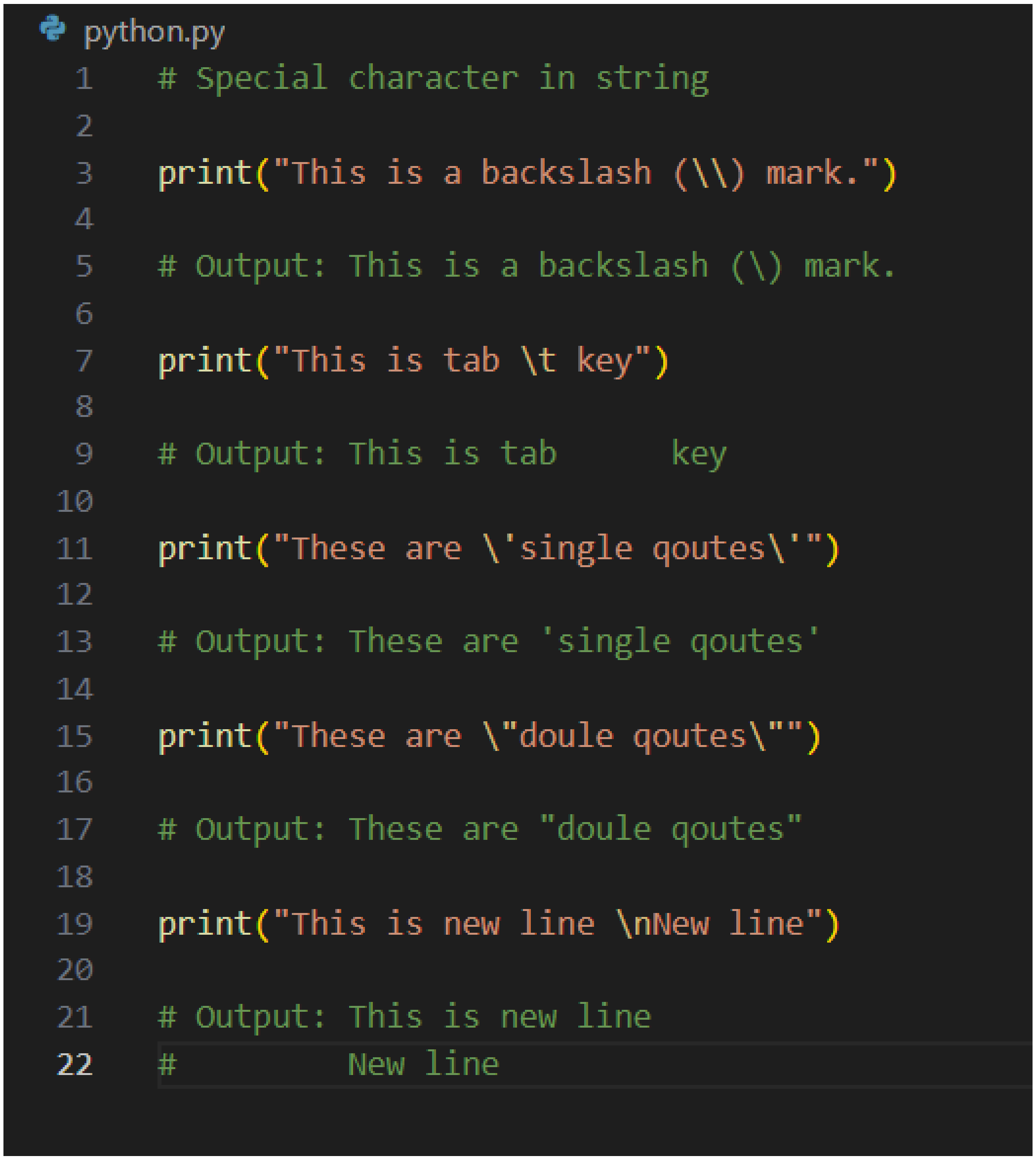
```python
# Complex Numbers

x = complex(1,2)
print(type(x))
print(x)

# Output: <class 'complex'>
# Output: (1+2j)

y=1+2j
print(type(y))

# Output: <class 'complex'>

z=1+2J
print(type(z))

# Output: <class 'complex'>
```



```python
# Boolean

x = True
print(type (x))

# Output: <class 'bool'>

y = False
print(type (y))

#Output: <class 'bool'>
```

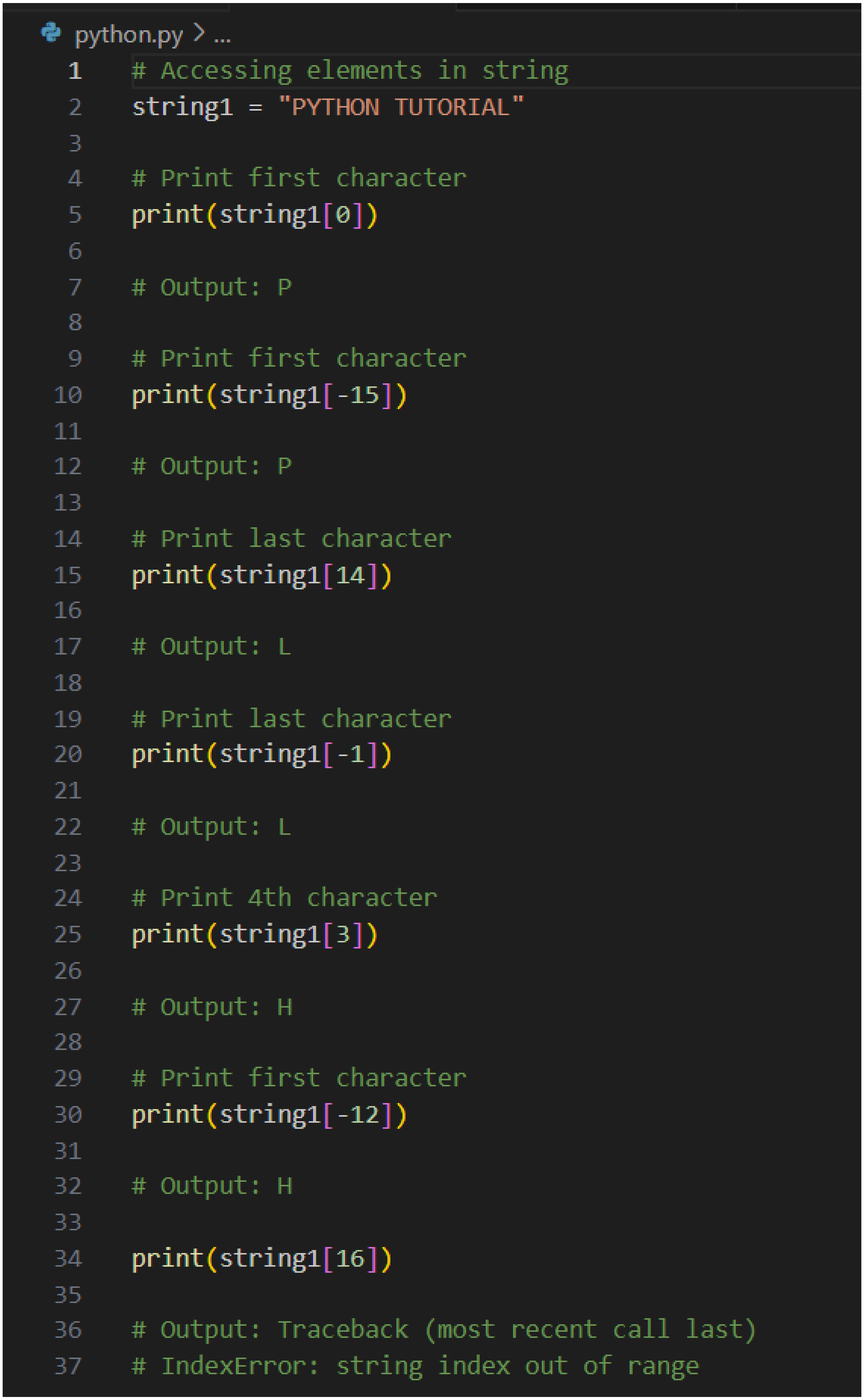


```python
# String

#Strings start and end with double quotes
str1= "String"
print (str1)

# Output: String

#Strings start and end with single quotes
str2 = 'String'
print (str2)

#Output: String

#Strings start with double quote and end with single quote
str3 ="String'
# SyntaxError

#Stringa start with single quote and end with double quote
str4 ='String"
# SyntaxError

#Single quote within double quotes
str2 = "Day's"
print (str2)

#Output: Day's

#Double quote within single quotes
str2 = 'Day"s'
print (str2)

#Output: Days"s
```

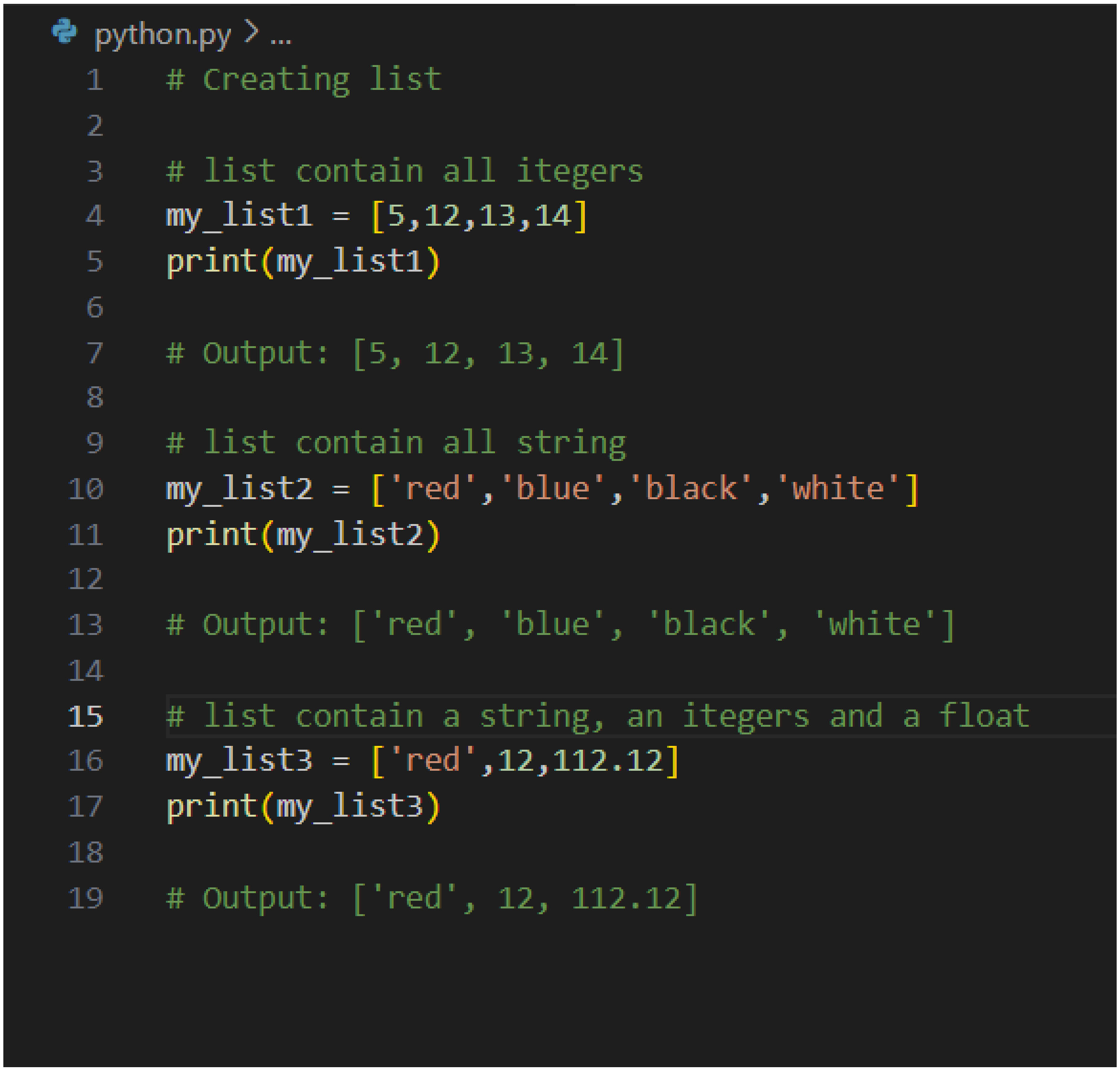


python.py

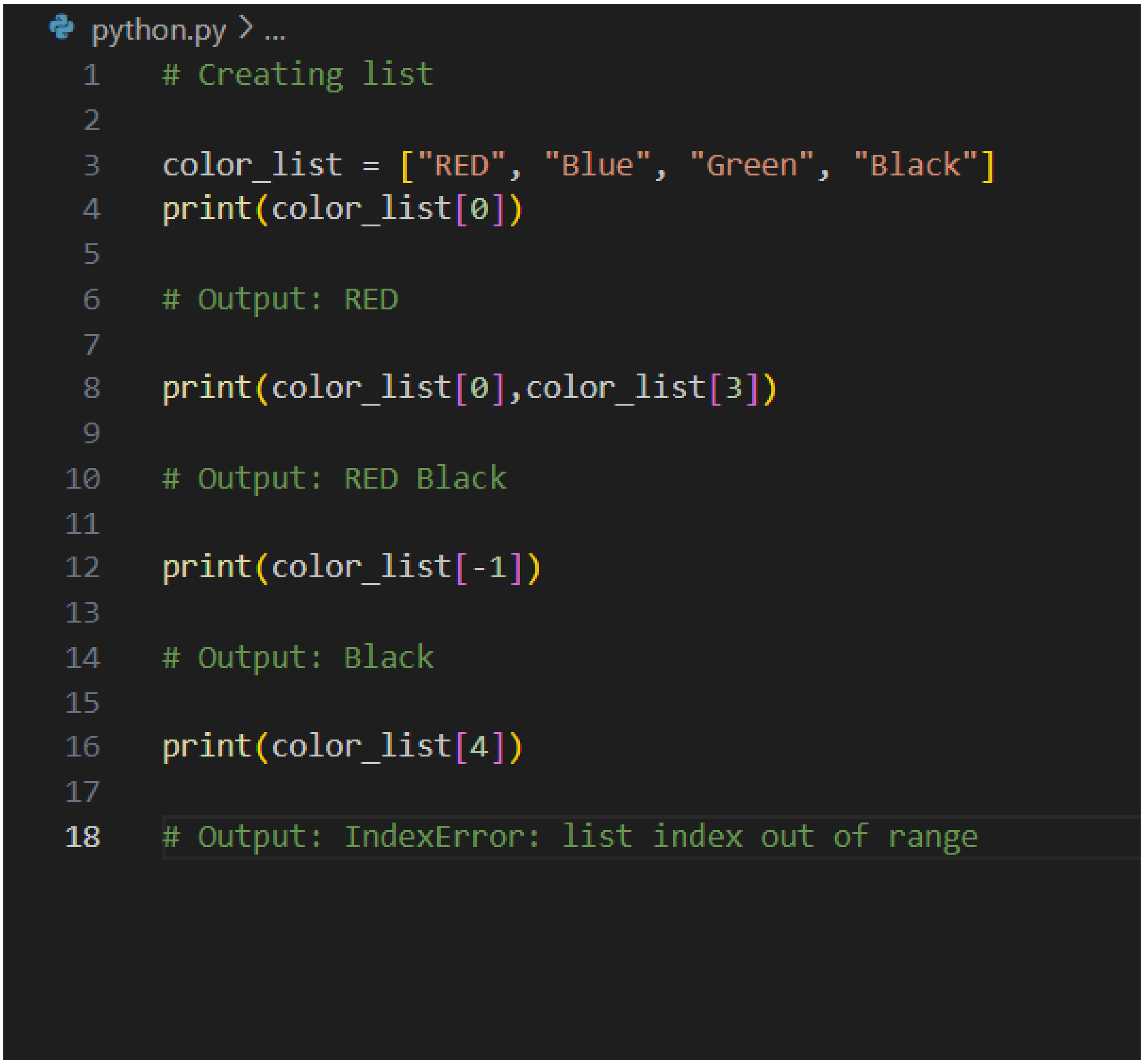
```python
# Special character in string

print("This is a backslash (\\) mark.")

# Output: This is a backslash (\) mark.

print("This is tab \t key")

# Output: This is tab      key

print("These are \'single qoutes\'")

# Output: These are 'single qoutes'

print("These are \"doule qoutes\"")

# Output: These are "doule qoutes"

print("This is new line \nNew line")

# Output: This is new line
#         New line
```



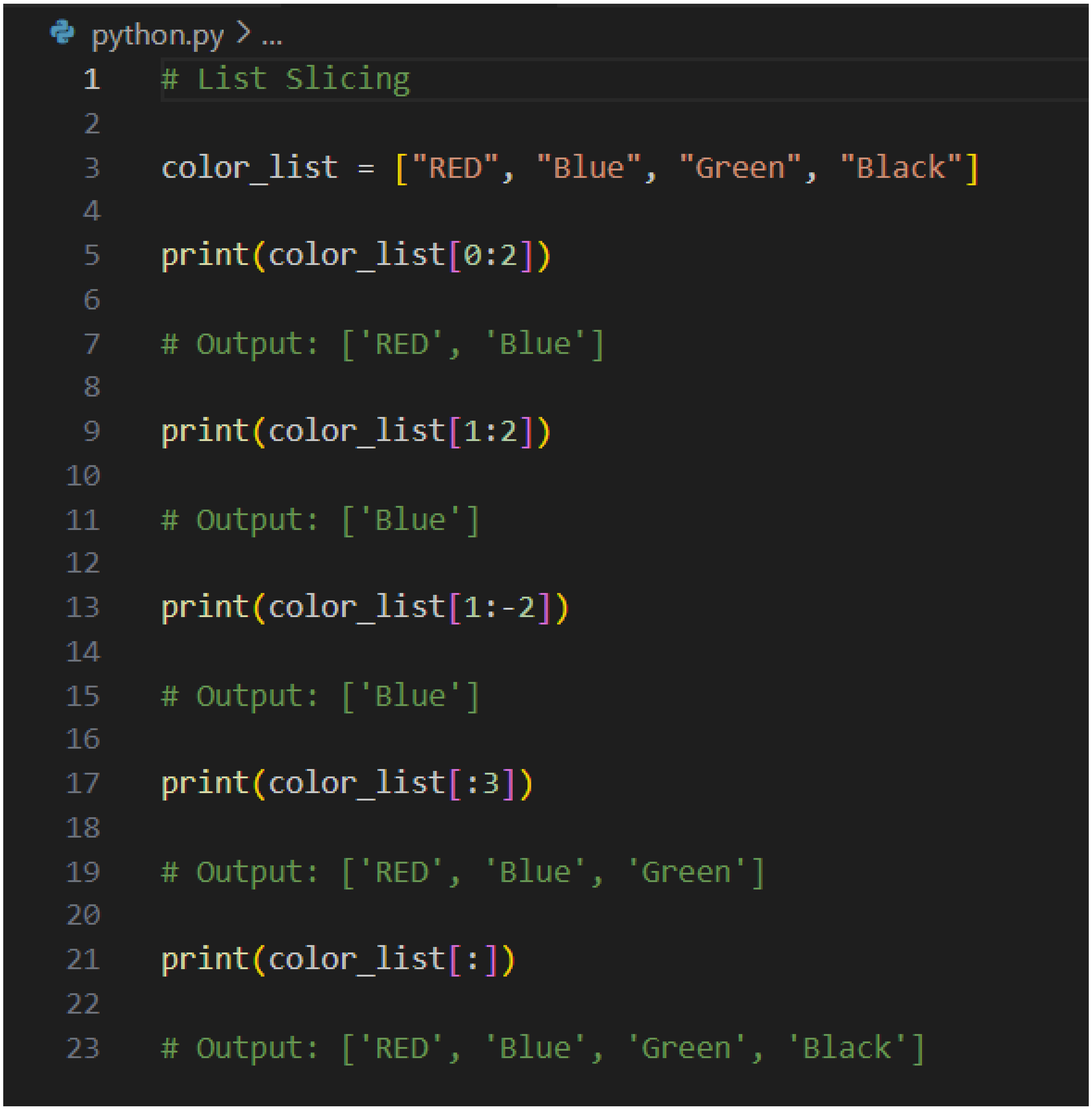
```python
# Accessing elements in string
string1 = "PYTHON TUTORIAL"

# Print first character
print(string1[0])

# Output: P

# Print first character
print(string1[-15])

# Output: P

# Print last character
print(string1[14])

# Output: L

# Print last character
print(string1[-1])

# Output: L

# Print 4th character
print(string1[3])

# Output: H

# Print first character
print(string1[-12])

# Output: H

print(string1[16])

# Output: Traceback (most recent call last)
# IndexError: string index out of range
```



```python
# Creating list

# list contain all itegers
my_list1 = [5,12,13,14]
print(my_list1)

# Output: [5, 12, 13, 14]

# list contain all string
my_list2 = ['red','blue','black','white']
print(my_list2)

# Output: ['red', 'blue', 'black', 'white']

# list contain a string, an itegers and a float
my_list3 = ['red',12,112.12]
print(my_list3)

# Output: ['red', 12, 112.12]
```



```python
# Creating list

color_list = ["RED", "Blue", "Green", "Black"]
print(color_list[0])

# Output: RED

print(color_list[0],color_list[3])

# Output: RED Black

print(color_list[-1])

# Output: Black

print(color_list[4])

# Output: IndexError: list index out of range
```



```python
# List Slicing

color_list = ["RED", "Blue", "Green", "Black"]

print(color_list[0:2])

# Output: ['RED', 'Blue']

print(color_list[1:2])

# Output: ['Blue']

print(color_list[1:-2])

# Output: ['Blue']

print(color_list[:3])

# Output: ['RED', 'Blue', 'Green']

print(color_list[:])

# Output: ['RED', 'Blue', 'Green', 'Black']
```