f string in python

f-strings in Python

PEP 498 introduced a new string formatting mechanism known as *Literal String Interpolation* or more commonly as *F-strings* (because of the leading *f* character preceding the string literal). The idea behind f-strings is to make string interpolation simpler.

```python
# Python3 program introducing f-string
val = 'Metaeducators'
print(f"{val} is a portal for learning coding in mother language .")


name = 'shubhanshu'
age = 17
print(f"Hello, My name is {shubhanshu} and I'm {age} years old.")
```
Output :

```
Hello, My name is shubhanshu and I'm 17 years old.
metaeducators is a portal for learning coding in mother language .
```

```python
# Prints today's date with help
# of datetime library

import datetime

today = datetime.datetime.today()
print(f"{today:%B %d, %Y}")
```

Output :

```
April 04, 2018
```

Note : F-strings are faster than the two most commonly used string formatting mechanisms, which are % formatting and str.format().

Let's see few error examples, which might occur while using f-string :

Code #3 : Demonstrating Syntax error.

```python
answer = 456
f"Your answer is "{answer}""
```
Code #4 : Backslash Cannot be used in format string directly.

```python
f"newline: {ord('\n')}"
```

Output :

```
Traceback (most recent call last):
  Python Shell, prompt 29, line 1
Syntax Error: f-string expression part cannot include a backslash: , line
```

```
1, pos 0
```

But the documentation points out that we can put the backslash into a variable as a workaround though :

```python
newline = ord('\n')

f"newline: {newline}"
```

Output :

```
newline: 10
```

```python
newline = ord('\n')
```