COMP110

CL08 - f-strings and Practice with Conditionals

Reminders

Quiz 00:

- *Regrade requests will be open till Wednesday, Feb 5 at 11:59pm*. Please submit a regrade request if you believe your quiz was not graded correctly according to the rubric
    - Please do not ask questions about content in regrade requests. Instead, come see us in office hours/tutoring!

f-strings (formatted string literals)

A helpful way to embed expressions directly into strings!

Without f-strings:

```
print("They are " + str(30 + 1))
```

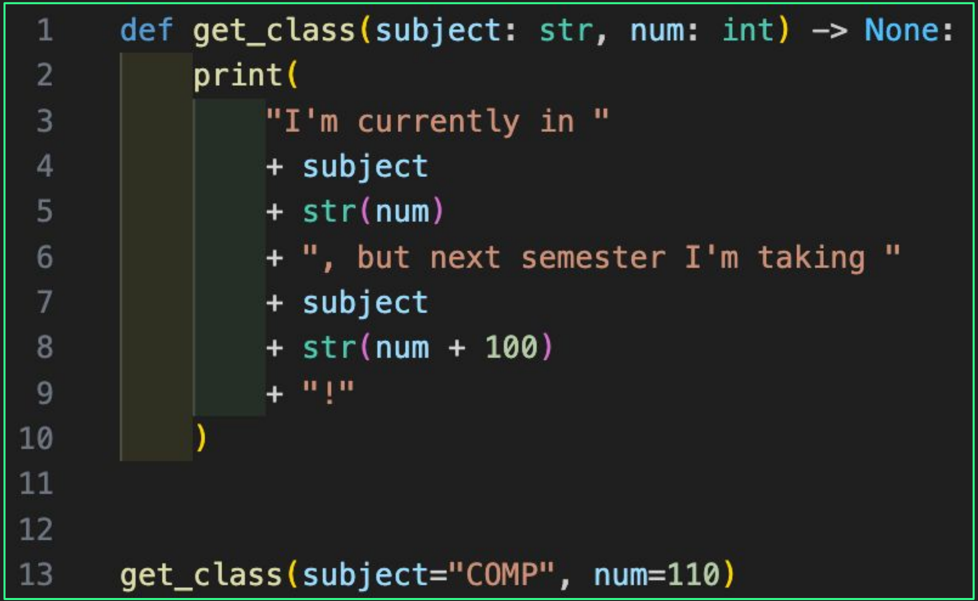
With f-strings:

```
print(f"They are {30 + 1}")
```

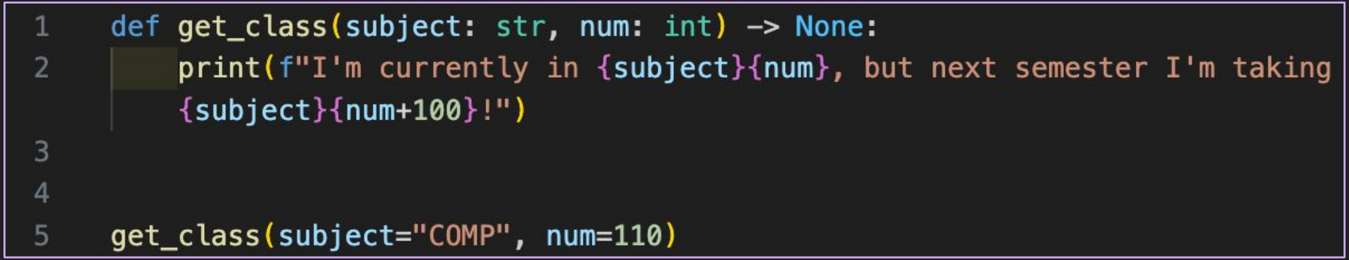
Both will output the string:

```
They are 31
```

f-strings (formatted string literals)

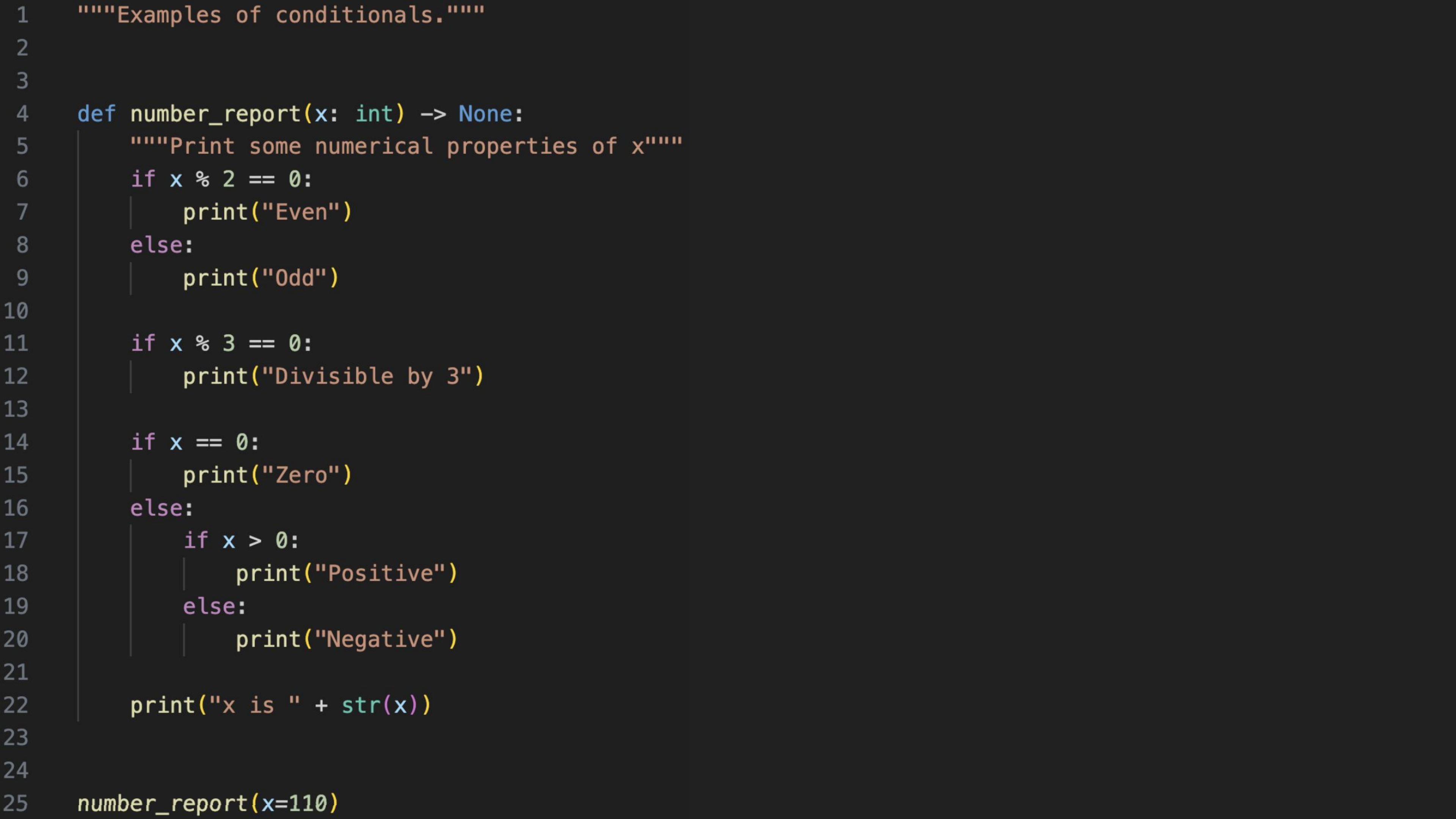


```python
def get_class(subject: str, num: int) -> None:
    print(
        "I'm currently in "
        + subject
        + str(num)
        + ", but next semester I'm taking "
        + subject
        + str(num + 100)
        + "!"
    )


get_class(subject="COMP", num=110)
```

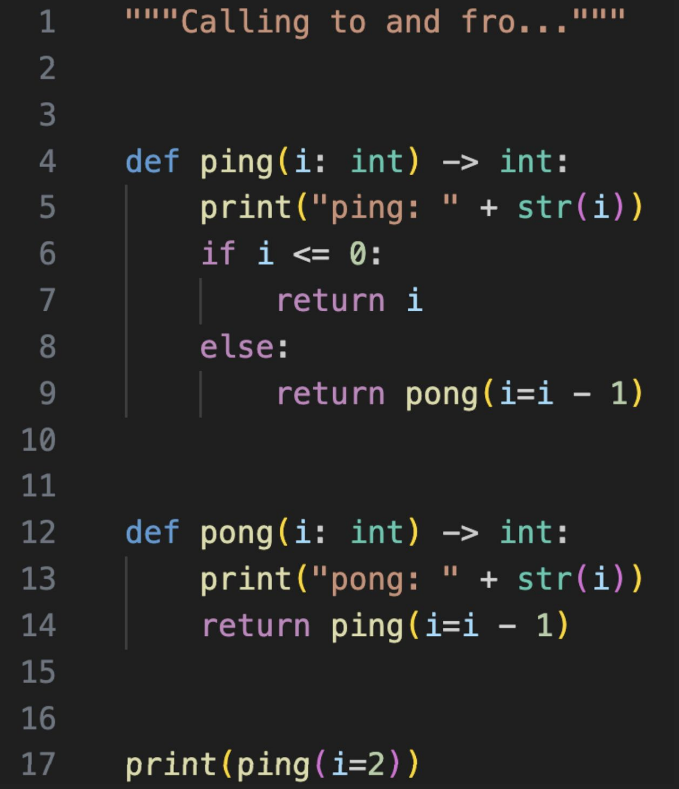


```python
def get_class(subject: str, num: int) -> None:
    print(f"I'm currently in {subject}{num}, but next semester I'm taking {subject}{num+100}!")


get_class(subject="COMP", num=110)
```

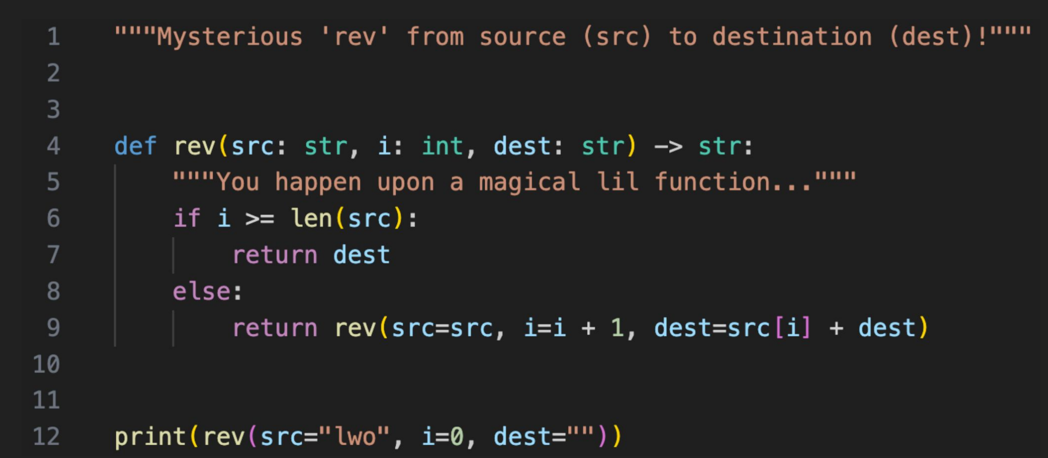
Will these two versions of the get_class function print the exact same phrase?



```python
"""Examples of conditionals."""


def number_report(x: int) -> None:
    """Print some numerical properties of x"""
    if x % 2 == 0:
        print("Even")
    else:
        print("Odd")

    if x % 3 == 0:
        print("Divisible by 3")

    if x == 0:
        print("Zero")
    else:
        if x > 0:
            print("Positive")
        else:
            print("Negative")

    print("x is " + str(x))


number_report(x=110)
```



```python
"""Calling to and fro..."""


def ping(i: int) -> int:
    print("ping: " + str(i))
    if i <= 0:
        return i
    else:
        return pong(i=i - 1)


def pong(i: int) -> int:
    print("pong: " + str(i))
    return ping(i=i - 1)


print(ping(i=2))
```



```python
"""Mysterious 'rev' from source (src) to destination (dest)!"""


def rev(src: str, i: int, dest: str) -> str:
    """You happen upon a magical lil function..."""
    if i >= len(src):
        return dest
    else:
        return rev(src=src, i=i + 1, dest=src[i] + dest)


print(rev(src="lwo", i=0, dest=""))
```