NYU | TANDON SCHOOL OF ENGINEERING

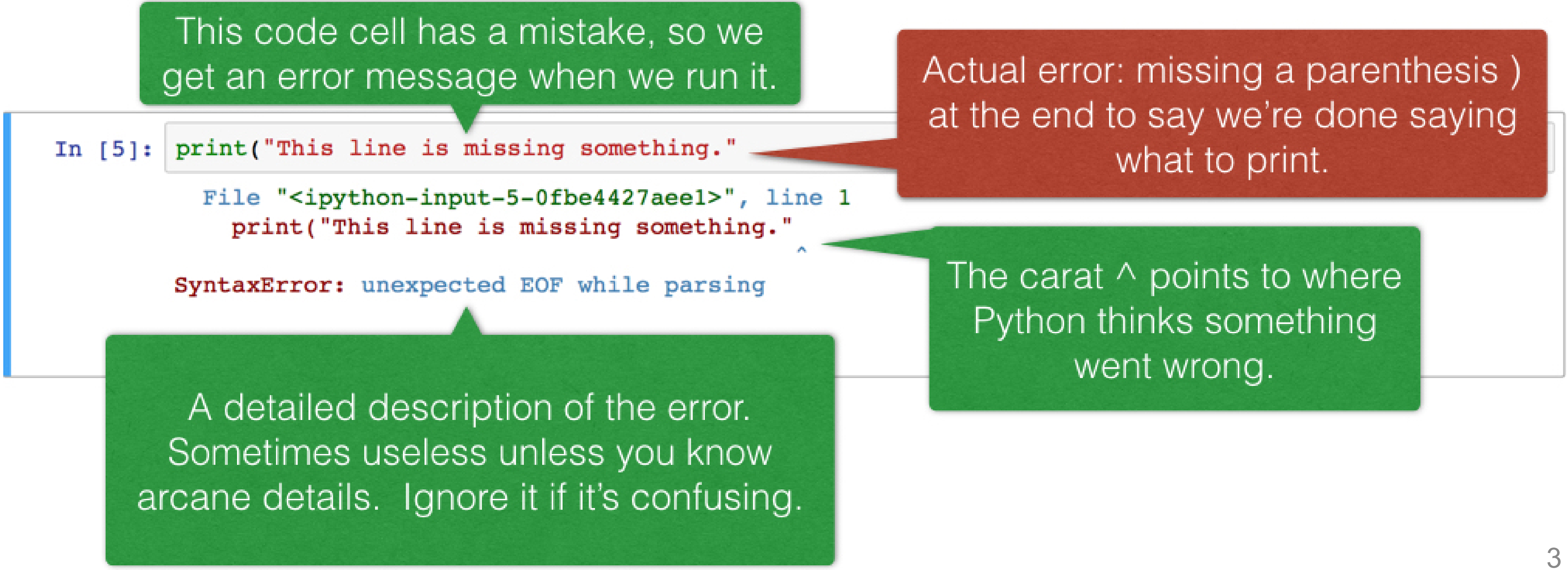
Lab 2
Packaging Code

Programming for Business
Intelligence and Analytics
MG-GY 8401

Agenda

- Exceptions
  - try statements
- Testing
  - assert keyword
  - raise keyword

- Exceptions are event that disrupts the execution of a program



This code cell has a mistake, so we get an error message when we run it.

```
In [5]: print("This line is missing something."

          File "<ipython-input-5-0fbe4427aee1>", line 1
            print("This line is missing something."
                                                   ^

        SyntaxError: unexpected EOF while parsing
```

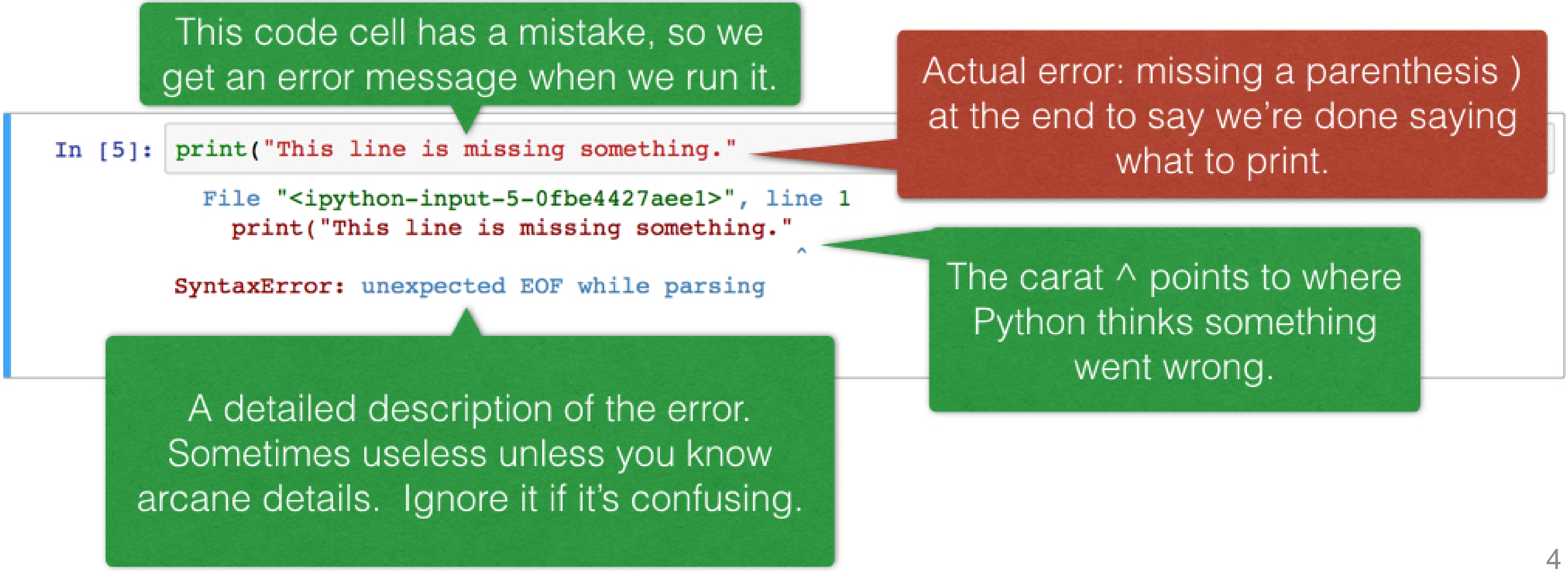
Actual error: missing a parenthesis) at the end to say we're done saying what to print.

The carat ^ points to where Python thinks something went wrong.

A detailed description of the error. Sometimes useless unless you know arcane details. Ignore it if it's confusing.

Exceptions

- Exceptions causes programs to terminate execution by default.



This code cell has a mistake, so we get an error message when we run it.

```
In [5]: print("This line is missing something."

          File "<ipython-input-5-0fbe4427aee1>", line 1
            print("This line is missing something."
                                                   ^
        SyntaxError: unexpected EOF while parsing
```

Actual error: missing a parenthesis) at the end to say we're done saying what to print.

The carat ^ points to where Python thinks something went wrong.

A detailed description of the error. Sometimes useless unless you know arcane details. Ignore it if it's confusing.

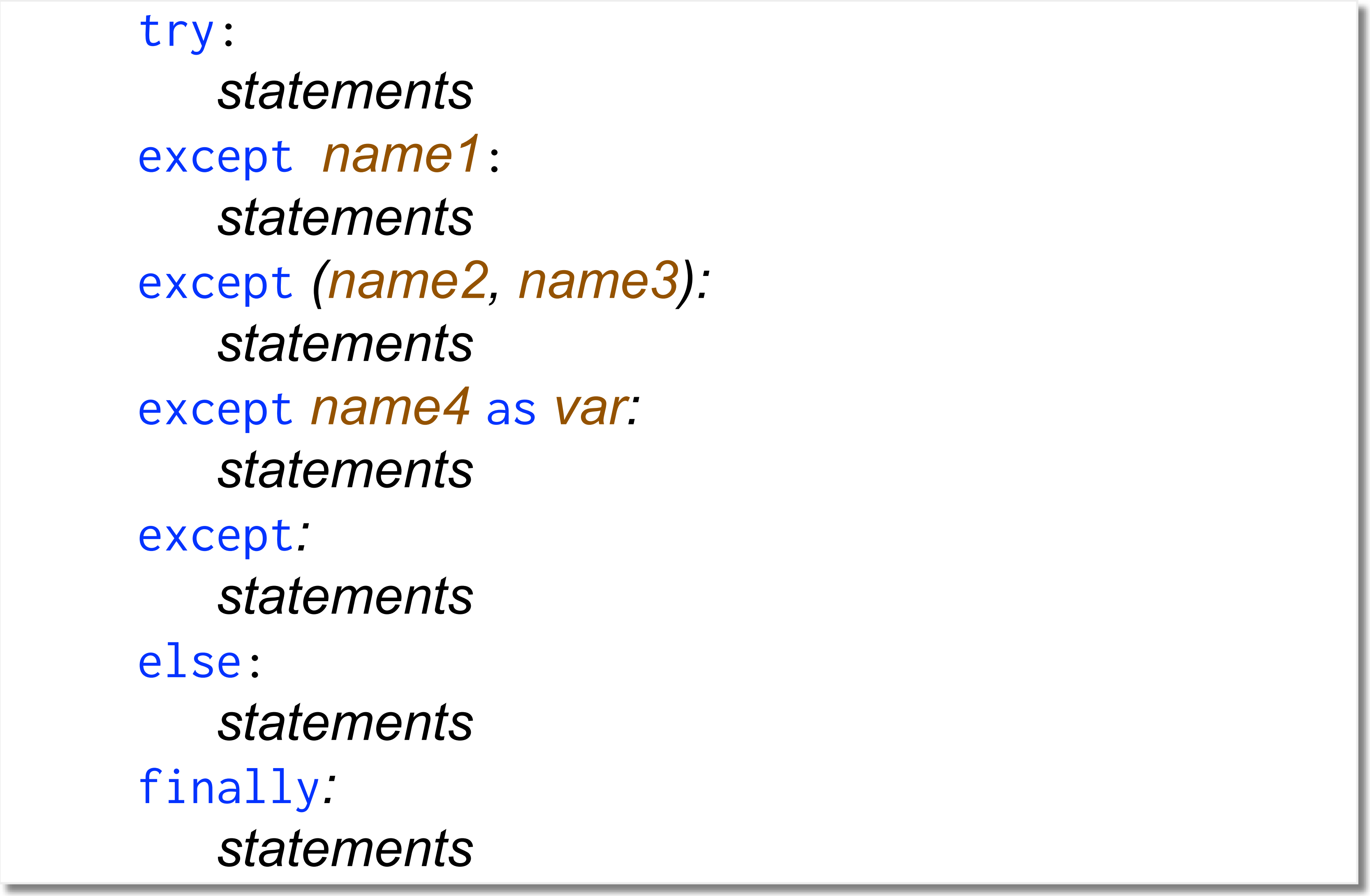
Exceptions

- Developers should attempt to handle exceptions to make more robust code

```python
def fetcher(obj, index):
    return obj[index]
```

```
>>> fetcher(x, 4)
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
File "<stdin>", line 2, in fetcher
IndexError: string index out of range
```

Handling Exceptions



```
try:
    statements
except name1:
    statements
except (name2, name3):
    statements
except name4 as var:
    statements
except:
    statements
else:
    statements
finally:
    statements
```

NYU TANDON SCHOOL OF ENGINEERING

Raising Exceptions

- The raise statement is used to force a specified exception to occur

```
>>> raise NameError(‘Hi There’)
Traceback (most recent call last):
  File "<stdin>", line 1, in ?
NameError: HiThere
```

Raising Exceptions

- The assert statement is used to raise an exception based on the value of a logical expression

```
>>> assert 1 == 2
Traceback (most recent call last):
  File "<stdin>", line 1, in ?
AssertionError:
```