Think Python 2e, Chapter 19 Notes

Goodies: Conditional expressions and list comprehensions

November 11, 2022

Conditional expressions

```
1  if x > 0:
2      y = math.log(x)
3  else:
4      y = float('nan')
```

```
1  y = math.log(x) if x > 0 else float('nan')
```

Recursive functions using conditional expressions

```python
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)
```

```python
def factorial(n):
    return 1 if n == 0 else n * factorial(n-1)
```

Optional arguments using conditional expressions

```python
def __init__(self, name, contents=None):
    self.name = name
    if contents == None:
        contents = []
    self.pouch_contents = contents
```

```python
def __init__(self, name, contents=None):
    self.name = name
    self.pouch_contents = [] if contents == None
else contents
```

List comprehensions

```
1 def capitalize_all(t):
2     res = []
3     for s in t:
4         res.append(s.capitalize())
5     return res
```

```
1 def capitalize_all(t):
2     return [s.capitalize() for s in t]
```

List comprehensions

```python
def only_upper(t):
    res = []
    for s in t:
        if s.isupper():
            res.append(s)
    return res
```

```python
def only_upper(t):
    return [s for s in t if s.isupper()]
```

Debugging list comprehensions

- Cannot put a print statement in the loop.
- Recommended only if simple or already done in loop