COMP
110

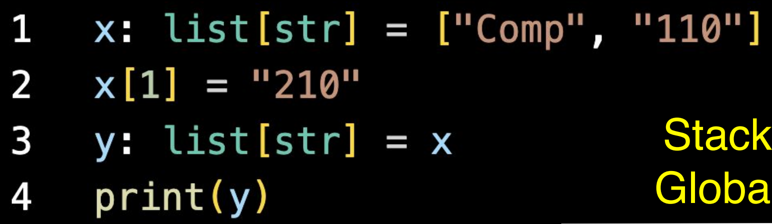
CL05 - Lists Practice

Thursday Recap…

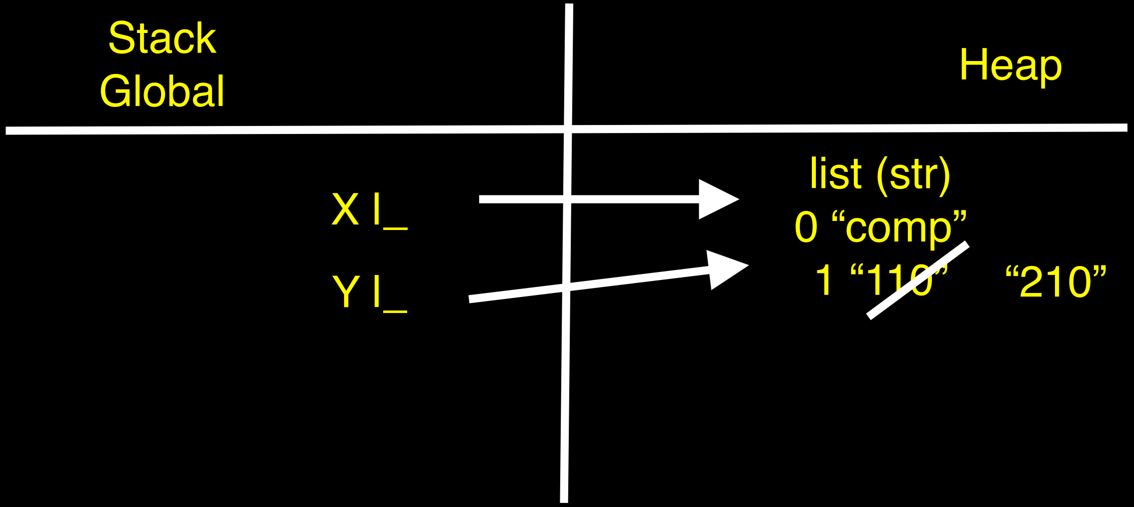
- A list is a **data structure**–something that lets you reason about multiple items.
- Syntax:
  - grocery_list: list[str] = [“eggs”, “milk”, “bread”]
- Can be an arbitrary length
- Empty List: list() or []
- Indexing like strings, but can *modify* by index
- Methods: append and pop

Lists in Memory

if we modify x then we are modifying y



```python
x: list[str] = ["Comp", "110"]
x[1] = "210"
y: list[str] = x
print(y)
```



Stack
Global

Heap

X |_

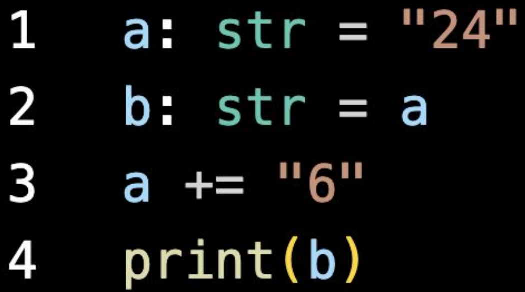
Y |_

list (str)
0 “comp”
1 ~~“110”~~ “210”

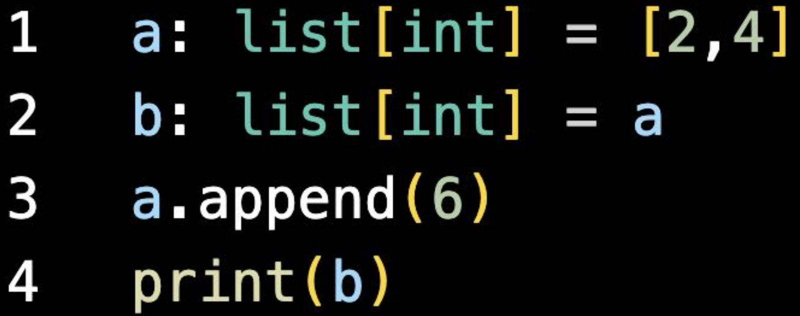
output

[“Comp”, “210”]

Comparing Lists and Strings

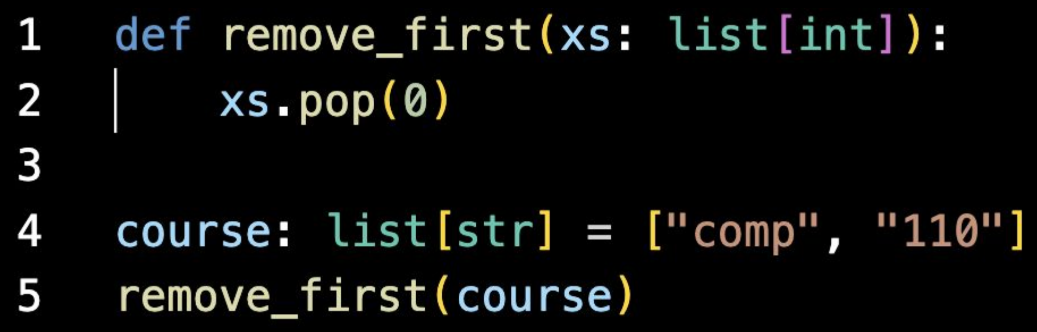


```
a: str = "24"
b: str = a
a += "6"
print(b)
```



```
a: list[int] = [2,4]
b: list[int] = a
a.append(6)
print(b)
```

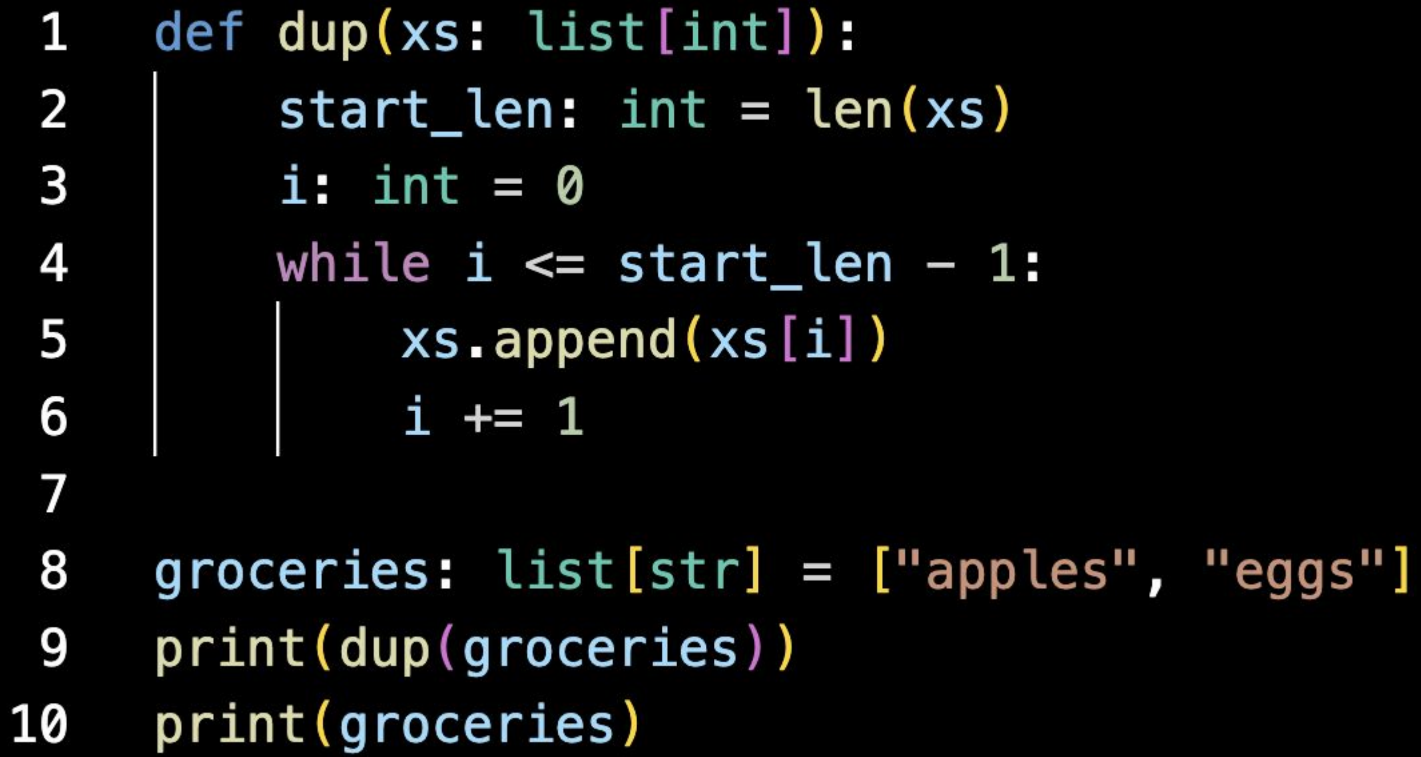
Lists + Functions



```python
def remove_first(xs: list[int]):
    xs.pop(0)

course: list[str] = ["comp", "110"]
remove_first(course)
```

Lists + Functions



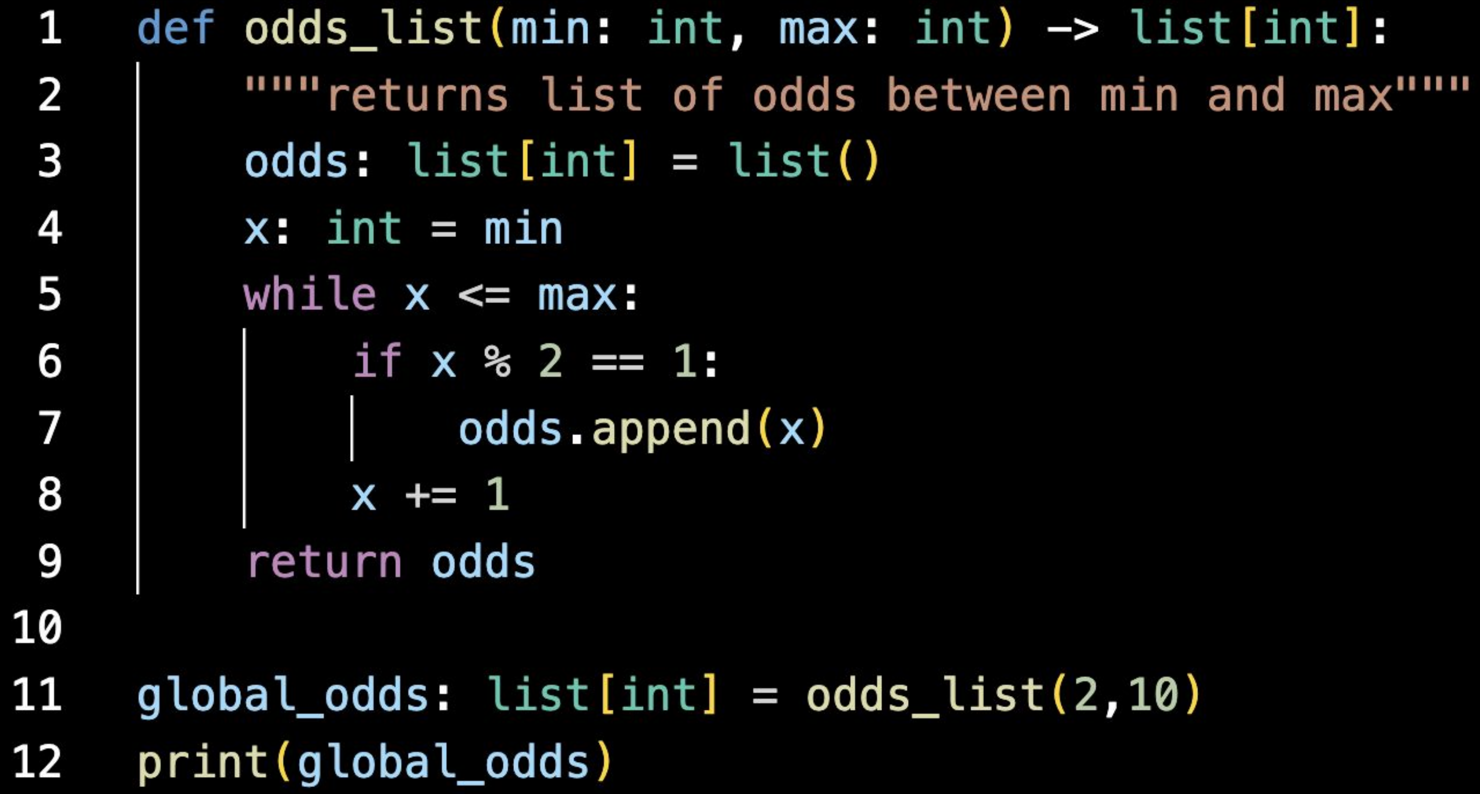
```python
def dup(xs: list[int]):
    start_len: int = len(xs)
    i: int = 0
    while i <= start_len - 1:
        xs.append(xs[i])
        i += 1

groceries: list[str] = ["apples", "eggs"]
print(dup(groceries))
print(groceries)
```

```python
def dup(xs: list[int]):
    start_len: int = len(xs)
    i: int = 0
    while i <= start_len - 1:
        xs.append(xs[i])
        i += 1

groceries: list[str] = ["apples", "eggs"]
print(dup(groceries))
print(groceries)
```

Creating a list with a function



```python
def odds_list(min: int, max: int) -> list[int]:
    """returns list of odds between min and max"""
    odds: list[int] = list()
    x: int = min
    while x <= max:
        if x % 2 == 1:
            odds.append(x)
        x += 1
    return odds

global_odds: list[int] = odds_list(2,10)
print(global_odds)
```

```python
def odds_list(min: int, max: int) -> list[int]:
    """returns list of odds between min and max"""
    odds: list[int] = list()
    x: int = min
    while x <= max:
        if x % 2 == 1:
            odds.append(x)
        x += 1
    return odds

global_odds: list[int] = odds_list(2,10)
print(global_odds)
```

Coding Example (if we have time)

- Let's implement a function where we can call with 2 arguments:
  - A “needle" int value we are searching for
  - A "haystack" list of values we are searching in

The return value of the function should be True if in the haystack at least once and False otherwise

The name of the function will be contains