

Method syntax

Michael Ernst

CSE 190p

University of Washington

Digression: two meanings of period (.)

- **Namespace management: `math.sin(math.pi)`**
 - Purpose: avoid polluting the global environment (the oldest “stack frame”) with too many names
 - Example:
 - `abs` is in the global namespace
 - When writing a program about bodybuilding, you might use `abs` (for abdominals)
 - Suddenly, all code that depends on the built-in `abs` fails!
 - Another examples: `draw()` function for pictures, for cowboys, and for curtains
 - Solution: A different namespace or module
- **“Method syntax” for a function call: `mylist.append(new_elt)`**
 - `append()` takes two arguments
 - First argument may appear before a period
- These are *distinct* meanings of a single token, “.”

Method call syntax

- Ordinary function call: `fn(arg1, arg2, arg3)`
 - Any Python function can be invoked this way
- “Method syntax”: `arg1.fn(arg2, arg3)`
 - First argument may appear before a period
 - Some Python functions can be invoked this way
 - Only works for functions defined in a type’s namespace
 - Such a function is called a “method”

```
nums = [1, 2, 3]
nums.append(4)
list.append(nums, 5) # append is defined in the list namespace
append(nums, 6)     # NameError: name 'append' is not defined
len(nums)
nums.len()          # AttributeError: 'list' object has no attribute 'len'
```