

Types

CSE 190p

University of Washington

Michael Ernst

What types do you know about?

- int
- float
- string
- boolean
- list
- set
- dict
- function
- none
- file

A function works for particular types

```
def fahr_to_cent(fahr):  
    """Argument is a number, in degrees Fahrenheit.  
    Returns a number, the corresponding degrees Centigrade.  
    """  
    return (fahr - 32) * 5.0 / 9
```

```
fahr_to_cent(212)
```

```
# When and where does the error occur?
```

```
fahr_to_cent("32" + "F")
```

Review: how to evaluate a function call

1. Evaluate the function and its arguments to values
 - If the function value is not a function, execution terminates with an error
2. Create a new stack frame
 - A frame has bindings from variables to values
 - Looking up a variable starts here
 - Proceeds to the next older frame if no match here
 - The oldest frame is the “global” frame
 - All the frames together are called the “environment”
 - Assignments happen here
3. Assign the actual argument values to the formal parameter variables
4. Evaluate the body
 - At a return statement, remember the value and exit
 - If at end of the body, return **None**
5. Remove the stack frame
6. The call evaluates to the returned value

More examples of function calls

- Use the Python Tutor:
<http://people.csail.mit.edu/pgbovine/python/tutor.html>

The function can be an expression

```
def double(x):  
    return 2*x  
  
myfns = [math.sqrt, int, double, math.cos]  
myfns[1](3.14)  
myfns[2](3.14)  
myfns[3](3.14)  
  
def myfunc():  
    return double  
myfunc()(2.718)
```