CS 160
CS Orientation

Creating Functions...
What if we don’t have an isnumeric()?

```python
num_emp = input("enter num employees: ");

error = False; # we assume we don't have an error
# all characters in string must be 0-9 digits to be positive int
for x in range(len(num_emp)):
    if num_emp[x] < '0' or num_emp[x] > '9':
        print("You idiot! I said integer number!!");
        error = True; # We found an error!
        break; # quit looking at characters when one is found to be invalid

print(x); # what is the value of x after the loop
print(len(num_emp));
#if (not num_emp.isnumeric()):
#    print("You idiot! I said integer number!!");
#else:
if error == False: # if there wasn't an error, then valid
    # change to int, when you know it is valid
    num_emp = int(num_emp);  
    for x in range(len(num_emp)):
        hours = float(input("Employee #" + str(x+1) + " enter hour: "));
        pay = float(input("enter pay: "));
        gross = hours * pay;
        print(gross);
```

Pre-defined Functions

• May need to import a library
• Use the function from library/object
• Example:
  import math
  math.sqrt(4);
What is the purpose of a function?

• Perform some task!
  – May take input (arguments)
  – May produce output (print)
  – May return a value (return statement)
  – May alter input (change argument values if mutable)

• Example: \( y = f(x) = 3 \times x; \)
Python Function Syntax

def func_to_define(possible_parameters):
    print("hello");
    possible_parameters = 3;
    return "jennifer";

a=2;
name = func_to_define(a);
As a class, let’s define functions...

- For all the employees in our company, calculate their gross pay based on their hours and pay rate.
- Create a function to get number of employees.
- What if you want to make sure this function only returns a valid integer?
- Create function called gross_pay that takes the number of employees and calculates gross pay for each.
- Create a main function where the program begins.