CS 160
CS Orientation

More Lists
Chap. 4
 Implement more design...

```python
# Description: prompt and receive pay rate and hour info for
# the employees
# Parameters: list of employee pay rate, list of employee of
# hours, number of employees
# Returns: none
# Pre-conditions: there would be two empty lists for pay rate
# and hours
# Post-conditions: both lists would no longer be empty

# Define get_employee_info
def get_employee_info(p, h):
    # for all employees
    for x in range(len(p)):
        # prompt user for pay rate
        # read the pay rate for the employee
        p[x] = float(input("Enter pay for Employee "+str(x+1)+": "))
        # prompt user for hours
        # read the hours for the employee
        h[x] = float(input("Enter hours for Employee "+str(x+1)+": "))
```

Test functions with main()...

#Main function calls and tests other functions. Needs to be below the #definitions of the functions that it calls

def main()
    num_emp = get_num_employees()

    pay = [0] * num_emp
    hours = [0] * num_emp
    print(pay)
    print(hours)

    get_employee_info(pay, hours)
    print(pay)
    print(hours)
    main()  #Make sure to call your main function to start the program:
Why can we modify a list in a function?
How do we append info to a list?

\[ a = [0] \times 5 \]

Pay-hour:

\[ [10, 30, 30, 10] \]

pay-hour: \[ [0] \times [0] = 10 \]

\[ [1] \times [1] = 5 \]
Appending elements in a list...

```python
# Define get_employee_info
def get_employee_info(p, h):
    # for all employees
    for x in range(len(p)):
        # prompt user for pay rate
        # read the pay rate for the employee
        p[x] = float(input("Enter pay for Employee #"+str(x+1)+": "))
        # prompt user for hours
        # read the hours for the employee
        h[x] = float(input("Enter hours for Employee #"+str(x+1)+": "))

# Main function calls and tests other functions. Needs to be below the
# definitions of the functions that it calls
def main():
    num_emp = get_num_employees()
    pay = []
    hours = []

    # pay = [0] * num_emp  # This doesn't work for 2-d lists
    for x in range(num_emp):
        pay.append(0)
        hours.append(0)

    print(pay)
    print(hours)

    get_employee_info(pay, hours)
    print(pay)
    print(hours)

main()  # Make sure to call your main function to start the program:
```
2-d lists...

• What is it?
• How do we create this?
• How do we use this?
# Define get_employee_info
def get_employee_info(e):
    #for all employees
    for x in range(len(e)):
        #prompt user for pay rate
        #read the pay rate for the employee
        e[x][0] = float(input("Enter pay for Employee #"+str(x+1)+": "))
        #prompt user for hours
        #read the hours for the employee
        e[x][1] = float(input("Enter hours for Employee #"+str(x+1)+": "))
    #Main function calls and tests other functions. Needs to be below the
    #definitions of the functions that it calls
    def main():
        num_emp = get_num_employees()
        e = [] #First, create an empty list

        #Don't do this!!!!!!
        crazy = [[0,0]] * num_emp
        crazy[0][0] = 5
        print(crazy)

        for x in range(num_emp):
            e.append([0] * 2) #Append a 1-d list to each 1-d list item
            e[0][0] = 5
        print(e) |
        get_employee_info(e)
        print(e)

    main() #Make sure to call your main function to start the program:}
# Define get_employee_info
def get_employee_info(e):
    # for all employees
    for x in range(len(e)):
        # prompt user for pay rate
        # read the pay rate for the employee
        e[x][0] = float(input("Enter pay for Employee #"+str(x+1)+": "))
        # prompt user for hours
        # read the hours for the employee
        e[x][1] = float(input("Enter hours for Employee #"+str(x+1)+": "))
    # Main function calls and tests other functions. Needs to be below the
    # definitions of the functions it calls
    def main():
        num_emp = get_num_employees()
        e = []  # First, create an empty list

        # Don't do this!!!!!!!
        crazy = [[0, 0]] * num_emp
        crazy[0][0] = 5
        print(crazy)

        for x in range(num_emp):
            e.append([0, 0])  # Append a 1-d list to each 1-d list item
            e[0][0] = 5
            print(e)
            get_employee_info(e)
            print(e)

    main()  # Make sure to call your main function to start the program:)
# Define get_employee_info

def get_employee_info(e):
    #for all employees
    for x in range(len(e)):
        #prompt user for pay rate
        #read the pay rate for the employee
        e[x][0] = float(input("Enter pay for Employee #"+str(x+1)+": "))
        #prompt user for hours
        #read the hours for the employee
        e[x][1] = float(input("Enter hours for Employee #"+str(x+1)+": "))

    #Main function calls and tests other functions. Needs to be below the
    #definitions of the functions that it calls

def main():
    num_emp = get_num_employees()
    e = []  #First, create an empty list

    #Don't do this!!!!!!
    crazy = [[0,0]] * num_emp
    crazy[0][0] = 5
    print(crazy)

    for x in range(num_emp):
        e.append([])  #Append a 1-d list
        e[x].append(0)  #Append a zero for the pay as one list item
        e[x].append(0)  #Append a zero for the hour as another list item
    e[0][0] = 5
    print(e)
    get_employee_info(e)
    print(e)