1. Create a Payroll class that takes in number of hours worked and hourly rate in the constructor with a method named calculate that prints their total Gross Wages by multiplying total hours x hourly rate (assume no overtime).

Create test case to test the payroll class with values hours: 80 and rate: 10 the result should be 400.

Create a dictionary, name it payroll\_log, create three instance of the Payroll class, add them to the dictionary, and loop through the dictionary and call the calculate function for each.  
  
Sample output:  
  
Hours     Rate     Gross Pay  
40           10          400  
30           10          300  
20            10         200

2. Create a Shape class with a draw class function. Derive two classes , Line and Circle that inherit from Shape class .The Line class draw method/ function will display 'Draw a line' and the Circle draw method/ function will display 'Draw a Circle'.

Create a list named shape and add one instance of Line and one instance of circle. Loop through the list and…

Call the draw function for each shape to display the following:

Draw Line  
Draw Circle

3. Use loop of your choice to write a program that prompts user for a number three times, prints the sum of the numbers, and the sum plus 8% tax..

You must derive the tax and total numbers with multiplication and division.

Example output:

Total Tax Total  
10 .80 10.80

4. The following sample file called employee.dat contains one line for each employee. The employee’s name is the first item on each line, followed by some sales counts.

joe 10 15 20 30 40  
bill 23 16 19 22  
sue 8 22 17 14 32 17 24 21 2 9 11 17  
grace 12 28 21 45 26 10  
john 14 32 25 16 89

Write a program that averages each student's sales and display the name and sales average for each as follows:  
a. Create a get\_average function that takes a list of integers as a parameter and returns the average.  
b. Create a test class and test case to test the average for a list of numbers 10 15 20 30 40  
result should be 23.  
  
Example for one employee, your program must display all the employees.

Name Sales Avg  
joe 23

5. Write a program with two variables, value1 with a value of 10 and value2 with a value of ‘90’ (this is a string and must have the quotes around the number 90).   
  
Create a function named sum\_values with value1 and value2 as parameters that returns the sum of the two numbers.  
Create a test case that test sum\_values with value 10 and ‘90’ the result should be 100.  
In your program call the sum\_values function to display the result to screen.

6. Write a value returning function count\_odd\_even with a list of integers parameter to count the odd and even numbers in the list [1,2,3,10,13, 15,17,20, 21,22].

The function should return two values, the odd count and the even count.   
Create a test case to test the list [1,2,3,10,13, 15,17,20, 21,22] the result should be 6 and 4.  
Call the function and assign the two return values to variables odd and even.

Finally, display:

The list 1 2 3 10 13 15 17 20 21 22 has 6 odd numbers and 4 even numbers

7. Write the pseudocode for a program that prompts user for miles driven, total gasoline cost for trip, and calculates miles per gallon for the total trip.

8. Create the UML diagram for the following code:

**class** Customer:  
  
 **def** \_\_init\_\_(self, name, address, phone):  
  
 self.name = name  
 self.address = address  
 self.phone = phone  
  
**class** Car:  
  
 **def** \_\_init\_\_(self, make, model, year):  
  
 self.make = make  
 self.model = model  
 self.year = year  
  
**class** ServiceQuote:  
  
 **def** \_\_init\_\_(self, customer, car, parts\_cost, labor\_cost):  
  
 self.customer = customer  
 self.car = car  
 self.parts\_cost = parts\_cost  
 self.labor\_cost = labor\_cost  
  
 **def** get\_cost(self):  
  
 **return** self.parts\_cost + labor\_cost  
  
**class** MainApp:  
  
 **def** main(self, customer, car):  
  
 quote = ServiceQuote(customer, car, 10000, 5000)