**Let's look at those grades!**

Creating a program to compute statistics means that you won't have to whip out your calculator and manually crunch numbers. All you'll have to do is supply a new set of numbers and our program does all of the hard work.

This mini-project will give you some practice with functions, lists, and translating mathematical formulae into programming statements.

In order to use the scores in our program, we'll need them in a container, namely a list.

On the right, you'll see the grades *listed* (see what I did there). The data is anonymous to protect the privacy of the students.

**Print those grades**

As a refresher, let's start off by writing a function to print out the list of grades, one element at a time.

**Instructions**

**1.**

Define a function on line 3 called print\_grades with one argument, a list called grades\_input.

Inside the function, iterate through grades\_input and print each item on its own line.

After your function, call print\_gradeswith the grades list as the parameter.

Hint

Need help with your for loop?

numbers = [2, 3, 5, 8, 13] for n in numbers: print n

|  |
| --- |
| grades = [100, 100, 90, 40, 80, 100, 85, 70, 90, 65, 90, 85, 50.5]  def print\_grades(grades\_input):  for n in grades\_input:  print n    print\_grades(grades) |

**Review**

So far, you've created a helper function that will be used in the next sections.

You also have a solid handle on the concepts that we'll need to continue.

The next step in the creation of our grade statistics program involves computing the *mean* (average) of the grades.

Onwards.

# The sum of scores

Now that we have a function to print the grades, let's create another function to compute the sum of all of the test grades.

This will be super-helpful when we need to compute the average score.

I know what you're thinking, "let's just use the built-in sum function!" The built-in function would work beautifully, but it would be too easy.

Computing the sum manually involves computing a rolling sum. As you loop through the list, add the current grade to a variable that keeps track of the total, let's call that variable total.

**Instructions**

**1.**

On line 3, define a function, grades\_sum, that does the following:

* Takes in a list of scores, scores
* Computes the sum of the scores
* Returns the computed sum.

Call the newly created grades\_sumfunction with the list of grades and print the result.

Hint

To compute a rolling sum, create a variable total that's initialized to zero. Then, as you loop through the list of grades, add the current grade to total.

Avoid using sum as a variable name as it has a special meaning in Python!

|  |
| --- |
| grades = [100, 100, 90, 40, 80, 100, 85, 70, 90, 65, 90, 85, 50.5]  def grades\_sum(scores):  total = 0  for temp in scores:  total = total + temp  return total  print grades\_sum(grades) |

**Computing the Average**

The average test grade can be found by dividing the sum of the grades by the total number of grades.

Luckily, we just created an awesome function, grades\_sum to compute the sum.

**Instructions**

**1.**

Define a function, grades\_average, below the grades\_sum function that does the following:

* Has one argument, grades\_input, a list
* Calls grades\_sum with grades\_input
* Computes the average of the grades by dividing that sum by float(len(grades\_input)).
* Returns the average.

Call the newly created grades\_average function with the list of grades and **print the result**.

Hint

Your grades\_average function should use the built-in Python lenfunction and your grades\_sumfunction to compute the average.

Remember that integer division in Python will always result in an integer. We convert len(grades\_input) is a float so that the average is a float.

|  |
| --- |
| grades = [100, 100, 90, 40, 80, 100, 85, 70, 90, 65, 90, 85, 50.5]  def grades\_sum(scores):  total = 0  for score in scores:  total += score  return total  print grades\_sum(grades)  def grades\_average(grades\_input):  sum\_of\_grades = grades\_sum(grades\_input)  average = sum\_of\_grades / float(len(grades\_input))  return average  print grades\_average(grades) |

# The Variance

Let's see how the grades varied against the average. This is called computing the variance.

A very large variance means that the students' grades were all over the place, while a small variance (relatively close to the average) means that the majority of the students had similar grades.

**1.**

On line 18, define a new function called grades\_variance that accepts one argument, scores, a list.

First, create a variable average and store the result of calling grades\_average(scores).

Next, create another variable variance and set it to zero. We will use this as a rolling sum.

for each score in scores: Compute its squared difference: (average - score) \*\* 2 and add that to variance.

Divide the total variance by the number of scores.

Then, return that result.

Finally, after your function code, print grades\_variance(grades)

|  |
| --- |
| grades = [100, 100, 90, 40, 80, 100, 85, 70, 90, 65, 90, 85, 50.5]  def print\_grades(grades\_input):  for grade in grades\_input:  print grade  def grades\_sum(scores):  total = 0  for score in scores:  total += score  return total    def grades\_average(grades\_input):  sum\_of\_grades = grades\_sum(grades\_input)  average = sum\_of\_grades / float(len(grades\_input))  return average  def grades\_variance(scores):  average = grades\_average(scores)  variance = 0  for score in scores:  variance += (average - score) \*\* 2  return variance / len(scores)  print grades\_variance(grades) |

# Standard Deviation

Great job computing the variance! The last statistic will be much simpler: standard deviation.

The standard deviation is the square root of the variance. You can calculate the square root by raising the number to the one-half power.

**1.**

Define a function, grades\_std\_deviation, that takes one argument called variance.

return the result of variance \*\* 0.5

After the function, create a new variable called variance and store the result of calling grades\_variance(grades).

Finally print the result of calling grades\_std\_deviation(variance).

Hint

Remember, raising a number to an exponent involves using the exponentiation \*\* operator.

|  |
| --- |
| def grades\_std\_deviation(variance):  return variance\*\*0.5    variance = grades\_variance(grades)  print grades\_std\_deviation(variance) |

**Review**

You've done a great job completing this program.

We've created quite a few meaningful functions. Namely, we've created helper functions to print a list of grades, compute the sum, average, variance, and standard deviation about a set of grades.

Let's wrap up by printing out all of the statistics.

Who needs to pay for grade calculation software when you can write your own? :)

**Instructions**

**1.**

Print out the following:

* all of the grades
* sum of grades
* average grade
* variance
* standard deviation

Hint

You need to use all of the helper functions that you've created and print out their results.

|  |
| --- |
| grades = [100, 100, 90, 40, 80, 100, 85, 70, 90, 65, 90, 85, 50.5]  def print\_grades(grades\_input):  for grade in grades\_input:  print grade  def grades\_sum(scores):  total = 0  for score in scores:  total += score  return total    def grades\_average(grades\_input):  sum\_of\_grades = grades\_sum(grades\_input)  average = sum\_of\_grades / float(len(grades\_input))  return average  def grades\_variance(scores):  average = grades\_average(scores)  variance = 0  for score in scores:  variance = variance + (average - score)\*\*2  return variance / len(scores)  def grades\_std\_deviation(variance):  return variance\*\*0.5    variance = grades\_variance(grades)  print grades\_std\_deviation(variance)  print print\_grades(grades)  print grades\_sum(grades)  print grades\_average(grades)  print grades\_variance(grades) |