One of the greatest things you can do in computer programming is calculating. Seriously. It's awesome.
Take a look at the code below, it asks the user to enter the length and width of a rectangle, and then outputs the area:

```
print "---RECTANGLE AREA PROGRAM---"

print "Please enter the length of the rectangle:"
length = int (raw_input())

print "Please enter the width of the rectangle:"
width = int (raw_input())

area = length * width

print "The length of the rectangle is", length, "and the width is", width
print "The area is", area
```

```
======================== RESTART: C:/Python27/asdf.py ========================
---RECTANGLE AREA PROGRAM---
Please enter the length of the rectangle:
25
Please enter the width of the rectangle:
13
The length of the rectangle is 25 and the width is 13
The area is 325
```

Try to create the following programs:

1) A program that prompts the user for three numbers, then adds them all up and outputs the answer.
2) A program that prompts the user for 2 numbers, multiplies them, and then outputs the number.
3) A program that prompts the user for a price of an item, then outputs the total price with 13% tax added.
4) A program that prompts the user for their age, then asks them the number of days since their last birthday. The program the outputs how many days they have been alive.
5) A program that prompts the user for the base and height of a triangle, then outputs the area of the triangle.
6) A program that prompts the user for four course marks, then outputs their average.

You will need to figure out what data needs to be entered in by the user for the next ones:

7) A program that calculates and outputs the area of a circle based on data given by the user.
8) A program that calculates and outputs the volume of a cube.
9) A program that calculates the speed in km/h of a runner in a 100 m race.
10) A program that calculates the volume of a rectangular prism.

Now it's time to get creative. Think of a program that you would like to write that involves calculating something that people need to know quite frequently. Imagine it sort of as an app.

Create that program.

---

You might need this for the programs above:

## Mathematical Operators [edit]

Here are some commonly used mathematical operators

| Syntax | Math | Operation Name |
|---|---|---|
| `a+b` | $a + b$ | addition |
| `a-b` | $a - b$ | subtraction |
| `a*b` | $a \times b$ | multiplication |
| `a/b` | $a \div b$ | division (see note below) |
| `a//b` | $\lfloor a \div b \rfloor$ | floor division (e.g. 5//2=2) - Available in Python 2.2 and later |
| `a%b` | $a \mod b$ | modulo |
| `-a` | $-a$ | negation |
| `abs(a)` | $\lvert a \rvert$ | absolute value |
| `a**b` | $a^b$ | exponent |
| `math.sqrt(a)` | $\sqrt{a}$ | square root |

**FOR LOOPS**

FOR LOOPS are a Kind of loop that can be used when a programmer knows exactly how many times they would like code to be repeated.

In a FOR LOOP the programmer makes a counter (which is usually named i or j or k) and then sets a range of values for the counter to loop through
They also set the increment or decrement (does the loop go up by 2, or up by 7, or down by 3, etc)

---

Type the following code and see how the program runs:

```
for i in range (1, 11):
    print "So Cool!"
```

---

In the code above, i is actually a variable, so you can output the value of i to the screen as well.
Try out the following code and see what happens:

```
for i in range (1, 201):
    print i
```

---

We can also have loops that increment by other numbers, not just 1.
Try out the loop below, it counts from 1 to 1000 but counts by 50.

```
for i in range (1, 1000, 50):
    print i
```

---

We can also have loops count backwards.
Try the following:

```
for i in range (50, 0, -1):
    print i
```

---

Try the following loop, it counts from 200 to 100, backwards by 20:

```
for i in range (200, 100, -20):
    print i
```

---