Section 3: Console Programming

Links to Section 3 Code:

Problem: Planetary Weight Calculator

Milestone #1: Mars Weight

A few years ago, NASA made history with the first controlled flight on another planet. Its latest Mars Rover, *Perseverance*, has onboard a 50cm high helicopter called *Ingenuity*. *Ingenuity* made its **third flight**, during which it flew faster and further than it had on any of its test flights on Earth. Interestingly, *Ingenuity* **uses Python** for some of its flight modeling software!
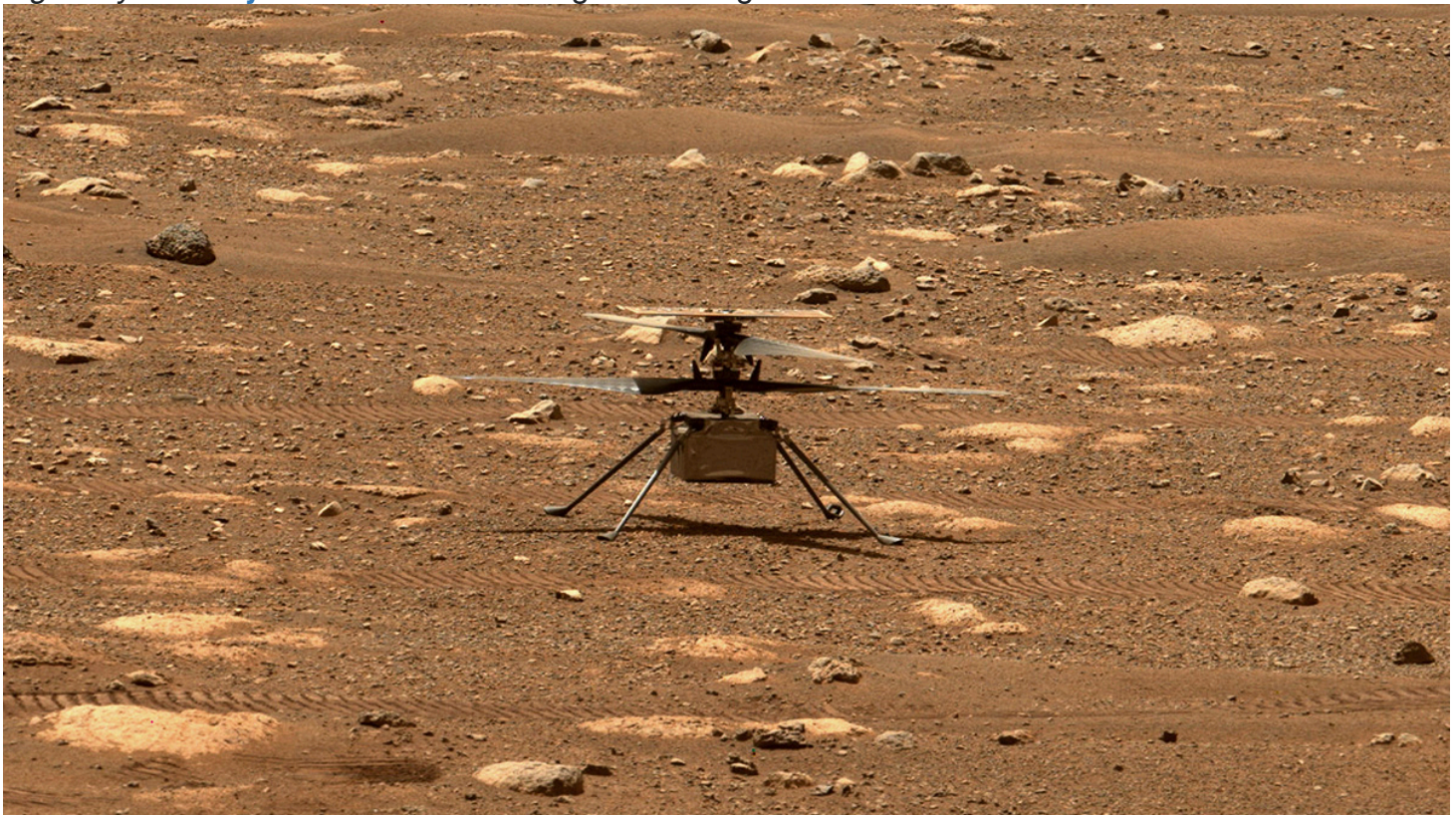
Ingenuity on the surface of Mars (source: NASA)

When programming *Ingenuity*, one of the things that NASA engineers need to account for is the fact that due to the weaker gravity on Mars, an Earthling's weight on Mars is 37.8% of their weight on Earth. Write a Python program that prompts an Earthling to enter their weight on Earth and prints their calculated weight on Mars.

The output should be rounded to two decimal places when necessary. Python has a round function which can help you with this. You pass in the value to be rounded and the number of decimal places to use. In the example below, the number 3.1415926 is rounded to 2 decimal places which is 3.14.

```python
x = 3.1415926
rounded_x = round(x, 2) # rounds x to 2 decimal places
print(rounded_x)

# This would print out out 3.14

x = 2.71828
rounded_x = round(x, 4)
print(rounded_x)

# This would print out 2.7183

x = 3
```

```
rounded_x = round(x, 4)
print(rounded_x)

# This would print 3
# Note that the round function does not add on decimal places that are not there already
```

Sample Run

```
$ python marsweight.py

Enter a weight on Earth: 120

The equivalent on Mars: 45.36
```

Sample Run

```
$ python marsweight.py

Enter a weight on Earth: 186

The equivalent on Mars: 70.31
```

Milestone #2: Adding in All Planets

Mars is not the only planet in our solar system with its own unique gravity. In fact, each planet has a different gravitational constant, which affects how much an object would weigh on that planet. Below is a list of the constants for each planet compared to Earth's gravity:
- Mercury: 37.6%
- Venus: 88.9%
- Mars: 37.8%
- Jupiter: 236.0%
- Saturn: 108.1%
- Uranus: 81.5%
- Neptune: 114.0%

Write a Python program that prompts an Earthling to enter their weight on Earth and then to enter the name of a planet in our solar system. The program should print the equivalent weight on that planet rounded to 2 decimal places if necessary.

You can assume that the user will always type in a planet with the first letter capitalized and you do not need to worry about the case where they type in something other than one of the above planets.

Sample Run

```
$ python planetaryweight.py

Enter a weight on Earth: 120

Enter a planet: Mars

The equivalent weight on Mars: 45.36
```

Sample Run

```
$ python planetaryweight.py

Enter a weight on Earth: 150

Enter a planet: Jupiter
```

```
The equivalent weight on Jupiter: 354.0
```

Useful Syntax

Python has an if statement, just like in Karel! This if statement passes if the value of x is the same as the value of y. x and y can be literal numbers, strings, variables, or constants.

```python
x = 42
y = 42

if x == y:
    print("x and y are equal!")
```