P4. leaky_relu function

Write a function leaky_relu to take 2 arguments: (1) a list of primary input values and (2) a leaky factor (as a floating-point number). The function then computes output values each corresponding to each primary input value as follows,

$$f(x) = \begin{cases} x & for \quad x \geq 0 \\ a \cdot x & for \quad x < 0 \end{cases},$$

where x is the primary input value and a is the leaky factor.

Use the P4 template. (P4_template.py; note: template is to ensure the exact display format and allows smooth auto-grading.)

Example 1:

When it is invoked by

```
r = leaky_relu(-10,0.005)
print(r)
```

We will see

```
================================================
-0.05
================================================
```

Example 2:

When it is invoked by

```
r = leaky_relu(-10, 0.05)
print(r)
r = leaky_relu(-2, 0.05)
print(r)
r = leaky_relu(0, 0.05)
print(r)
r = leaky_relu(2, 0.05)
print(r)
r = leaky_relu(10, 0.05)
print(r)
```

We will see

```
================================================
-0.5
-0.1
0
2
10
================================================
```

Here is P4_template.py

```
"""
Write a function leaky_relu to take 2 arguments:
(1) a list of primary input values and
(2) a leaky factor (as a floating-point number).
"""

# Write your function here

if __name__ == '__main__':

    r = leaky_relu(-10, 0.005)
    print(r)

    r = leaky_relu(-10, 0.05)
    print(r)

    r = leaky_relu(-2, 0.05)
    print(r)

    r = leaky_relu(0, 0.05)
    print(r)

    r = leaky_relu(2, 0.05)
    print(r)

    r = leaky_relu(10, 0.05)
    print(r)
```