Digital Root

THE CHALLENGE

Write a function that finds the digital root of a non-negative integer *n*.

More Details

Sum the digits of a non-negative integer *n*, then sum the digits of that sum and repeat this process until the sum has only one digit. This digit is called the digital root of *n*.

For example, 123456789 ⟶ 45 ⟶ 9.

What Your Function Should Do

Write a function `DigitalRoot` that takes in a non-negative number and returns its digital root.

```
DigitalRoot[36]
```

Out[1] = 9

More Examples

```
DigitalRoot[36]
```

Out[2] = 9

```
DigitalRoot[172]
```

Out[3] = 1

```
DigitalRoot[65 536]
```

Out[4] = 7

```
DigitalRoot[1 812 689 729 846 509 827 349 851 405 125]
```

Out[5] = 5

Things You May Find Useful

Digital Root

SCRATCH AREA

```
? Mod
```

Out[ ]=

Symbol

Mod[m, n] gives the remainder on division of m by n.

Mod[m, n, d] uses an offset d.

```
Mod[9, 45]
```

Out[ ]= 9

```
? DigitSum
```

Out[ ]=

Symbol

Global`DigitSum

Full Name Global`DigitSum

```
DigitSum[36, 9]
```

Out[ ]= DigitSum[36, 9]

ENTER YOUR CODE HERE

In[ ]:=

```
DigitalRoot[n_Integer] /; n ≥ 0 := 1 + Mod[n - 1, 9]
```

Submit