

Guide for Programming Assignment

Unit 4

There are 3 values we're concerned with:

1. The "value" The first argument of the `is_power` function
2. The Base Number The second argument of the `is_power` function
3. The "dividend" The quotient of the first and second arguments

Keep dividing the current value by the base number (which is 3 in this example) to get your dividend each time the function is run. When your dividend becomes equal to one, stop running the function and return True.

For example:

`is_power(27, 3)` # returns True

value ÷ base number = dividend

$27 \div 3 = 9$

$9 \div 3 = 3$

$3 \div 3 = 1$ # Stop running the function. Dividend equals one. Base case is hit. Return True!

Another example:

`is_power(10, 2)` # return False

$10 \div 2 = 5$

$5 \div 2 = 2.5$ # Not a whole number. Return False. Will never be equal to one.

$2.5 \div 2 = 1.25$

Any number to the power of 1 is equal to itself.

$2^1 = 2$ and $3^1 = 3$.

So it makes sense to check if the base number is equal to the value (if $2 == 2$: return True). If the value is not equal to the base number and the base number is equal to one, it needs to return False. The only power of one is one.

If the dividend is not a whole number or less than one, it needs to return False.

Python has a built-in `is_integer` method.

```
>>> 1.3.is_integer()
```

```
False
```

```
>>> 1.5.is_integer()
```

```
False
```

```
>>> 1.0.is_integer()
```

```
True
```

```
>>> 2.0.is_integer()
```

```
True
```

```
>>> 5.is_integer()
```

```
SyntaxError: invalid syntax
```