Challenge: Recursive Powers

Write a recursive function **power(x, n)** that returns the value of $\mathbf{x^n}$ (assume that **n** is an integer). Here are the 4 following cases that you need to handle.

1. Base Case

Start by writing the base case. $\mathbf{x^0 = 1}$ for any value of **x**.

2. Recursive case: n is odd

In this step, write the recursive case for which n is odd. Assume you have a function isOdd() to check if n is odd.

3. Recursive case: n is even

In this step, write the recursive case for which n is even. Assume you have a function isEven() to check if n is even.

4. Recursive case: n is negative

In this step, write the recursive case for which **n** is negative. Compute **x** raised to **-n** recursively, and return the reciprocal of that number.

Python | C++ | JS

```
def power(x, n):
  # base case

  # recursive case: n is negative

  # recursive case: n is odd

  # recursive case: n is even
  return None
```