RGB to HEX conversion

The rgb function is incomplete. Complete it so that passing in RGB decimal values will result in a hexadecimal representation being returned. Valid decimal values for RGB are 0 - 255. Any values that fall out of that range must be rounded to the closest valid value.

Note: Your answer should always be 6 characters long, the shorthand with 3 will not work here.

```
def rgb(r, g, b):
    # Helper function to clamp values within the range 0 to 255
    def clamp(value):
        return max(0, min(value, 255))

    # Helper function to convert a single value to its 2-digit h
    def to_hex(value):
        return f"{clamp(value):02X}"

    # Convert each component and concatenate
    return to_hex(r) + to_hex(g) + to_hex(b)

# Test cases
print(rgb(255, 255, 255))  # Should return "FFFFFF"
print(rgb(255, 255, 300))  # Should return "FFFFFF"
print(rgb(0, 0, 0))        # Should return "000000"
print(rgb(148, 0, 211))    # Should return "9400D3"
```

Cheated here

But a function is created to clamp the values from 0 to 255

max and min functions are used to return largest and smallest value among a given argument max function returns largest of the given arugments if you provide 2 arguments

max(20, 5) = will be 20

min function is vice versa, as it will return the smallest value

```
def clamp(x):
    return max(low, min(x, high))
```

essentially, if x is higher then high, then it will return high in the min bracket and then will be picked in the max bracket

if its low, min bracket will pick x, and x will be picked in the max bracket

```
def to_hex(value):
        return f"{clamp(value):02X}"
```

:02X is a format specifier

02 means zero padded, 2 digit wide, meaning if the number is less then 2 digit it will pad it with zeros

X means convert the number to uppercase hexadecimal

: is used to introduce format specifier