Euler 0030

The Problem:

Surprisingly there are only three numbers that can be written as the sum of fourth powers of their digits: 1634, 8208, 9474
We ignore 1 for the trivial case,lol.

Find the sum of all the numbers that can be written as the sum of fifth powers of their digits.

Considerations and Approach:

We need to find an upper limit first, which we can do by adding 9^5 to a sum until the sum is less than the number of digits.
There is no number that is going to fit these conditions above that upper limit.

Then we increment 1 to the upper limit and find all these multiples

The Code:

```
upper_limit = 9**5
digit_count = 1

#find the upper limit
while len(str(upper_limit))/digit_count > 1:
    upper_limit += 9**5
    digit_count += 1

print(upper_limit, digit_count)

#calculate every number between here and there
digit_fifths = []
for i in range(2, upper_limit):
```

```
    current_digit_sum = 0
    for j in [int(x) for x in str(i)]:
        current_digit_sum += j**5
    if current_digit_sum == i:
        digit_fifths += [i]

print(digit_fifths)
print(sum(digit_fifths))
```

Revision #2
Created 2026-02-09 20:35:51 UTC by Maxwell
Updated 2026-02-09 20:40:42 UTC by Maxwell