UK Postcodes matching a Reasonable Regex

```
def n_poss_postcodes_for_re():
    """
    Number of strings matching:
      ^[A-Z]{1,2}[0-9]{1,2}[A-Z]? [0-9][A-Z]{2}$
    """
    from letters import nL          # number of letters
    n_postal_areas = nL + nL * nL  # 1 or two letters
    n_postal_districts = 10 + 100  # Any one or two digit number
                                   # 0 and 0x aren't used, but match the regex
    n_subdistricts = nL + 1        # Not all letters are used,
                                   # and only for some London codes,
                                   # but for our regex...
                                   # The +1 is for ones not using a subdistrict

    n_outcodes = n_postal_areas * n_postal_districts * n_subdistricts

    n_incodes = 10 * nL * nL       # Digit then two letters

    n_postcodes = n_outcodes * n_incodes

    return n_postcodes


RE = r'^[A-Z]{1,2}[0-9]{1,2}[A-Z]? [0-9][A-Z]{2}$'
n = n_poss_postcodes_for_re()
n_formatted = f'{n:,}'
```

The number of "postcodes" (strings) matching the regular expression:

^[A-Z]{1,2}[0-9]{1,2}[A-Z]? [0-9][A-Z]{2}$

is 434,464,659,200.