**Stack Exchange Question**

I have 3 different strategies that are largely the same but have some minor differences and 2 of the strategies require some post processing. I want to create a strategy class like this:

Class Strategy:

def \_\_init\_\_(self, stock\_list, portfolio\_size, columns, url):

self.stock\_list = stock\_list 🡨 General, global (class) variable

self.portfolio\_size = portfolio\_size 🡨 General, global (class) variable

self.columns = columns 🡨 Different for each of the 3 types of strategy

self.url = url 🡨 Different for each of the 3 strategies

def setup(self, columns, url):

“””similar for each strategy but require different columns and urls”””

return dataframe

def postprocessing\_for\_strategy\_2(self, dataframe):

“””some dataframe postprocessing for strategy 2”””

return dataframe

def postprocessing\_for\_strategy\_2(self, dataframe):

“””some dataframe postprocessing for strategy 3”””

return dataframe

I am just curious of the best practise in this situation. I am more comfortable with the method I have in place but believe this breaks the DNRY paradigm since the setup in each case is similar. I think the use of kwarg’s is valid for the postprocessing since the post processing steps are completely different for each strategy.