**CODE USED IN NOTEBOOK 2 TO FILTER OUT INVESTMENT ENTITIES (vs. Officers and Directors)**

# Filter out entity filers (Investment entities that are not officers or directors)

filtered["RPTOWNERNAME"] = filtered["RPTOWNERNAME"].str.upper()

entity\_keywords = [

"LLC", "L L C", "L.L.C.", "LP", "L P", "L.P.", "LTD", "INC", "TRUST", "CORP",

"FOUNDATION", "COMPANY", "CO", "CO.", "PARTNERS", "ADVISORS", "ADVISORY",

"CAPITAL", "INVESTMENT", "INVESTMENTS", "HOLDINGS", "MGMT", "MANAGEMENT", "FUND",

"GROUP", "VENTURES", "BIOVENTURES", "INVESTORS", "EQUITY", "LIFE INSURANCE", "GP",

"FAMILY", "PBC", "SDN BHD", "GMBH"

]

# Create regex pattern that detects keywords even with leading punctuation or spacing

pattern = "(?i)" + "|".join(

r"(?<!\w)" + re.escape(k) + r"(?=\W|$)"

for k in entity\_keywords

)

# Save the rows that will be the filtered out entities (for later review)

# (temp comment out) filtered\_out\_df = filtered[filtered["RPTOWNERNAME"].str.contains(pattern, case=False, na=False, regex=True)]

filtered\_out\_df = before\_entity\_filter[before\_entity\_filter["RPTOWNERNAME"].str.contains(pattern, case=False, na=False, regex=True)].copy()

**Breakdown of the Regex Pattern:**

pattern = "(?i)" + "|".join(

r"(?<!\w)" + re.escape(k) + r"(?=\W|$)"

for k in entity\_keywords

)

“ marks the beginning and end of strings.

(?i) makes the entire regex case insensitive, so “LLC” will match “llc”

+ joins parts of the string into one full string.

"|".join( ) joins all individual patterns with "|", which is the regex **OR** operator

‘r’ ‘raw string’, prevents Python from interpreting the backslashes, e.g. as a newline in \n.

re.escape(k) escapes special characters in the keyword k so they are treated as literal characters. For example, if k = “L.L.C.”, it becomes “L\\.L\\.C\\.”, so the dots are not treated as any character.

(?<!\w) this is a negative “lookbehind” to ensure that what comes before the keyword is not a word character. This is needed for situations where the keyword could be misinterpreted to occur in the middle of another word. For example, a false match where the name PR**INC**ETON JOHN is interpreted as INC.

(?=\W|$) this is a “lookahead” to ensure that what comes after the keyword is a non-word character (\W like space, comma, dot) **OR** the end of a string ($). Similar to the above “lookbehind”, it ensures that they keyword is not embedded in the middle of another word (ensure that PR**INC**ETON JOHN is not interpreted as INC).

For k in entity keywords for each string k in entity\_keywords apply this regex formatting.