

LinkUp Step 2 Process LU-COMP link

November 18, 2025

0.1 Intro

This script processes U.S. job records and updates company identifiers based on shared URLs and company names.

The same firm may appear under multiple company_id values in the dataset while only some company_ids are covered by link table.

```
[1]: import re
import os
import pandas as pd
import numpy as np
import difflib
from tqdm import tqdm
from fake_useragent import UserAgent
import tarfile
import time
import datetime
import sqlite3
```

```
[2]: pd.set_option('display.max_colwidth', 100)
pd.set_option('display.max_rows', 2000)
pd.set_option('display.max_columns', 2000)
```

0.1.1 Compustat Company File

```
[3]: comp = pd.read_sas("WRDS/funda_identifier.sas7bdat", encoding="utf-8")
comp = comp[["gvkey", "datadate", "cusip"]]
comp["cusip_6"] = comp["cusip"].apply(lambda x: str(x)[0:6])
comp['datadate'] = comp['datadate'].dt.date
comp.head()
```

```
[3]:   gvkey  datadate   cusip cusip_6
0  001000  1964-12-31  000032102  000032
1  001000  1965-12-31  000032102  000032
2  001000  1966-12-31  000032102  000032
3  001000  1967-12-31  000032102  000032
4  001000  1968-12-31  000032102  000032
```

0.2 Full PIT Company Reference

```
[4]: LU_PIT = pd.read_csv('E:/LinkUp/Feeds/Raw Full PIT Company Reference/
↳raw_pit_company_reference_full_2022-05-31.csv');
LU_PIT.head(20)
```

C:\Users\Dan\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3437:
DtypeWarning: Columns (6) have mixed types.Specify dtype option on import or set
low_memory=False.

```
exec(code_obj, self.user_global_ns, self.user_ns)
```

```
[4]:
```

	company_id	start_date	end_date	company_name	\
0	1	2005-01-16	2016-02-24	Target	
1	1	2016-02-25	NaN	Target	
2	2	2005-01-16	2016-02-23	General Mills, Inc.	
3	2	2016-02-24	NaN	General Mills, Inc.	
4	3	2005-01-16	2016-02-23	Ecolab Inc.	
5	3	2016-02-24	NaN	Ecolab Inc.	
6	5	2005-01-16	2016-02-23	Wilsons Leather	
7	5	2016-02-24	NaN	Wilsons Leather	
8	6	2005-01-16	2016-02-23	Hawkins Chemical	
9	6	2016-02-24	NaN	Hawkins Chemical	
10	7	2005-01-16	2016-02-23	Zomax	
11	7	2016-02-24	NaN	Zomax	
12	8	2005-01-16	2022-01-09	RTW	
13	8	2022-01-10	NaN	RTW	
14	9	2005-01-16	2016-02-23	H.B. Fuller	
15	9	2016-02-24	2021-10-25	H.B. Fuller	
16	9	2021-10-26	NaN	H.B. Fuller	
17	10	2005-01-16	2022-01-09	Retek	
18	10	2022-01-10	NaN	Retek	
19	11	2005-01-16	NaN	TCF Financial Corporation	

	company_url	lei	open_perm_id	\
0	http://www.target.com	NaN	NaN	
1	http://www.target.com	8WDDFXB5T1Z6J0XC1L66	4295912282	
2	http://www.generalmills.com	NaN	NaN	
3	http://www.generalmills.com	2TGYMUGIO8P08X8L6150	4295904061	
4	http://www.ecolab.com	NaN	NaN	
5	http://www.ecolab.com	82DYEISM090VG8LTL526	4295903916	
6	http://www.wilsonsleather.com	NaN	NaN	
7	http://www.wilsonsleather.com	NaN	4296053169	
8	http://www.hawkinsinc.com	NaN	NaN	
9	http://www.hawkinsinc.com	549300VL3IJ230WX1Y34	4295906673	
10	http://www.zomax.com	NaN	NaN	
11	http://www.zomax.com	NaN	4295915504	
12	http://www.rtwi.com	NaN	NaN	
13	http://www.rtwi.com	NaN	NaN	

14	http://www.hbfuller.com	NaN	NaN
15	http://www.hbfuller.com	JSMX3Z5FOYM8V0Q52357	4295906546
16	http://www.hbfuller.com	JSMX3Z5FOYM8V0Q52357	4295906546
17	http://www.retek.com	NaN	NaN
18	http://www.retek.com	NaN	NaN
19	http://www.tcfexpress.com	NaN	NaN

	naics_code
0	NaN
1	452210.0
2	NaN
3	311999.0
4	NaN
5	325611.0
6	NaN
7	561499.0
8	NaN
9	541612.0
10	NaN
11	NaN
12	NaN
13	524126.0
14	NaN
15	NaN
16	325520.0
17	NaN
18	334614.0
19	NaN

0.3 Company Link Table

First step is to filter out firms that never operation in the US

```
[5]: LU_lnkhist = pd.read_csv('E:/LinkUp/Feeds/FS Daily Company Reference/
↳fs_company_reference_daily_2022-05-30.csv');
LU_lnkhist.head(20)
```

```
[5]:
```

	company_id	factset_entity_id	start_date	end_date	stock_ticker	\
0	1	002RXT-E	NaN	NaN	DYH	
1	1	002RXT-E	NaN	NaN	TGT	
2	1	002RXT-E	NaN	NaN	TGT.RM.NM	
3	1	002RXT-E	NaN	NaN	OLD8	
4	1	002RXT-E	NaN	NaN	TGTC	
5	1	002RXT-E	NaN	NaN	DYH	
6	1	002RXT-E	NaN	NaN	TGT	
7	1	002RXT-E	NaN	NaN	TGT	
8	1	002RXT-E	NaN	NaN	DYH	

9	1	002RXT-E	1972-01-21	2000-01-31	DH
10	1	002RXT-E	2000-01-31	NaN	TGT
11	2	000KYW-E	NaN	NaN	ORIX
12	2	000KYW-E	NaN	NaN	GRM
13	2	000KYW-E	NaN	NaN	GIS
14	2	000KYW-E	NaN	NaN	GIS
15	2	000KYW-E	NaN	NaN	GIS
16	2	000KYW-E	NaN	NaN	GRM
17	2	000KYW-E	1972-01-21	NaN	GIS
18	3	000HXM-E	NaN	NaN	ECON.XX9
19	3	000HXM-E	NaN	NaN	ECL

	stock_exchange_country	stock_exchange_name	isin	cusip	\
0	DE	ETR	US87612E1064	87612E106	
1	MX	MEX	US87612E1064	87612E106	
2	RU	MIC	US87612E1064	87612E106	
3	GB	LON	US87612E1064	87612E106	
4	AT	WBO	US87612E1064	87612E106	
5	IT	ETLX	US87612E1064	87612E106	
6	PE	LIM	US87612E1064	87612E106	
7	CL	SGO	US87612E1064	87612E106	
8	CH	BRN	US87612E1064	87612E106	
9	US	NYS	US87612E1064	87612E106	
10	US	NYS	US87612E1064	87612E106	
11	GB	LON	US3703341046	370334104	
12	IT	ETLX	US3703341046	370334104	
13	CH	SWX	US3703341046	370334104	
14	AT	WBO	US3703341046	370334104	
15	MX	MEX	US3703341046	370334104	
16	DE	ETR	US3703341046	370334104	
17	US	NYS	US3703341046	370334104	
18	US	OTC	US2790291023	279029102	
19	AT	WBO	US2788651006	278865100	

	sedol	primary_flag
0	5550469	False
1	B051WF6	False
2	BN4H331	False
3	BYZHG72	False
4	BKS49P1	False
5	BKKGTD3	False
6	BN4JNT3	False
7	B9FJJW9	False
8	BJ05580	False
9	2259101	True
10	2259101	True
11	BSJC7Q7	False

12	BGPK675	False
13	BRTM942	False
14	BFXPCM2	False
15	BGDT210	False
16	B12GMT5	False
17	2367026	True
18	NaN	False
19	BJK3KK1	False

1 Check all PIT reference if can match a link

Some firms does not have a link, but actually share the company website with a linked firm.

```
[6]: LU_lnkhist_temp = LU_lnkhist.copy()

[7]: LU_lnkhist_temp["Link_covered"] = 1
link_covered = LU_lnkhist_temp[["company_id", "Link_covered"]]
link_covered = link_covered.drop_duplicates()

[8]: LU_PIT2 = LU_PIT.merge(link_covered, how = "left", left_on = "company_id",
    ↪right_on = "company_id")
LU_PIT2.to_csv("LU_PIT2.csv")

[9]: LU_PIT2_covered = LU_PIT2[LU_PIT2["Link_covered"] == 1]
LU_PIT2_uncovered = LU_PIT2[LU_PIT2["Link_covered"].isna()]
print(len(LU_PIT2_covered))
print(len(LU_PIT2_uncovered))

124998
11285
```

1.0.1 Match uncovered firms to covered firms by URL

```
[10]: LU_PIT2_uncovered2 = LU_PIT2_uncovered.merge(LU_PIT2_covered, how = "left",
    ↪left_on = "company_url", right_on = "company_url")

LU_PIT2_uncovered2 = LU_PIT2_uncovered2[LU_PIT2_uncovered2["company_url"].
    ↪notna()]

LU_PIT2_uncovered2 = LU_PIT2_uncovered2.sort_values(by=['company_id_x',
    ↪'start_date_y'])
print(len(LU_PIT2_uncovered2))

LU_PIT2_uncovered3 = LU_PIT2_uncovered2.
    ↪drop_duplicates(subset=['company_id_x'], keep='first')
print(len(LU_PIT2_uncovered3))
```

```

LU_PIT2_uncovered3_1 = LU_PIT2_uncovered3[LU_PIT2_uncovered3["company_id_y"].
↳notna()]
print(len(LU_PIT2_uncovered3_1))

```

16687

9941

1248

1.0.2 Match uncovered firms to covered firms by company name

```

[11]: LU_PIT2_uncovered2 = LU_PIT2_uncovered.merge(LU_PIT2_covered, how = "left",
↳left_on = "company_name", right_on = "company_name")
LU_PIT2_uncovered2 = LU_PIT2_uncovered2[LU_PIT2_uncovered2["company_name"].
↳notna()]

LU_PIT2_uncovered2 = LU_PIT2_uncovered2.sort_values(by=['company_id_x',
↳'start_date_y'])
print(len(LU_PIT2_uncovered2))

LU_PIT2_uncovered3 = LU_PIT2_uncovered2.
↳drop_duplicates(subset=['company_id_x'], keep='first')
print(len(LU_PIT2_uncovered3))

LU_PIT2_uncovered3_2 = LU_PIT2_uncovered3[LU_PIT2_uncovered3["company_id_y"].
↳notna()]
print(len(LU_PIT2_uncovered3_2))

```

13332

10473

941

```

[12]: LU_PIT2_uncovered3 = LU_PIT2_uncovered3_1.append(LU_PIT2_uncovered3_2)
LU_PIT2_uncovered3 = LU_PIT2_uncovered3.sort_values(by=['company_id_x',
↳'start_date_y'])
LU_PIT2_uncovered3 = LU_PIT2_uncovered3.
↳drop_duplicates(subset=['company_id_x'], keep='first')
print(len(LU_PIT2_uncovered3))

```

1632

Create replacement table, replace some firms' company id by matched linked company's id that share the same company website.

```

[13]: LU_PIT2_uncovered4 = LU_PIT2_uncovered3[["company_id_x", "company_id_y"]]
LU_PIT2_uncovered4 = LU_PIT2_uncovered4.rename(index={"company_id_x":
↳"company_id", "company_id_y": "company_id_updated"})
print(len(LU_PIT2_uncovered4))
LU_PIT2_uncovered4.to_csv("Processed_Data/20220612companyid_updatetable.csv")

```

1632

Keep private firms and public firms that listed in the US

```
[26]: LU_lnkhist = pd.read_csv('E:/LinkUp/Feeds/FS Daily Company Reference/
↳fs_company_reference_daily_2022-05-30.csv');
LU_lnkhist.head(20)
print(len(LU_lnkhist))
LU_lnkhist2 = LU_lnkhist[(LU_lnkhist["stock_exchange_country"] == "US") |
↳(LU_lnkhist["stock_exchange_country"].isna())]
print(len(LU_lnkhist2))
```

172442

87647

1.0.3 Merge with Compustat

```
[27]: LU_lnkhist2["end_date"] = LU_lnkhist2["end_date"].replace({np.nan:"2099-12-31"})
LU_lnkhist2['start_date'] = LU_lnkhist2['start_date'].apply(lambda x:pd.
↳to_datetime(str(x), format='%Y-%m-%d')).dt.date
LU_lnkhist2['end_date'] = LU_lnkhist2['end_date'].apply(lambda x:pd.
↳to_datetime(str(x), format='%Y-%m-%d')).dt.date
LU_lnkhist2["cusip_6"] = LU_lnkhist2["cusip"].apply(lambda x:str(x)[0:6])
```

<ipython-input-27-7ba0dea2b7c8>:1: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
LU_lnkhist2["end_date"] =
LU_lnkhist2["end_date"].replace({np.nan:"2099-12-31"})
```

<ipython-input-27-7ba0dea2b7c8>:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
LU_lnkhist2['start_date'] = LU_lnkhist2['start_date'].apply(lambda
x:pd.to_datetime(str(x), format='%Y-%m-%d')).dt.date
```

<ipython-input-27-7ba0dea2b7c8>:3: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
LU_lnkhist2['end_date'] = LU_lnkhist2['end_date'].apply(lambda
x:pd.to_datetime(str(x), format='%Y-%m-%d')).dt.date
```

<ipython-input-27-7ba0dea2b7c8>:4: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using `.loc[row_indexer,col_indexer] = value` instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
LU_lnkhist2["cusip_6"] = LU_lnkhist2["cusip"].apply(lambda x:str(x)[0:6])
```

```
[29]: LU_lnkhist2.head(100)
```

```
[29]:
```

	company_id	factset_entity_id	start_date	end_date	stock_ticker	\
9	1	002RXT-E	1972-01-21	2000-01-31	DH	
10	1	002RXT-E	2000-01-31	2099-12-31	TGT	
17	2	000KYW-E	1972-01-21	2099-12-31	GIS	
18	3	000HXM-E	NaT	2099-12-31	ECON.XX9	
25	3	000HXM-E	1986-12-10	2099-12-31	ECL	
27	5	001VCC-E	1997-05-28	2008-07-22	WLSN	
28	5	001VCC-E	2008-07-22	2099-12-31	PRVU	
30	6	05NSST-E	1975-08-19	2099-12-31	HWKN	
31	7	089WBX-E	NaT	2099-12-31	NaN	
32	8	0011ZV-E	2001-05-24	2001-05-31	RTWIE	
33	8	0011ZV-E	2001-05-31	2002-11-25	RTWI	
34	8	0011ZV-E	2002-11-25	2002-12-24	RTWID	
35	8	0011ZV-E	2002-12-24	2007-12-17	RTWI	
38	9	000KT8-E	1972-01-21	2002-12-02	FULL	
39	9	000KT8-E	2002-12-02	2099-12-31	FUL	
40	10	002MMX-E	1999-11-18	2005-04-11	RETK	
45	10	002MMX-E	2005-04-12	2099-12-31	ORCL	
52	12	000XNB-E	1972-01-21	2000-10-30	TANT	
53	12	000XNB-E	2000-10-30	2099-12-31	TNC	
54	13	001R78-E	NaT	2099-12-31	FMOL	
62	13	001R78-E	1994-05-06	1998-11-20	STB	
63	13	001R78-E	1998-11-20	2001-02-27	FSR	
64	13	001R78-E	2001-02-27	2099-12-31	USB	
66	14	0016WN-E	2007-05-31	2008-10-29	NWA	
67	14	0016WN-E	2007-05-31	2008-10-30	NWA.XX9	
68	14	0016WN-E	2008-10-29	2008-10-30	NWACQ	
70	14	0016WN-E	2008-10-30	2099-12-31	DAL	
77	14	0016WN-E	2008-10-30	2099-12-31	DALRQ	
84	15	0030MN-E	1972-01-21	2000-08-21	NSP	
85	15	0030MN-E	2000-08-21	2099-12-31	XEL	
89	17	001RND-E	2005-10-17	2005-11-14	CHPWD	
90	17	001RND-E	2005-11-14	2099-12-31	CHRW	
93	18	0FB9D2-E	1990-10-03	2015-02-10	ATK	
94	18	0FB9D2-E	2015-02-10	2018-06-06	OA	
98	18	0FB9D2-E	2018-06-06	2099-12-31	NOC	
105	20	00147D-E	1992-10-28	2099-12-31	PDCO	
107	21	000VC4-E	1991-06-21	2003-03-27	RGIS	

108	21	000VC4-E	2003-03-27	2099-12-31	RGS
111	22	000HMM-E	1979-11-27	2099-12-31	DCI
113	23	00378K-E	2001-03-14	2007-11-08	CEN
114	23	00378K-E	2007-11-09	2018-04-30	CEN.XX1
116	23	00378K-E	2018-04-30	2099-12-31	CDAY
121	24	000HH9-E	1980-04-25	2099-12-31	DLX
127	26	000JH2-E	1987-08-20	2099-12-31	FAST
129	29	001BN6-E	1990-06-26	2017-03-06	ACAT
131	29	001BN6-E	2017-03-06	2099-12-31	TXT
136	30	066JSW-E	1995-07-31	2014-11-20	VVTV
137	30	066JSW-E	2014-11-20	2019-07-17	EVLV
138	30	066JSW-E	2019-07-17	2019-12-12	IMBI
139	30	066JSW-E	2019-12-12	2099-12-31	IMBI
141	31	000C74-E	1972-01-21	2099-12-31	APOG
142	32	0015MV-E	1996-02-08	2008-08-06	RCCC
145	32	0015MV-E	2014-02-21	2099-12-31	VZ
158	33	000LVQ-E	1985-08-15	2016-10-05	HTCH
160	33	000LVQ-E	2016-10-06	2099-12-31	TTDKY
165	33	000LVQ-E	2016-10-06	2099-12-31	TTDKF
167	34	0026CQ-E	1998-12-04	2000-01-10	AIRB
168	34	0026CQ-E	2000-01-10	2017-11-01	SCSS
169	34	0026CQ-E	2017-11-01	2099-12-31	SNBR
172	37	002YG6-E	2000-08-09	2007-01-29	MCDT
173	37	002YG6-E	2001-02-05	2001-02-08	MCDAV
174	37	002YG6-E	2001-02-08	2007-01-29	MCDTA
177	37	002YG6-E	2007-06-27	2007-07-02	BRCDD
178	37	002YG6-E	2007-07-02	2017-11-17	BRCD
179	37	002YG6-E	2018-04-04	2018-04-05	AVGO
187	37	002YG6-E	2018-04-05	2099-12-31	AVGO
191	38	06FS8C-E	2001-12-07	2006-04-25	LWSN
192	38	06FS8C-E	2006-04-25	2011-07-05	LWSN
194	39	000Q93-E	2002-09-25	2003-01-07	MTSCE
195	39	000Q93-E	2003-01-07	2021-04-06	MTSC
202	39	000Q93-E	2021-04-07	2099-12-31	APH
204	40	000C4F-E	2010-03-01	2010-03-29	ANLYD
205	40	000C4F-E	2010-03-29	2013-10-11	ANLY
207	43	000XVG-E	1994-11-04	2006-02-28	TCAM
212	43	000XVG-E	2014-07-03	2016-12-30	TFIFF
213	43	000XVG-E	2016-12-30	2020-02-13	TFIFF
214	43	000XVG-E	2020-02-13	2099-12-31	TFII
216	45	000ZH3-E	1994-08-11	2019-01-07	BBOX
219	47	0823JR-E	2000-08-11	2011-06-17	AMMD
220	47	0823JR-E	2011-06-17	2014-03-03	ENDP
221	47	0823JR-E	2011-06-17	2014-03-03	ENDP
222	47	0823JR-E	2016-03-31	2099-12-31	ENDP
223	47	0823JR-E	2016-03-31	2099-12-31	ENDP
225	48	000M4H-E	2013-09-30	2099-12-31	INVX

227	49	0023FC-E	2003-11-21	2018-02-05	BWLD
229	50	000HKB-E	1989-10-06	2099-12-31	DGII
231	51	000XXQ-E	2016-09-12	2099-12-31	TUTR
233	52	09ZRJ8-E	2019-04-09	2099-12-31	CNSL
234	53	0JTTJJ-E	NaT	2099-12-31	NaN
236	54	000LY1-E	1991-04-26	2001-10-29	IDX
237	54	000LY1-E	2001-10-29	2006-08-29	IDNX
239	54	000LY1-E	2006-08-29	2006-08-30	VISG
240	54	000LY1-E	2006-08-30	2011-07-25	ID
242	54	000LY1-E	2011-07-26	2014-04-04	SAFRY
249	54	000LY1-E	2011-07-26	2014-04-04	SAFRF
256	55	05HWCG-E	2003-12-24	2099-12-31	TOELF
257	55	05HWCG-E	2008-11-10	2099-12-31	TOELY
259	56	000DRC-E	1994-07-29	2004-08-12	CIMA
262	56	000DRC-E	2004-08-13	2011-10-14	CEPH
266	56	000DRC-E	2011-10-14	2099-12-31	TEVA

	stock_exchange_country	stock_exchange_name	isin	cusip	\
9	US	NYS	US87612E1064	87612E106	
10	US	NYS	US87612E1064	87612E106	
17	US	NYS	US3703341046	370334104	
18	US	OTC	US2790291023	279029102	
25	US	NYS	US2788651006	278865100	
27	US	OTC	US7413901080	741390108	
28	US	OTC	US7413901080	741390108	
30	US	NAS	US4202611095	420261109	
31	NaN	NaN	NaN	NaN	
32	US	NAS	US74974R2067	74974R206	
33	US	NAS	US74974R2067	74974R206	
34	US	NAS	US74974R2067	74974R206	
35	US	NAS	US74974R2067	74974R206	
38	US	NYS	US3596941068	359694106	
39	US	NYS	US3596941068	359694106	
40	US	NAS	US76128Q1094	76128Q109	
45	US	NYS	US68389X1054	68389X105	
52	US	NYS	US8803451033	880345103	
53	US	NYS	US8803451033	880345103	
54	US	OTC	US3210041030	321004103	
62	US	NYS	US9029733048	902973304	
63	US	NYS	US9029733048	902973304	
64	US	NYS	US9029733048	902973304	
66	US	NYS	US6672804084	667280408	
67	US	NYS	US6672811097	667281109	
68	US	OTC	US6672801015	667280101	
70	US	NYS	US2473617023	247361702	
77	US	OTC	US2473611083	247361108	
84	US	NAS	US98389B1008	98389B100	

85	US	NAS	US98389B1008	98389B100
89	US	NAS	US12541W2098	12541W209
90	US	NAS	US12541W2098	12541W209
93	US	NYS	US68557N1037	68557N103
94	US	NYS	US68557N1037	68557N103
98	US	NYS	US6668071029	666807102
105	US	NAS	US7033951036	703395103
107	US	NYS	US7589321071	758932107
108	US	NYS	US7589321071	758932107
111	US	NYS	US2576511099	257651109
113	US	NYS	US1567791009	156779100
114	US	NYS	US1567791009	156779100
116	US	NYS	US15677J1088	15677J108
121	US	NYS	US2480191012	248019101
127	US	NAS	US3119001044	311900104
129	US	NAS	US0396701049	039670104
131	US	NYS	US8832031012	883203101
136	US	NAS	US4524652066	452465206
137	US	NAS	US4524652066	452465206
138	US	NAS	US4524652066	452465206
139	US	NAS	US4524652066	452465206
141	US	NAS	US0375981091	037598109
142	US	NAS	US7819041077	781904107
145	US	NYS	US92343V1044	92343V104
158	US	NAS	US4484071067	448407106
160	US	OTC	US8723514084	872351408
165	US	OTC	JP3538800008	J82141136
167	US	NAS	US83125X1037	83125X103
168	US	NAS	US83125X1037	83125X103
169	US	NAS	US83125X1037	83125X103
172	US	NAS	US5800311020	580031102
173	US	NAS	US5800312010	580031201
174	US	NAS	US5800312010	580031201
177	US	NAS	US1116213067	111621306
178	US	NAS	US1116213067	111621306
179	US	NAS	US11135F1012	11135F101
187	US	NAS	US11135F1012	11135F101
191	US	NAS	US52078P1021	52078P102
192	US	NAS	US52078P1021	52078P102
194	US	NAS	US5537771033	553777103
195	US	NAS	US5537771033	553777103
202	US	NYS	US0320951017	032095101
204	US	NAS	US0326812072	032681207
205	US	NAS	US0326812072	032681207
207	US	NAS	US89385P1021	89385P102
212	US	NYS	CA87241L1094	87241L109
213	US	NYS	CA87241L1094	87241L109

214	US	NYS	CA87241L1094	87241L109
216	US	NAS	US0918261076	091826107
219	US	NAS	US02744M1080	02744M108
220	US	NAS	US29264F1066	29264F106
221	US	NAS	US29264F1140	29264F114
222	US	NAS	US29264F1066	29264F106
223	US	NAS	US29264F1140	29264F114
225	US	OTC	US4576471050	457647105
227	US	NAS	US1198481095	119848109
229	US	NAS	US2537981027	253798102
231	US	NAS	US72764Y1001	72764Y100
233	US	NAS	US2090341072	209034107
234	NaN	NaN	NaN	NaN
236	US	NAS	US4519061018	451906101
237	US	NAS	US4519061018	451906101
239	US	NYS	US50212A1060	50212A106
240	US	NYS	US50212A1060	50212A106
242	US	OTC	US7865841024	786584102
249	US	OTC	FR0000073272	F4035A557
256	US	OTC	JP3571400005	J86957115
257	US	OTC	US8891101029	889110102
259	US	NAS	US1717961056	171796105
262	US	NAS	US1567081096	156708109
266	US	NYS	US8816242098	881624209

	sedol	primary_flag	cusip_6
9	2259101	True	87612E
10	2259101	True	87612E
17	2367026	True	370334
18	NaN	False	279029
25	2304227	True	278865
27	2017440	True	741390
28	2017440	True	741390
30	2415594	True	420261
31	NaN	False	nan
32	2756668	True	74974R
33	2756668	True	74974R
34	2109121	True	74974R
35	2109121	True	74974R
38	2354664	True	359694
39	2354664	True	359694
40	2520517	True	76128Q
45	2661568	True	68389X
52	2883641	True	880345
53	2883641	True	880345
54	NaN	False	321004
62	2340124	True	902973

63	2314709	True	902973
64	2736035	True	902973
66	B1XJYL9	True	667280
67	NaN	False	667281
68	2642767	False	667280
70	B1W9D46	True	247361
77	2261708	False	247361
84	2648312	True	98389B
85	2614807	True	98389B
89	2116228	True	12541W
90	2116228	True	12541W
93	2017677	True	68557N
94	BVGC686	True	68557N
98	2648806	True	666807
105	2672689	True	703395
107	2730327	True	758932
108	2730327	True	758932
111	2276467	True	257651
113	2745202	True	156779
114	2745202	True	156779
116	BFX1V56	True	15677J
121	2260363	True	248019
127	2332262	True	311900
129	2047500	True	039670
131	2885937	True	883203
136	2934378	True	452465
137	BSPPYL4	True	452465
138	BK8QFM6	True	452465
139	BK72V50	True	452465
141	2046176	True	037598
142	2758363	True	781904
145	2090571	True	92343V
158	2449054	True	448407
160	2868529	False	872351
165	B00LPQ3	False	J82141
167	2320892	True	83125X
168	2320892	True	83125X
169	BF4HN46	True	83125X
172	2617107	False	580031
173	2722669	True	580031
174	2722669	True	580031
177	2422174	True	111621
178	2422174	True	111621
179	BD9WQP4	True	11135F
187	BDZ78H9	True	11135F
191	2814135	True	52078P
192	B13MMW5	True	52078P

194	2608985	True	553777
195	2608985	True	553777
202	2145084	True	032095
204	2032153	True	032681
205	2032153	True	032681
207	2871486	True	89385P
212	B39F103	False	87241L
213	BYPJP59	False	87241L
214	BYPJP59	False	87241L
216	2546977	True	091826
219	2619998	True	02744M
220	2612287	True	29264F
221	2612287	True	29264F
222	BJ3V905	True	29264F
223	BJ3V905	True	29264F
225	2461047	True	457647
227	2170309	True	119848
229	2269661	True	253798
231	2898968	True	72764Y
233	B07LSQ7	True	209034
234	NaN	False	nan
236	2456623	True	451906
237	2456623	True	451906
239	B1CWHZ4	True	50212A
240	B0V3Y61	True	50212A
242	B23WS79	False	786584
249	B065FV4	False	F4035A
256	B02LVL8	False	J86957
257	B3F3QK6	False	889110
259	2199296	True	171796
262	2185143	True	156708
266	2883878	False	881624

```
[31]: conn2 = sqlite3.connect(':memory:')

LU_lnkhist2.to_sql('LU_lnkhist2', conn2, index=False)
comp.to_sql('comp', conn2, index=False)
# Keep a wide matching range. As the start_date and end_date is about the
# website LinkUp get job posting from not about fiscal year.
# When a job_post_date is slightly after the recorded end_date, or before the
# startdate, still treat it as a valid match
qry = '''
SELECT
    b.*, a.*
FROM LU_lnkhist2 b
LEFT JOIN comp a
ON a.cusip_6 = b.cusip_6
```

```

        and ((b.start_date <= a.datadate and b.end_date >= a.datadate) or
        (b.end_date <= a.datadate and b.end_date >= a.datadate -365) or
        (b.start_date <= a.datadate + 365 and b.start_date >= a.datadate))

    '''
LU_lnkhist3 = pd.read_sql_query(qry, conn2)
LU_lnkhist3 = LU_lnkhist3.drop_duplicates(subset=['factset_entity_id',
↪ 'start_date', "stock_ticker", "stock_exchange_country", "cusip"],
↪ keep='first')

```

```
[32]: LU_lnkhist3.to_csv("Processed_Data/LU_lnkhist3.csv", index = "False")
```

```
[ ]:
```