Financial Statement Ratio Builder Write-up

Executive Summary

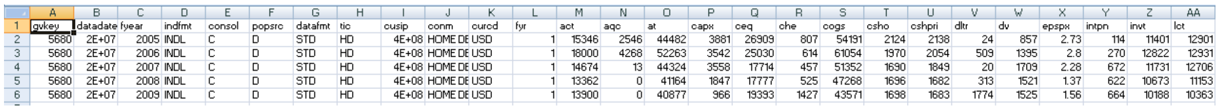
All Professional Stem (audit and business analysis) Masters of Accounting students at Brigham Young University are required to complete a course in financial statement analysis. The final project for that course is to present a valuation of a company of the student's choice. As part of this valuation, students are required to conduct a ratio analysis of the company and its respective industry. Data for this assignment is obtained via S&P's Compustat, and is found on the Wharton School of Business website. After the company- and industry-specific data is downloaded, students must manually manipulate the data to create 22 separate ratios relating to profitability, efficiency, leverage, cash flow, assets, and other. This process takes a long time, and – because of this – not every student will create all the ratios.

Having a program in Excel that can automatically create these ratios for both the company and its industry will save a lot of time and will also make available better information for the company's valuation. The program from this project does just that.

Improvement Needed

To begin the ratio analysis assignment, students are required to access a Compustat database (a database that collects financial information), select specific accounts and measurements for their chosen company and industry, and download the data sheets. You'll see below two screenshots of the downloaded data that are needed to create the company and industry ratios, respectively. Notice how the data needs improvement in categorization, alignment, subtotaling, sorting, filtering, isolation, and formatting. It's entirely possible to construct a ratio analysis from the data in its current form, but the process is laborious.

Company-specific data

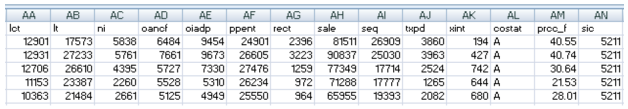


	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	gvkey	datadate	fyear	indfmt	consol	popsrc	datafmt	tic	cusip	conm	curcd	fyr	act	aqc
2	5680	2E+07	2005	INDL	C	D	STD	HD	4E+08	HOME DE	USD	1	15346	2546
3	5680	2E+07	2006	INDL	C	D	STD	HD	4E+08	HOME DE	USD	1	18000	4268
4	5680	2E+07	2007	INDL	C	D	STD	HD	4E+08	HOME DE	USD	1	14674	13
5	5680	2E+07	2008	INDL	C	D	STD	HD	4E+08	HOME DE	USD	1	13362	0
6	5680	2E+07	2009	INDL	C	D	STD	HD	4E+08	HOME DE	USD	1	13900	0

	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	at	capx	ceq	che	cogs	csho	cshpri	dltr	dv	epspx	intpn	invt	lct
2	44482	3881	26909	807	54191	2124	2138	24	857	2.73	114	11401	12901
3	52263	3542	25030	614	61054	1970	2054	509	1395	2.8	270	12822	12931
4	44324	3558	17714	457	51352	1690	1849	20	1709	2.28	672	11731	12706
5	41164	1847	17777	525	47268	1696	1682	313	1521	1.37	622	10673	11153
6	40877	966	19393	1427	43571	1698	1683	1774	1525	1.56	664	10188	10363

…

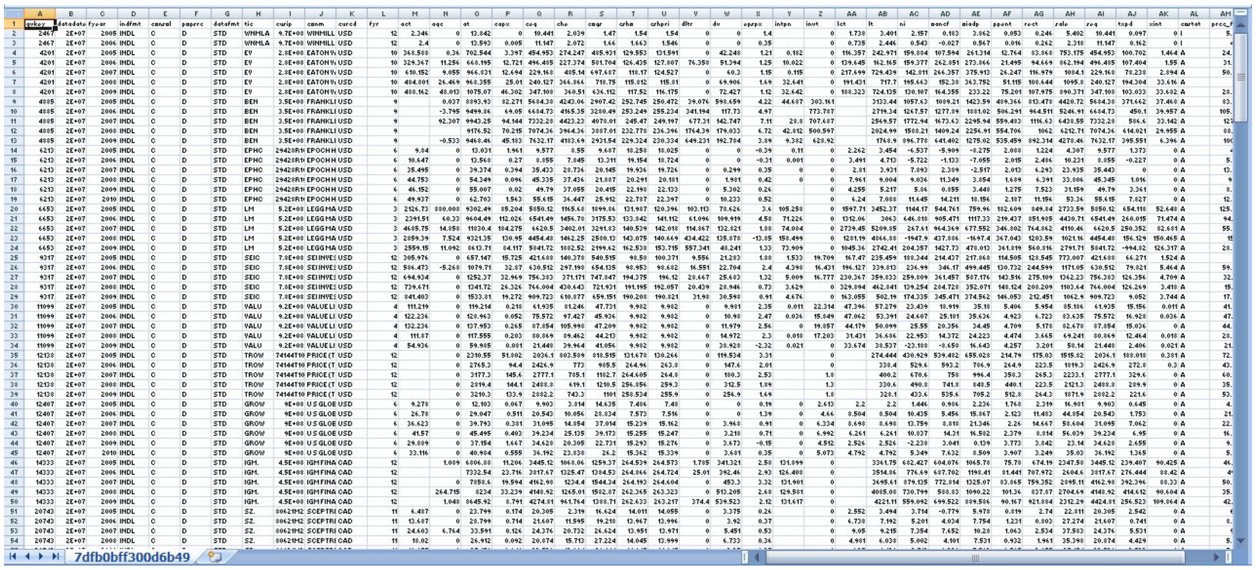
…



AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN
lct	lt	ni	oancf	oiadp	ppent	rect	sale	seq	txpd	xint	costat	prcc_f	sic
12901	17573	5838	6484	9454	24901	2396	81511	26909	3860	194	A	40.55	5211
12931	27233	5761	7661	9673	26605	3223	90837	25030	3963	427	A	40.74	5211
12706	26610	4395	5727	7330	27476	1259	77349	17714	2524	742	A	30.64	5211
11153	23387	2260	5528	5310	26234	972	71288	17777	1265	644	A	21.53	5211
10363	21484	2661	5125	4949	25550	964	65955	19393	2082	680	A	28.01	5211

(Notice how the first row contains the account or measurement name, followed by the data.)

Industry-specific (not all the data can fit in one screenshot)



	A	B	C	D	E	F	G	H	I	J	K	L
1	gvkey	datadate	fyear	indfmt	consol	popsrc	datafmt	tic	cusip	conm	curcd	fyr
2	2467	2E+07	2005	INDL	C	D	STD	WNMLA	9.7E+08	WINMILL	USD	12
3	2467	2E+07	2006	INDL	C	D	STD	WNMLA	9.7E+08	WINMILL	USD	12
4	4201	2E+07	2005	INDL	C	D	STD	EV	2.8E+08	EATON V	USD	10
5	4201	2E+07	2006	INDL	C	D	STD	EV	2.8E+08	EATON V	USD	10
6	4201	2E+07	2007	INDL	C	D	STD	EV	2.8E+08	EATON V	USD	10
7	4201	2E+07	2008	INDL	C	D	STD	EV	2.8E+08	EATON V	USD	10
8	4201	2E+07	2009	INDL	C	D	STD	EV	2.8E+08	EATON V	USD	10
9	4885	2E+07	2005	INDL	C	D	STD	BEN	3.5E+08	FRANKLI	USD	9
10	4885	2E+07	2006	INDL	C	D	STD	BEN	3.5E+08	FRANKLI	USD	9
11	4885	2E+07	2007	INDL	C	D	STD	BEN	3.5E+08	FRANKLI	USD	9
12	4885	2E+07	2008	INDL	C	D	STD	BEN	3.5E+08	FRANKLI	USD	9
13	4885	2E+07	2009	INDL	C	D	STD	BEN	3.5E+08	FRANKLI	USD	9
14	6213	2E+07	2005	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
15	6213	2E+07	2006	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
16	6213	2E+07	2007	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
17	6213	2E+07	2008	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
18	6213	2E+07	2009	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
19	6213	2E+07	2010	INDL	C	D	STD	EPHC	29428R1	EPOCH H	USD	6
20	6653	2E+07	2005	INDL	C	D	STD	LM	5.2E+08	LEGG MA	USD	3
21	6653	2E+07	2006	INDL	C	D	STD	LM	5.2E+08	LEGG MA	USD	3
22	6653	2E+07	2007	INDL	C	D	STD	LM	5.2E+08	LEGG MA	USD	3
23	6653	2E+07	2008	INDL	C	D	STD	LM	5.2E+08	LEGG MA	USD	3
24	6653	2E+07	2009	INDL	C	D	STD	LM	5.2E+08	LEGG MA	USD	3
25	9317	2E+07	2005	INDL	C	D	STD	SEIC	7.8E+08	SEI INVES	USD	12
26	9317	2E+07	2006	INDL	C	D	STD	SEIC	7.8E+08	SEI INVES	USD	12
27	9317	2E+07	2007	INDL	C	D	STD	SEIC	7.8E+08	SEI INVES	USD	12
28	9317	2E+07	2008	INDL	C	D	STD	SEIC	7.8E+08	SEI INVES	USD	12
29	9317	2E+07	2009	INDL	C	D	STD	SEIC	7.8E+08	SEI INVES	USD	12
30	11099	2E+07	2005	INDL	C	D	STD	VALU	9.2E+08	VALUE LI	USD	4
31	11099	2E+07	2006	INDL	C	D	STD	VALU	9.2E+08	VALUE LI	USD	4
32	11099	2E+07	2007	INDL	C	D	STD	VALU	9.2E+08	VALUE LI	USD	4
33	11099	2E+07	2008	INDL	C	D	STD	VALU	9.2E+08	VALUE LI	USD	4
34	11099	2E+07	2009	INDL	C	D	STD	VALU	9.2E+08	VALUE LI	USD	4
35	12138	2E+07	2005	INDL	C	D	STD	TROW	74144T10	PRICE (T	USD	12
36	12138	2E+07	2006	INDL	C	D	STD	TROW	74144T10	PRICE (T	USD	12
37	12138	2E+07	2007	INDL	C	D	STD	TROW	74144T10	PRICE (T	USD	12
38	12138	2E+07	2008	INDL	C	D	STD	TROW	74144T10	PRICE (T	USD	12
39	12138	2E+07	2009	INDL	C	D	STD	TROW	74144T10	PRICE (T	USD	12
40	12407	2E+07	2005	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
41	12407	2E+07	2006	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
42	12407	2E+07	2007	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
43	12407	2E+07	2008	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
44	12407	2E+07	2009	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
45	12407	2E+07	2010	INDL	C	D	STD	GROW	9E+08	U S GLOE	USD	6
46	14333	2E+07	2005	INDL	C	D	STD	IGM.	4.5E+08	IGM FINA	CAD	12
47	14333	2E+07	2006	INDL	C	D	STD	IGM.	4.5E+08	IGM FINA	CAD	12
48	14333	2E+07	2007	INDL	C	D	STD	IGM.	4.5E+08	IGM FINA	CAD	12
49	14333	2E+07	2008	INDL	C	D	STD	IGM.	4.5E+08	IGM FINA	CAD	12
50	14333	2E+07	2009	INDL	C	D	STD	IGM.	4.5E+08	IGM FINA	CAD	12
51	20743	2E+07	2005	INDL	C	D	STD	SZ.	80621M2	SCEPTRI	CAD	11
52	20743	2E+07	2006	INDL	C	D	STD	SZ.	80621M2	SCEPTRI	CAD	11
53	20743	2E+07	2007	INDL	C	D	STD	SZ.	80621M2	SCEPTRI	CAD	11
54	20743	2E+07	2008	INDL	C	D	STD	SZ.	80621M2	SCEPTRI	CAD	11

	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	act	aqc	at	capx	ceq	che	cogs	csho	cshpri	dltr	dv	epspx	intpn	invt
2	2.346	0	13.842	0	10.441	2.039	1.47	1.54	1.54	0	0	1.4		0
3	2.4	0	13.593	0.005	11.147	2.072	1.66	1.663	1.546	0	0	0.35		0
4	368.588	0.36	702.544	3.397	454.953	274.247	485.931	129.553	131.591	0	42.248	1.21	0.182	0
5	329.367	11.256	668.195	12.721	496.485	227.374	581.704	126.435	127.807	76.358	51.394	1.25	10.022	0
6	610.152	9.055	966.831	12.694	229.168	485.14	697.687	118.17	124.527	0	60.3	1.15	0.115	0
7	484.801	26.469	968.355	25.01	240.127	366.866	718.75	115.812	115.81	0	69.906	1.69	32.641	0
8	488.162	48.013	1075.07	46.302	347.108	360.51	636.112	117.52	116.175	0	72.427	1.12	32.642	0
9		0.037	8893.93	82.271	5684.38	4243.06	2907.42	252.745	250.472	39.076	598.659	4.22	44.687	303.161
10		-3.795	9499.86	69.05	6684.73	4165.35	3280.49	253.249	255.234	341.194	117.73	4.97		773.787
11		92.307	9943.25	94.144	7332.28	4423.23	4078.01	245.47	249.197	677.31	142.747	7.11	28.8	707.687
12			9176.52	70.215	7074.36	3964.36	3887.01	232.778	236.396	1764.39	179.033	6.72	42.812	500.597
13		-0.533	9468.46	45.183	7632.17	4183.69	2931.54	229.324	230.334	649.231	192.784	3.89	9.382	628.92
14	9.84	0	13.031	1.961	9.577	8.55	9.687	18.258	18.025	0	0	-0.39	0.11	0
15	10.647	0	13.568	0.27	8.855	7.845	13.311	19.154	18.724	0	0	-0.31	0.001	0
16	35.495	0	39.374	0.394	35.433	28.736	20.145	19.936	19.726	0	0.299	0.35	0	0
17	44.753	0	54.349	0.096	45.335	37.436	21.887	20.291	20.181	0	1.981	0.42	0	0
18	46.152	0	55.007	0.02	49.79	37.055	20.415	22.198	22.133	0	5.302	0.26		0
19	49.937	0	62.703	1.563	55.615	36.447	25.912	22.787	22.397	0	10.233	0.52		0
20	2126.73	880.008	9302.49	85.204	5850.12	1165.68	1899.86	131.987	120.396	103.113	78.626	3.6	105.258	0
21	2391.51	60.33	9604.49	112.026	6541.49	1456.78	3175.53	133.842	141.112	61.096	109.919	4.58	71.226	0
22	4685.75	14.858	11830.4	184.275	6620.5	3402.01	3291.83	140.539	142.018	114.867	132.821	1.88	74.084	0
23	2859.39	7.524	9321.35	130.95	4454.48	1462.25	2580.13	143.075	140.669	434.422	135.878	-13.85	158.499	0
24	2559.15	11.092	8613.71	84.117	5841.72	1882.52	2199.62	162.538	153.715	557.341	48.241	1.33	73.909	0
25	305.976	0	657.147	15.725	421.688	140.378	540.515	98.58	100.371	9.556	21.283	1.88	1.533	19.709
26	586.473	-5.268	1079.71	32.87	630.512	297.198	654.135	98.953	98.682	16.551	22.704	2.4	4.398	16.431
27	694.934	0	1252.37	32.969	756.383	371.171	747.847	194.375	196.12	28.667	25.683	1.32	5.009	16.777
28	739.671	0	1341.72	26.326	766.004	430.643	721.931	191.195	192.057	20.439	28.946	0.73	3.629	0
29	841.403	0	1533.81	19.272	909.723	610.877	659.151	190.208	190.821	31.98	30.598	0.91	4.676	0
30	111.219	0	119.214	0.218	61.935	81.246	47.731	9.982	9.982	0	9.981	2.35	0.011	22.314
31	122.236	0	128.963	0.052	75.572	97.427	45.936	9.982	9.982	0	10.98	2.47	0.036	15.849
32	132.236	0	137.953	0.265	87.854	105.998	47.209	9.982	9.982	0	11.979	2.56	0	19.857
33	111.87	0	117.555	0.203	80.869	89.462	44.213	9.982	9.982	0	14.972	2.3	0.018	17.203
34	54.936	0	59.985	0.081	21.448	39.964	41.056	9.982	9.982	0	38.928	-2.32	0.021	0
35		0	2310.55	51.802	2036.1	803.589	818.515	131.678	130.266	0	119.534	3.31		0
36		0	2765.3	94.4	2426.9	773	985.5	264.96	263.8	0	147.6	2.01		0
37		0	3177.3	145.6	2777.1	785.1	1182.7	264.605	264.8	0	180.3	2.53		1.8
38		0	2819.4	144.1	2488.8	619.1	1210.5	256.856	259.3	0	312.5	1.89		1.3
39		0	3210.3	133.9	2882.2	743.3	1101	258.534	255.9	0	256.9	1.69		1.8
40	9.278	0	12.103	0.067	9.903	3.814	14.635	7.486	7.48	0	0	0.19	0	2.613
41	26.78	0	29.047	0.511	20.543	10.056	28.834	7.573	7.516	0	0	1.39	0	4.66
42	36.623	0	39.793	0.381	31.095	14.854	37.014	15.239	15.162	0	3.968	0.91	0	6.334
43	41.57	0	45.495	0.403	39.234	25.135	39.173	15.255	15.247	0	3.218	0.71	0	6.992
44	29.889	0	37.154	1.667	34.628	20.305	22.731	15.293	15.276	0	3.673	-0.15	0	4.512
45	33.116	0	40.984	0.555	36.192	23.838	26.2	15.362	15.339	0	3.681	0.35	0	5.073
46		1.089	6806.88	11.206	3445.12	1068.06	1259.37	264.539	264.573	1.785	341.321	2.58	131.899	0
47			7332.54	23.716	3817.67	1325.47	1384.53	264.866	264.724	25.01	392.46	2.93	126.408	0
48		0	7858.6	19.594	4162.98	1234.4	1544.34	264.193	264.604	0	453.3	3.32	131.901	0
49		264.715	8234	33.239	4148.92	1265.01	1582.87	262.365	263.323	0	513.205	2.68	129.581	0
50		1.048	8645.92	8.791	4274.81	961.764	1388.71	262.633	263.217	374.4	539.523	2.12	131.617	0
51	6.487	0	23.799	0.174	20.305	2.319	16.624	14.011	14.055	0	3.375	0.26		0
52	13.687	0	28.799	0.714	21.607	11.595	19.218	13.967	13.996	0	3.92	0.37		0
53	24.603	6.764	33.591	0.126	24.376	20.732	26.624	13.951	13.971	0	5.451	0.53		0
54	18.02	0	26.912	0.092	20.874	15.713	27.224	14.045	13.999	0	6.733	0.36		0

	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM
1	lct	lt	ni	oancf	oiadp	ppent	rect	sale	seq	txpd	xint	costat	prcc_f
2	1.738	3.401	2.157	0.183	3.862	0.053	0.246	5.402	10.441	0.097	0	I	5.
3	0.735	2.446	0.543	-0.027	0.567	0.016	0.262	2.318	11.147	0.162	0	I	4
4	116.357	242.971	159.884	107.594	261.314	12.764	83.868	753.175	454.953	100.702	1.464	A	24.
5	139.645	162.165	159.377	262.851	273.866	21.495	94.669	862.194	496.485	107.404	1.55	A	31.
6	217.699	729.439	142.811	266.357	375.913	26.247	116.979	1084.1	229.168	78.238	2.894	A	50.
7	191.431	717.7	195.663	152.38	363.752	51.115	108.644	1095.8	240.127	194.304	33.616	A	
8	188.323	724.135	130.107	164.355	233.22	75.201	107.975	890.371	347.108	103.033	33.682	A	28.
9		3133.44	1057.63	1089.21	1423.59	489.366	813.478	4420.72	5684.38	371.662	37.468	A	83.
10		2719.34	1267.57	1277.89	1881.02	506.291	964.511	5246.91	6684.73	450.1	39.957	A	105.
11		2569.57	1772.94	1673.63	2295.94	559.483	1116.63	6438.55	7332.28	586.6	33.142	A	127
12		2024.99	1588.21	1409.24	2256.91	554.706	1062	6212.71	7074.36	614.021	29.955	A	88.
13		1768.9	896.778	641.402	1275.02	535.459	892.314	4278.46	7632.17	395.551	6.396	A	100
14	2.262	3.454	-6.537	-5.909	-8.275	2.088	1.224	4.307	9.577	1.373	0	A	4
15	3.491	4.713	-5.722	-1.133	-7.055	2.015	2.486	10.231	8.855	-0.227	0	A	5.
16	2.81	3.931	7.893	2.389	-2.517	2.013	6.293	23.935	35.443	0	0	A	13.
17	7.961	9.004	9.036	11.349	3.854	1.689	6.391	33.806	45.345	1.016	0	A	9
18	4.255	5.217	5.86	0.855	3.448	1.275	7.523	31.159	49.79	3.361	0	A	8.
19	6.24	7.088	11.645	14.211	18.156	2.187	11.156	53.36	55.615	7.827	0	A	12.
20	1597.71	3452.37	1144.17	544.761	759.96	182.609	849.84	2733.59	5850.12	654.118	52.648	A	125.
21	1312.06	3063	646.818	905.471	1117.33	219.437	851.985	4430.71	6541.49	260.015	71.474	A	94.
22	2739.45	5209.85	267.61	964.369	677.552	346.802	764.862	4110.46	6620.5	250.352	82.681	A	55.
23	1281.19	4866.88	-1947.9	437.886	-1697.4	367.043	1203.59	1021.16	4454.48	156.129	150.465	A	15
24	1045.36	2742.41	204.357	1427.73	478.013	361.819	560.816	2791.71	5841.72	-994.82	126.317	A	28.
25	167.47	235.459	188.344	214.437	217.868	114.505	128.545	773.007	421.688	66.271	1.524	A	
26	196.127	339.813	236.99	346.17	499.445	130.732	244.599	1171.05	630.512	79.821	5.464	A	59.
27	230.367	359.833	259.809	361.457	587.176	143.516	275.109	1362.23	756.383	126.356	4.709	A	32.
28	329.894	462.841	139.254	284.728	352.071	148.124	208.209	1103.64	766.004	126.269	3.418	A	15.
29	163.055	502.19	174.335	345.471	374.562	146.053	212.451	1062.9	909.723	9.052	3.744	A	17.
30	47.396	57.279	23.439	18.919	35.18	5.406	5.954	85.186	61.935	15.156	0.011	A	41.
31	47.062	53.391	24.607	25.181	35.636	4.923	6.723	83.635	75.572	16.928	0.036	A	47.
32	44.179	50.099	25.55	20.356	34.45	4.709	5.178	82.678	87.854	15.036	0	A	44.
33	31.431	36.686	22.953	14.372	24.223	4.474	3.665	69.241	80.869	12.464	0.018	A	28.
34	33.674	38.537	-23.188	-8.658	16.643	4.257	3.201	58.14	21.448	2.406	0.021	A	21.
35		274.444	430.929	539.482	655.028	214.79	175.03	1515.82	2036.1	188.018	0.381	A	72.
36		338.4	529.6	593.2	786.9	264.9	223.5	1819.3	2426.9	272.8	0.3	A	43.
37		400.2	670.6	758	996.4	358.3	265.3	2233.1	2777.1	329.6	0	A	60.
38		330.6	490.8	741.8	848.5	440.1	223.5	2121.3	2488.8	289.9	0	A	35.
39		328.1	433.6	535.6	705.2	512.8	264.3	1871.9	2882.2	221.6	0	A	53.
40	2.2	2.2	1.446	0.986	2.236	1.768	2.319	16.981	9.903	0.645	0	A	4.
41	8.504	8.504	10.435	5.456	15.867	2.123	11.483	44.854	20.543	1.753	0	A	21.
42	8.698	8.698	13.759	8.818	21.346	2.26	14.667	58.604	31.095	7.062	0	A	22.
43	6.261	6.261	10.837	14.31	16.582	2.379	8.814	56.039	39.234	6.95	0	A	16.
44	2.526	2.526	-2.238	3.041	0.139	3.773	3.842	23.14	34.628	2.655	0	A	9.
45	4.792	4.792	5.349	7.632	8.509	3.907	3.249	35.03	36.192	1.365	0	A	5.
46		3361.75	682.427	604.076	1065.78	75.78	674.19	2347.58	3445.12	239.407	90.425	A	46.
47		3514.86	776.69	687.702	1198.41	81.441	707.972	2604.6	3817.67	276.444	88.42	A	4
48		3695.61	879.135	772.814	1325.07	83.865	759.352	2895.11	4162.98	392.396	88.33	A	50.
49		4085.08	730.799	588.83	1090.22	101.36	837.87	2704.69	4148.92	414.612	90.604	A	35.
50		4221.11	559.092	699.522	889.586	90.167	921.884	2312.29	4424.81	256.523	109.864	A	42.
51	2.552	3.494	3.714	-0.779	5.978	0.819	2.74	22.811	20.305	2.542	0	A	6
52	6.738	7.192	5.201	4.034	7.754	1.231	0.803	27.274	21.607	0.741	0	A	8.
53	9.05	9.215	7.354	7.652	10.28	1.063	2.534	37.583	24.376	5.531	0	A	9
54	4.981	6.038	5.002	4.101	7.531	0.932	1.961	35.398	20.874	4.429	0	A	5.

7dfb0bff300d6b49

First Steps Toward Automation

Having already completed the ratio assignment without the use of this program, I was aware of a few quirks in the downloaded data that needed to be fixed. First, choosing which accounts needed to be included in the download was sometimes difficult to keep track of. In fact, the whole process from start to download was a bit choppy, and so I created an instruction sheet that lists everything one needs to do once they get to the Compustat website, including all instructions to finish up the calculations.

Next, once I had the data downloaded, I needed to ensure that there were not any blank cells where data needed to be. (Because Compustat has to provide standardized data for many different companies and industries, there will sometimes be certain accounts that are aggregated elsewhere. This is for the sake of comparison.) This was important because sometimes an automatic formula will only calculate up to a blank cell, and I didn't want to take any risks of this happening. Recording myself searching for blank cells ("") and replacing those cells with zeros remedied this. For the industry data sheet, I sorted the data by ascending fiscal year, and then subtotaled all the accounts/measurements. After automating sort, filter, and format functions, I was ready to create the ratio worksheet.

Building the Ratio Worksheet

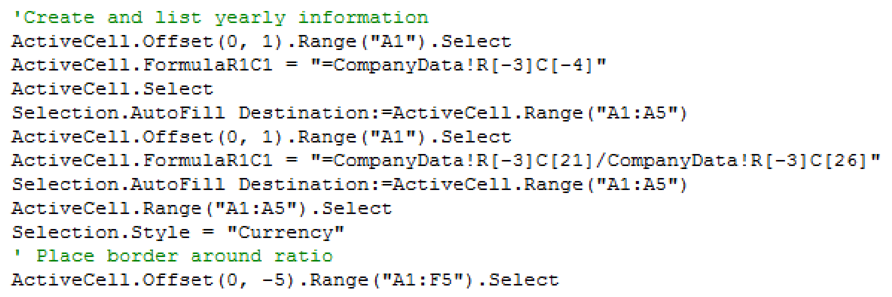
The program next adds another worksheet to the book and prepares it to manipulate data from the first sheet. What I failed to mention earlier was that all the information needed for the ratio assignment is found on a handout from the course professor. This handout includes what ratios are required for the assignment, how they're calculated, and their description (see below).

Summary of Financial Ratios

Ratio	Formula	WRDS Info[1]	Description
Profitability Ratios:			
Return on Sales	$\frac{\text{Net Income}}{\text{Sales}}$	$\frac{\text{NI}}{\text{SALE}}$	Number of pennies earned during the year on each dollar of sales.

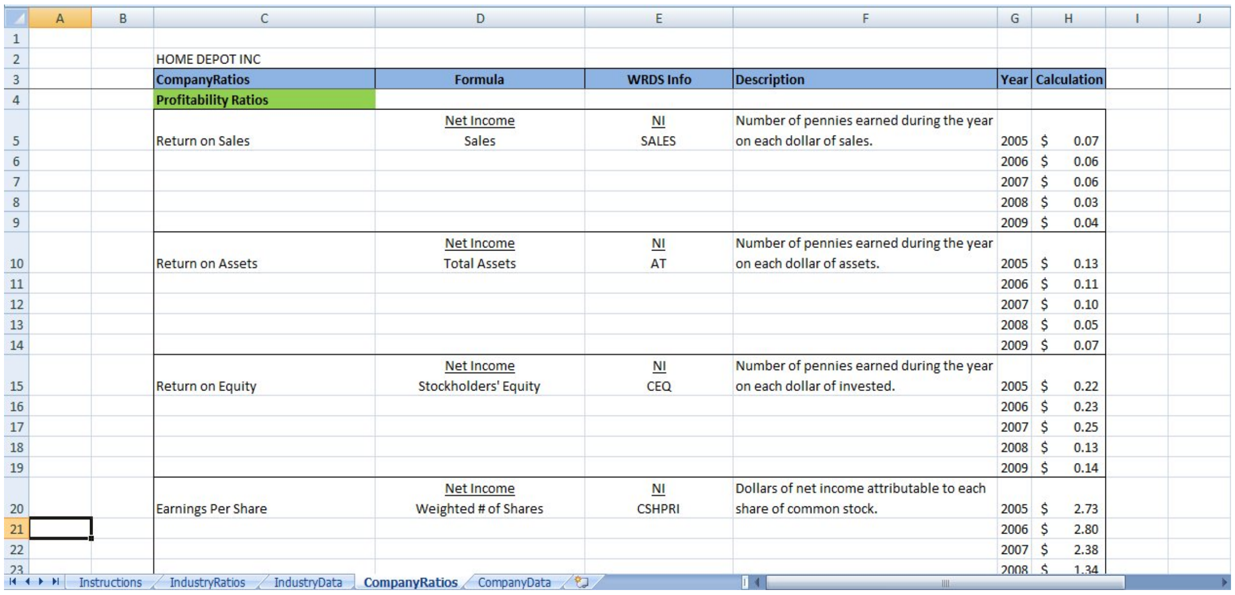
To maintain uniformity and not depart from the simplicity of the original assignment, I decided to format the ratio worksheet almost identically to the layout of the above handout. Again, recording myself writing the different column headings helped create somewhat of a template for other ratios and categories to follow. I was doing a lot of copying and pasting with the code, and it was nice to have an recorded example to reference. A very important point to mention is that the program I created does its best to use relative references in moving between cells and worksheets. This is important because the program has to be flexible when data of various sizes need manipulation. This is especially true for industry-specific data, which I will mention later. You'll notice in the code that nearly all references are R1C1, and not simply a Range("").

Finally, apart from formatting and adding borders, I was ready to start pulling in the data to create the ratios. The screenshot below shows a line of code that each ratio has. The company-specific data was relatively much easier to create because all I needed were the correct references to particular accounts/measurements, select the first year, and then auto-fill in the information until there was a blank cell on the data sheet. Working through the industry-specific ratios was much more difficult, and I'll discuss that later in the write-up.



```
'Create and list yearly information
ActiveCell.Offset(0, 1).Range("A1").Select
ActiveCell.FormulaR1C1 = "=CompanyData!R[-3]C[-4]"
ActiveCell.Select
Selection.AutoFill Destination:=ActiveCell.Range("A1:A5")
ActiveCell.Offset(0, 1).Range("A1").Select
ActiveCell.FormulaR1C1 = "=CompanyData!R[-3]C[21]/CompanyData!R[-3]C[26]"
Selection.AutoFill Destination:=ActiveCell.Range("A1:A5")
ActiveCell.Range("A1:A5").Select
Selection.Style = "Currency"
' Place border around ratio
ActiveCell.Offset(0, -5).Range("A1:F5").Select
```

To make the information as user-friendly as possible, I finished the ratio worksheet by aligning all the columns, wrapping the ratio "description" text, and freezing the top row of data for simple scrolling. What the user is left with is a spreadsheet that shows the name of the ratio, how the formula is calculated, the formula denominated in Compustat terms, a description of that ratio, the respective years being calculated, and the ratios themselves – conveniently and correctly formatted in currency, comma, or percentage (see below). What took at least an hour to complete now only took a few seconds. The information made available by the financial statement ratio builder could easily be converted into tables and graphs to the users' liking.



	A	B	C	D	E	F	G	H	I	J
1										
2			HOME DEPOT INC							
3			CompanyRatios	Formula	WRDS Info	Description	Year	Calculation		
4			Profitability Ratios							
5			Return on Sales	$\frac{\text{Net Income}}{\text{Sales}}$	$\frac{\text{NI}}{\text{SALES}}$	Number of pennies earned during the year on each dollar of sales.	2005	$ 0.07		
6							2006	$ 0.06		
7							2007	$ 0.06		
8							2008	$ 0.03		
9							2009	$ 0.04		
10			Return on Assets	$\frac{\text{Net Income}}{\text{Total Assets}}$	$\frac{\text{NI}}{\text{AT}}$	Number of pennies earned during the year on each dollar of assets.	2005	$ 0.13		
11							2006	$ 0.11		
12							2007	$ 0.10		
13							2008	$ 0.05		
14							2009	$ 0.07		
15			Return on Equity	$\frac{\text{Net Income}}{\text{Stockholders' Equity}}$	$\frac{\text{NI}}{\text{CEQ}}$	Number of pennies earned during the year on each dollar of invested.	2005	$ 0.22		
16							2006	$ 0.23		
17							2007	$ 0.25		
18							2008	$ 0.13		
19							2009	$ 0.14		
20			Earnings Per Share	$\frac{\text{Net Income}}{\text{Weighted \# of Shares}}$	$\frac{\text{NI}}{\text{CSHPRI}}$	Dollars of net income attributable to each share of common stock.	2005	$ 2.73		
21							2006	$ 2.80		
22							2007	$ 2.38		
23							2008	$ 1.34		

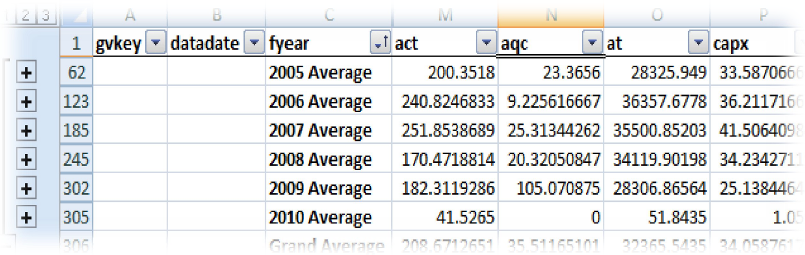
Learning and Conceptual Difficulties

This project was rife with learning and conceptual difficulties. Because I had not written quite an extensive block of code before, I had to overcome even the most elementary rules of writing a VBA program. Fortunately, recording myself creating different components of my planned program was very useful. By this I was able to easily write code for search and replace, auto-fill, name ranges, formatting, borders, and the calculations themselves.

However, the most difficult part of this project was figuring out a way to make it flexible for all years and especially for all sizes of data. By all years, I mean that the program would be able to take any five-year block of data and create the correct label on the ratio sheet. At first, I really worried about writing a user form-type program that would prompt the user for the years of data they wanted manipulated. I didn't want to do that, and so I realized that once I had the data in the Company-specific sheet sorted by year, I could simply tag the second row in that sheet and auto-fill down in the ratio sheet. It was this method that helped me learn more about relative references and their significance in flexible programs.

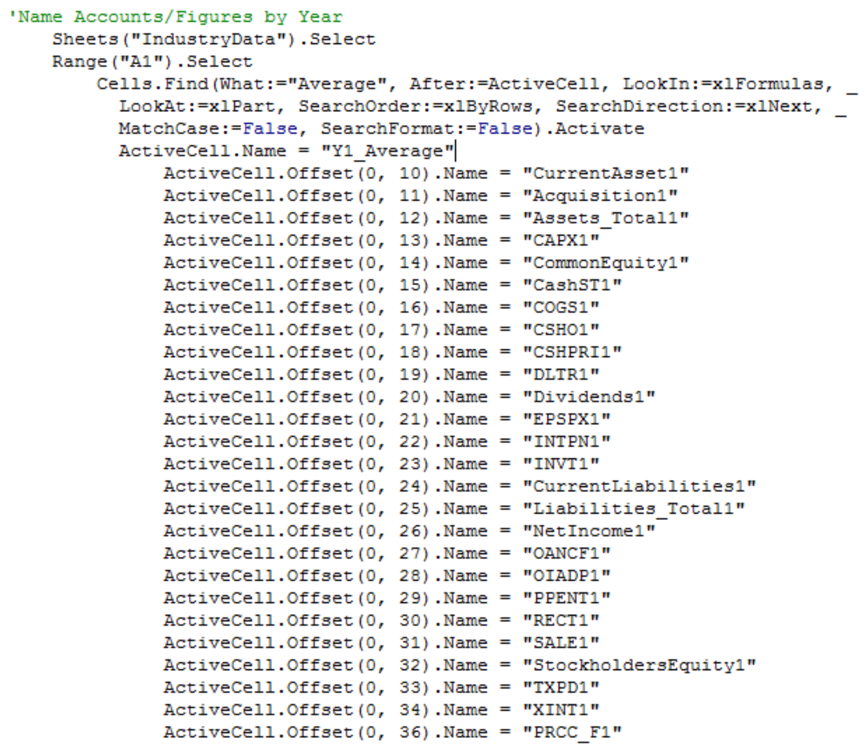
The problem that made me spend the most time pondering, testing, and writing was that of flexibly manipulating the industry-wide data. Finding amounts and years in the company-specific data was relatively easy. Because there are only five years downloaded for one company, the downloaded company spreadsheet only has five lines of data under the headers. This makes it really easy to relatively reference whatever you need, including the year, and auto-fill down. There is, however, a big difference in the industry data, and that is the number of companies in a respective industry. For example, Home Depot has only six competitors listed in its industry, whereas BlackRock (an investment management firm) has sixty competitors. This made it impossible to simply count down however many cells, grab a number or year, and then calculate a ratio. To fix the year problem, I knew that every person's analysis would start with their 'Year 1'. So, to provide an accurate year next to the ratio, I grabbed the first year on the data sheet (after sorting it by year), and then added 1, 2, 3, or 4 to the number down the row. What I had left to fix were the actual calculations.

In the process of remedying the calculation/relative reference problem, I spent a lot of time on one method that didn't work. I knew that I could easily find the yearly averages (subtotals) by conducting a search for "average". The search took me to the first average on the data sheet. If I wanted to go down to the next "average", I simply added a command to find the next "average" after the currently selected "average." After finding the first average, I named the range "Y1_Average", and so on until I named "Y5_Average." Now, all I [thought I] needed to do was relatively reference any cell in that row and send it to the calculation. My problem was this: I was already inside a formula code to write the ratio calculation. There may be a solution to this, but I could find no way to (1) find a cell, (2) move from that cell to others on the same row, and (3) make a calculation out of it … all while inside a formula command. I couldn't make it work without destroying the flexibility of the worksheet. Finally, I decided to carry out the one thing I didn't want to do.



	A	B	C	M	N	O	P
1	gvkey	datadate	fyear	act	aqc	at	capx
62			2005 Average	200.3518	23.3656	28325.949	33.587066
123			2006 Average	240.8246833	9.225616667	36357.6778	36.211716
185			2007 Average	251.8538689	25.31344262	35500.85203	41.506409
245			2008 Average	170.4718814	20.32050847	34119.90198	34.234271
302			2009 Average	182.3119286	105.070875	28306.86564	25.138446
305			2010 Average	41.5265	0	51.8435	1.05
306			Grand Average	208.6712651	35.51165101	32365.5435	34.058761

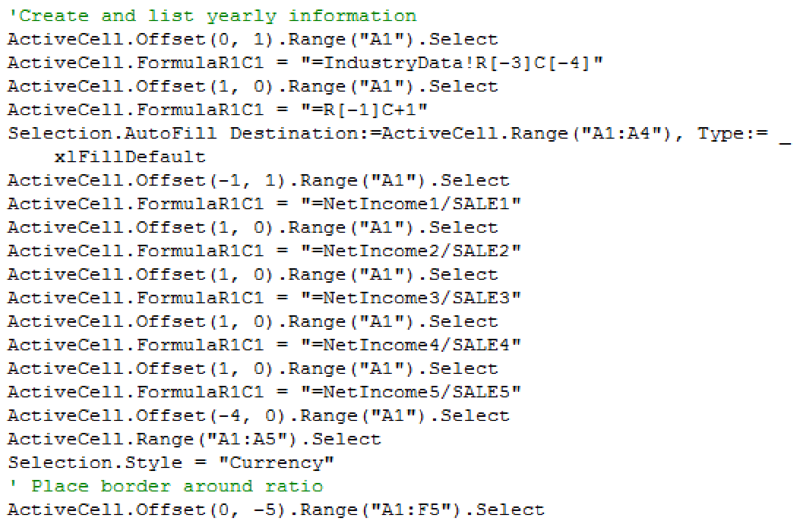
In order to identify which cells were which in their respective years, I had to name each individual reference by year. As mentioned before, I had no problem finding the right subtotal by moving to the cell that read "average." What I then had to do was move from left to right and have the program automatically name each range for each year. The screenshot below is an example of just one of those years. (Notice that each account/measurement is followed with a number '1', meaning that it is for the first year.



```
'Name Accounts/Figures by Year
    Sheets("IndustryData").Select
    Range("A1").Select
        Cells.Find(What:="Average", After:=ActiveCell, LookIn:=xlFormulas, _
          LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
          MatchCase:=False, SearchFormat:=False).Activate
          ActiveCell.Name = "Y1_Average"
              ActiveCell.Offset(0, 10).Name = "CurrentAsset1"
              ActiveCell.Offset(0, 11).Name = "Acquisition1"
              ActiveCell.Offset(0, 12).Name = "Assets_Total1"
              ActiveCell.Offset(0, 13).Name = "CAPX1"
              ActiveCell.Offset(0, 14).Name = "CommonEquity1"
              ActiveCell.Offset(0, 15).Name = "CashST1"
              ActiveCell.Offset(0, 16).Name = "COGS1"
              ActiveCell.Offset(0, 17).Name = "CSHO1"
              ActiveCell.Offset(0, 18).Name = "CSHPRI1"
              ActiveCell.Offset(0, 19).Name = "DLTR1"
              ActiveCell.Offset(0, 20).Name = "Dividends1"
              ActiveCell.Offset(0, 21).Name = "EPSPX1"
              ActiveCell.Offset(0, 22).Name = "INTPN1"
              ActiveCell.Offset(0, 23).Name = "INVT1"
              ActiveCell.Offset(0, 24).Name = "CurrentLiabilities1"
              ActiveCell.Offset(0, 25).Name = "Liabilities_Total1"
              ActiveCell.Offset(0, 26).Name = "NetIncome1"
              ActiveCell.Offset(0, 27).Name = "OANCF1"
              ActiveCell.Offset(0, 28).Name = "OIADP1"
              ActiveCell.Offset(0, 29).Name = "PPENT1"
              ActiveCell.Offset(0, 30).Name = "RECT1"
              ActiveCell.Offset(0, 31).Name = "SALE1"
              ActiveCell.Offset(0, 32).Name = "StockholdersEquity1"
              ActiveCell.Offset(0, 33).Name = "TXPD1"
              ActiveCell.Offset(0, 34).Name = "XINT1"
              ActiveCell.Offset(0, 36).Name = "PRCC_F1"
```

What was also a little frustrating about naming the ranges was that Excel already had claim on some of the most obscure collections of letters and numbers. That's why some of the names are actually spelled out. I wrote the above code for all five years.

Once I had all the ranges named, I then went back through my calculations and wrote in the correct references. Again, I could not simply auto-fill my first cell down to the fifth year. Each one had to be entered manually (as shown below).



```
'Create and list yearly information
ActiveCell.Offset(0, 1).Range("A1").Select
ActiveCell.FormulaR1C1 = "=IndustryData!R[-3]C[-4]"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "=R[-1]C+1"
Selection.AutoFill Destination:=ActiveCell.Range("A1:A4"), Type:= _
    xlFillDefault
ActiveCell.Offset(-1, 1).Range("A1").Select
ActiveCell.FormulaR1C1 = "=NetIncome1/SALE1"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "=NetIncome2/SALE2"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "=NetIncome3/SALE3"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "=NetIncome4/SALE4"
ActiveCell.Offset(1, 0).Range("A1").Select
ActiveCell.FormulaR1C1 = "=NetIncome5/SALE5"
ActiveCell.Offset(-4, 0).Range("A1").Select
ActiveCell.Range("A1:A5").Select
Selection.Style = "Currency"
' Place border around ratio
ActiveCell.Offset(0, -5).Range("A1:F5").Select
```

Having to complete this exercise of tagging each reference actually made the program much stronger because it doesn't have to rely so heavily on relative references. After fixing a few problems with border formatting and incorrect relative references, the program was complete. When I clicked 'play' for the program's maiden voyage as a completed work, an error message popped up and told me that the sub-procedure was too big. I was devastated, but only for a few seconds. I split the code into two separate codes, and the program ran fine.