

Receivables Detail – Town of Marblehead

Notes: This python scripts scrapes PDFs from a receivables details from this PDF and creates a csv output file with the following items: **Identifier, Bill Number, Total Charged, Total Paid, Interest Paid, Fees Paid, Abated/Exempt, and Balance.** The PDF filename is not unique and is static.

PDF

9/21/2016
5:12:40PM

Receivables Detail 2017

Town of Marblehead

Page 1 of 221

Real Estate as of 09/21/2016 for Unpaid Bills And ChargeType(s) P

Identifier	Bill Number	Owner Name	Total Charged	Total Paid	Interest Paid	Fees Paid	Abated/Exempt	Balance
110	7588	SUNBEAM DEV LTD PARTSHIP	15,959.02	0.00	0.00	0.00	0.00	15,959.02
120	7691	TEDESCO COUNTRY CLUB	50,985.60	25,492.80	0.00	0.00	0.00	25,492.80
1010	8699	ZISKIND MERTON	2,584.63	1,292.32	0.00	0.00	0.00	1,292.31
10110	7204	ALVAREZ-JIMENEZ JAVIER J	2,913.74	1,456.87	0.00	0.00	0.00	1,456.87
10150	8657	HEWIG WILLIAM III	186.48	93.24	0.00	0.00	0.00	93.24
10160	6259	PRICE PAUL W	3,389.94	1,694.97	0.00	0.00	0.00	1,694.97
1016A0	5832	STICKEVERS JOHN JOSEPH III	3,952.15	1,976.08	0.00	0.00	0.00	1,976.07
10170	7222	XAVIER MICHAEL G	2,848.26	1,424.13	0.00	0.00	0.00	1,424.13
10180	962	MILLER EILEEN O	3,396.60	1,698.30	0.00	0.00	0.00	1,698.30
10190	4029	KETOLA H NORMAN	3,512.04	1,756.02	0.00	0.00	0.00	1,756.02
1020	1642	FERRIS MARY R	2,313.24	1,156.62	0.00	0.00	0.00	1,156.62
10200	8658	HEWIG WILLIAM III	2,985.90	1,492.95	0.00	0.00	0.00	1,492.95
10210	2409	KUCHARSKY MIRA	3,183.48	1,591.74	0.00	0.00	0.00	1,591.74
10220	3850	KAIRYS LISE M	2,970.91	1,485.46	5.13	0.00	0.00	1,485.45
10240	8685	ZIEGLER PETER M	3,103.00	1,551.50	0.00	0.00	0.00	1,551.50
10250	8510	DIAWARA LASSANA	2,894.88	1,447.44	0.00	0.00	0.00	1,447.44
10260	4464	YOUNT SCOT	3,691.30	1,845.65	0.00	0.00	0.00	1,845.65
10270	2716	GEARHART GEORGE W CO TRSTEE	2,692.30	1,346.15	0.00	0.00	0.00	1,346.15
10280	5690	DOMBAL GREG E	2,927.07	1,463.54	0.00	0.00	0.00	1,463.53
10290	3917	FULLER ROBERT D	2,808.85	1,404.43	1.08	0.00	0.00	1,404.42
1030	3874	MCNERNEY KERRYANN	2,897.65	1,448.83	0.00	0.00	0.00	1,448.82
10300	3795	TRUDEL REMI	4,170.82	2,085.41	0.00	0.00	0.00	2,085.41
10310	4055	COEN KEVIN L	2,932.62	1,466.31	0.00	0.00	0.00	1,466.31
10320	3311	HASKIN CAROL J	3,461.53	1,730.77	0.00	0.00	0.00	1,730.76
10340	4289	FERREIRA JAMES	4,761.34	2,494.65	0.00	0.00	0.00	2,266.69
10350	3321	LIONE CHRISTOPHER J	3,635.25	1,817.63	0.00	0.00	0.00	1,817.62
10360	4978	MCCARTY BRETT C	6,740.47	3,370.24	0.00	0.00	0.00	3,370.23
10370	3540	HOOD NANCY PHALEN	3,638.02	1,819.01	0.00	0.00	0.00	1,819.01
10380	6700	ROMANOVSKIY MIKHAIL	2,687.86	1,343.93	0.00	0.00	0.00	1,343.93
10390	6160	GOLDSTEIN NANCY (PLYMPTON)	3,625.81	1,812.91	0.00	0.00	0.00	1,812.90
1039A0	5681	RAZIN BARRY L	3,252.85	1,626.43	0.00	0.00	0.00	1,626.42
1040	2228	ESSES DAVID J	2,580.19	1,290.10	10.89	0.00	0.00	1,290.09
10410	823	BROWN BARRY	3,094.68	1,547.34	0.00	0.00	0.00	1,547.34
10420	3820	SCOGLAND JEFFREY P	2,678.43	1,339.22	0.00	0.00	0.00	1,339.21
10430	5024	MCCLELLAND NICHOLAS W	3,080.80	1,540.40	0.00	0.00	0.00	1,540.40
10440	2571	BISHOP DALE A	2,736.15	1,368.08	0.00	0.00	0.00	1,368.07

Python Code (20190729_Marblehead.py)

```
# Read and dump PDF file data to text file for Town of Marblehead
# Accumatch
# by Michael Keller
# July 29, 2019
# requires pdfplumber, re, datetime

# Libraries
import pdfplumber
import re
from datetime import datetime

# Start Timer
start = datetime.now()

# Reg Expressions
regDateField = "^\\d{1,2}[\\./]\\d{1,2}[\\./]\\d{4}"
spaces = '\\s{2,}'

# Booleans
Date_Found = False
Identifier_Found = False
dateFound = False
identifierFound = False
bottomTotalsFound = False

# Variables
IdentifierString = 'Identifier'
BottomTotalsString = 'Real Estate Totals'
restOfLine = ''
identifier = ''
billNumber = ''
totalCharged = ''
totalPaid = ''
interstPaid = ''
feesPaid = ''
```

```
abatedExempt = ''
balance = ''
dollarSign = '$'
outputType = '.csv'
outputFile = 'TownOfMarblehead'
filename = ''
count = 0
outString = []
nospace = ''
space = ' '
comma = ','
_return = '\r'

colHeadings = 'Identifier,Bill Number,Total Charged,Total Paid,Interest Paid,Fees
Paid,Abated/Exempt,Balance'

# Messages
messageRun = 'Reading \'input.pdf\' file . . .'
messageSaveBeg = 'Saving \''
messageSaveEnd = '\' file . . .'
messageEOL = '. . . end of line'

# PDF File Open
pdf = pdfplumber.open("2017_Marblehead.pdf")

print(messageRun)

for page in pdf.pages:
    p1 = pdf.pages[count]
    count = count + 1
    pltext = p1.extract_text()
    text = pltext.splitlines()

## Break for Testing
## if(count == 5):
##     break

for item in text:
```

```

    Date_Found = re.search(regDateField,item)

    if not (Date_Found is None):

        dateFound = True

        blob = item.lstrip()

        if(blob[0:18] == BottomTotalsString):

            bottomTotalsFound = True

            if((identifierFound == True) and (dateFound == False) and (bottomTotalsFound == False)):

                carp1 = item.split(space,4)

                identifier = carp1[0] + space + carp1[1] + space + carp1[2]

                billNumber = carp1[3]

                restOfLine = carp1[4]

                restOfLine = restOfLine.replace(comma,nospace)

                restOfLine = restOfLine.replace(dollarSign,nospace)

                restOfLine = re.sub(spaces, space, restOfLine)

                carp2 = restOfLine.rsplit(space,6)

                totalCharged = carp2[1]

                totalPaid = carp2[2]

                interstPaid = carp2[3]

                feesPaid = carp2[4]

                abatedExempt = carp2[4]

                balance = carp2[6]

                outString.append(identifier + comma + billNumber + comma + totalCharged + comma +
totalPaid + comma + interstPaid + comma + feesPaid + comma + abatedExempt + comma + balance)

            if(item[0:10] == IdentifierString):

                identifierFound = True

                dateFound = False

# PDF File Close

pdf.close()

filename = outputFile + outputFile

print(messageSaveBeg + filename + messageSaveEnd)

# Create and open output file

try:

    file = open(filename,'w')

    # Write out lines from array

```

```
file.write(colHeadings + _return)
# Now dump string array into file
for taxLine in outString:
    file.write(taxLine + _return)
file.close()
except PermissionError:
    print('ERROR: Unable able to open output file for writing. Is the file currently open?')

#end timer
print('Runtime: ' + str(datetime.now() - start))

print(messageEOL)

# EOF
```

CSV Output

	A	B	C	D	E	F	G	H
1	Identifier	Bill Number	Total Charged	Total Paid	Interest Paid	Fees Paid	Abated/Exempt	Balance
2	1 1 0	7588	\$15,959.02	\$0.00	\$0.00	\$0.00	\$0.00	\$15,959.02
3	1 2 0	7691	\$50,985.60	\$25,492.80	\$0.00	\$0.00	\$0.00	\$25,492.80
4	10 1 0	8699	\$2,584.63	\$1,292.32	\$0.00	\$0.00	\$0.00	\$1,292.31
5	10 11 0	7204	\$2,913.74	\$1,456.87	\$0.00	\$0.00	\$0.00	\$1,456.87
6	10 15 0	8657	\$186.48	\$93.24	\$0.00	\$0.00	\$0.00	\$93.24
7	10 16 0	6259	\$3,389.94	\$1,694.97	\$0.00	\$0.00	\$0.00	\$1,694.97
8	10 16A 0	5832	\$3,952.15	\$1,976.08	\$0.00	\$0.00	\$0.00	\$1,976.07
9	10 17 0	7222	\$2,848.26	\$1,424.13	\$0.00	\$0.00	\$0.00	\$1,424.13
10	10 18 0	962	\$3,396.60	\$1,698.30	\$0.00	\$0.00	\$0.00	\$1,698.30
11	10 19 0	4029	\$3,512.04	\$1,756.02	\$0.00	\$0.00	\$0.00	\$1,756.02
12	10 2 0	1642	\$2,313.24	\$1,156.62	\$0.00	\$0.00	\$0.00	\$1,156.62
13	10 20 0	8658	\$2,985.90	\$1,492.95	\$0.00	\$0.00	\$0.00	\$1,492.95
14	10 21 0	2409	\$3,183.48	\$1,591.74	\$0.00	\$0.00	\$0.00	\$1,591.74
15	10 22 0	3850	\$2,970.91	\$1,485.46	\$5.13	\$0.00	\$0.00	\$1,485.45
16	10 24 0	8685	\$3,103.00	\$1,551.50	\$0.00	\$0.00	\$0.00	\$1,551.50
17	10 25 0	8510	\$2,894.88	\$1,447.44	\$0.00	\$0.00	\$0.00	\$1,447.44
18	10 26 0	4464	\$3,691.30	\$1,845.65	\$0.00	\$0.00	\$0.00	\$1,845.65
19	10 27 0	2716	\$2,692.30	\$1,346.15	\$0.00	\$0.00	\$0.00	\$1,346.15
20	10 28 0	5690	\$2,927.07	\$1,463.54	\$0.00	\$0.00	\$0.00	\$1,463.53
21	10 29 0	3917	\$2,808.85	\$1,404.43	\$1.08	\$0.00	\$0.00	\$1,404.42
22	10 3 0	3874	\$2,897.65	\$1,448.83	\$0.00	\$0.00	\$0.00	\$1,448.82
23	10 30 0	3795	\$4,170.82	\$2,085.41	\$0.00	\$0.00	\$0.00	\$2,085.41
24	10 31 0	4055	\$2,932.62	\$1,466.31	\$0.00	\$0.00	\$0.00	\$1,466.31
25	10 32 0	3311	\$3,461.53	\$1,730.77	\$0.00	\$0.00	\$0.00	\$1,730.76
26	10 34 0	4289	\$4,761.34	\$2,494.65	\$0.00	\$0.00	\$0.00	\$2,266.69
27	10 35 0	3321	\$3,635.25	\$1,817.63	\$0.00	\$0.00	\$0.00	\$1,817.62
28	10 36 0	4978	\$6,740.47	\$3,370.24	\$0.00	\$0.00	\$0.00	\$3,370.23
29	10 37 0	3540	\$3,638.02	\$1,819.01	\$0.00	\$0.00	\$0.00	\$1,819.01

Michael Keller
 Accumatch
 20190729_Receivables
 July 29, 2019