



---

# CYCLE-BASED BUDGETING TOOLKIT

---

Microsoft Word Investment Tracking Form Implementation Manual



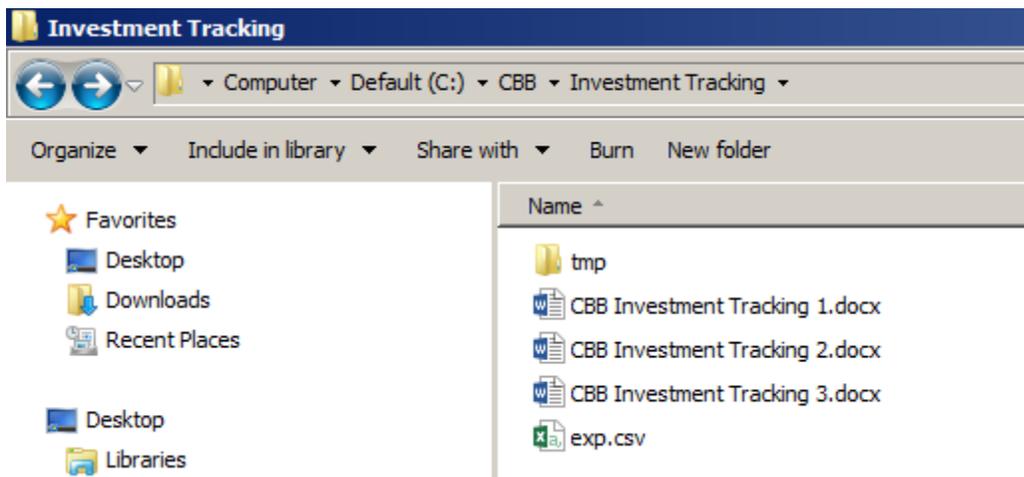
FEBRUARY 1, 2021

BO YAN

<http://www.cyclebasedbudgeting.org>

## STEP 1

After a filled-out [Investment Tracking Form](#) is returned, save it under the folder "C:\CBB\Investment Tracking". Eventually, all completed Investment Tracking Forms should be saved under this folder. It would help to develop a naming convention so that forms from different departments in different years can be easily differentiated. For example, all files can be named following the convention of "FY\_Cost Center\_Owner.docx" such as "2018-19\_Data Strategy\_Bo Yan.docx".



## STEP 2

In this step, an open-source tool called [R](#) is used to export data from all Investment Tracking Forms saved under the folder "C:\CBB\Investment Tracking". However, the code can be easily modified to export data for the desired forms only, such as forms from certain departments or years. Before running the code, you should have R and three R packages ("XML", "data.table", and "magrittr") installed on your computer. This blog post [Install R, RStudio, and R Commander in Windows and OS X](#) will be helpful if you need a step-by-step guide on how to do it.

```
#-----#
#
#           Cycle-based Budgeting Toolkit
#
# Developer: Bo Yan
# Email: support@cyclebasedbudgeting.org
# Created: 2018-01-23
#
# Description:
# 1. Export field data from the MS Word documents created using the CBB
#    Budget Request Form Template
# 2. Save the exported data into a .csv file
#
#-----#

library('XML')
library('data.table')
```

```

library('magrittr')

# set path where the submitted budget request forms are saved
setwd('c:/CBB/budget requests')

# create temp folder
if (!dir.exists('tmp')) {
  dir.create('tmp')
}

# retrieve submitted requests
docx <- list.files(pattern = '.docx')

# create empty exp data table
exp <- NULL

# unpack docx file
for (i in 1:length(docx)) {
  unzip(docx[i], files = "word/document.xml", exdir = './tmp')
  doc <- xmlParse('tmp/word/document.xml')
  root <- xmlRoot(doc)

  # retrieve bookmarks
  bms <- getNodeSet(doc, "//w:bookmarkStart")
  bme <- getNodeSet(doc, "//w:bookmarkEnd")

  # retrieveve field values
  wt <- xmlApply(bme, function(x) getSibling(x, after = F) %>% getSibling(., after =
F)) %>% sapply(., xmlValue)
  field <- xmlSApply(bms, xmlGetAttr, 'name')

  # add field name to create legagcy data table
  legacy <- data.table(var = field, value = wt) %>% .[grepl('[a-z]|[0-9]',
value)] %>% .[var != '_GoBack']

  # retrieve field values without bookmarks
  sdt <- getNodeSet(doc, "//w:sdtContent") %>% xmlSApply(., xmlValue)

  # create activex data table
  activex <- data.table(var = c('admDec', 'admRyr', 'saType', 'saCenter', 'saHead',
'saTitle', 'saDes', 'scPopStu', 'scPopSta', 'scPopPar', 'scSexF', 'scSexM', 'scRaceB',
'scLep', 'scRaceA', 'scLunch', 'scRaceH', 'scSped', 'scRaceW', 'scGift', 'scRace0'),
value = sdt[10:length(sdt)])
  activex[value == '\u{2612}', value := 'Y']
  activex[value == '\u{2610}', value := 'N']

  # create rd data table
  rd <- rbind(legacy, activex) %>% .[order(var)] %>%
  rbind(., data.table(var = 'saYear',
value = root[['body']][[8]] %>% xmlDoc %>% getNodeSet(.,
'//w:t') %>% .[[12]] %>% xmlValue()))

  # add id and file fields
  rd[, `:=`(id = i, file = docx[i])]

  # append rd to exp
  exp <- rbind(exp, rd)
}

```

```
write.csv(exp, file = 'exp.csv', row.names = F)
```

### STEP 3

After running the R code in Step 2, a CSV file called “exp.csv” will be generated under the folder where all returned Investment Tracking Forms are stored. There are four columns in this CSV file:

1. Column A “var”: variable name
2. Column B “value”: variable value
3. Column C “id”: unique ID number for an Investment Tracking Form
4. Column D “file”: name of the Investment Tracking Form file

	A	B	C	D	E
1	var	value	id	file	
2	admCyc		3	1 CBB Investment Tracking 1.docx	
3	admDec	Approve	1	1 CBB Investment Tracking 1.docx	
4	admRyr	20-21	1	1 CBB Investment Tracking 1.docx	
5	saCenter	Head 1	1	1 CBB Investment Tracking 1.docx	
6	saDes	Description 1	1	1 CBB Investment Tracking 1.docx	
7	saHead	Title 1	1	1 CBB Investment Tracking 1.docx	
8	saTitle	Recurrent	1	1 CBB Investment Tracking 1.docx	
9	saType	Cost Center 1	1	1 CBB Investment Tracking 1.docx	
10	sbOpeCat1	Hardware	1	1 CBB Investment Tracking 1.docx	
11	sbOpeCost		\$400	1 CBB Investment Tracking 1.docx	
12	sbOpeCost1		\$400	1 CBB Investment Tracking 1.docx	
13	sbOpeNote1	2 Chromebooks	1	1 CBB Investment Tracking 1.docx	
14	sbOthCat1	Overtime	1	1 CBB Investment Tracking 1.docx	
15	sbOthCost		\$2,000	1 CBB Investment Tracking 1.docx	
16	sbOthCost1		\$2,000	1 CBB Investment Tracking 1.docx	
17	sbStaAcc1	xxx-xxx-xxxx	1	1 CBB Investment Tracking 1.docx	
18	sbStaAcc2	xxx-xxx-xxxx	1	1 CBB Investment Tracking 1.docx	
19	sbStaCost		\$188,500	1 CBB Investment Tracking 1.docx	
20	sbStaFri		\$8,500	1 CBB Investment Tracking 1.docx	
21	sbStaFri1		\$6,000	1 CBB Investment Tracking 1.docx	
22	sbStaFri2		\$2,500	1 CBB Investment Tracking 1.docx	
23	sbStaN		3	1 CBB Investment Tracking 1.docx	
24	sbStaN1		2	1 CBB Investment Tracking 1.docx	
25	sbStaN2		1	1 CBB Investment Tracking 1.docx	

In this CSV file, the exported data are stored in the long format where each row is a data point from a completed Investment Tracking Form. Depending on which fields are filled out, the total number of rows for each completed form can vary because empty fields are not exported. This data can be easily



transformed into the wide format where all data points from an Investment Tracking Form are represented in one row.