Type Inference

This README file describe how to reproduce our tool to infer type information.

There are four steps in total:

step 1 is collecting the differential results;

step 2 is to parse the differential results to get the type indicator of basic types(bool, string and number) and array type;

step 3 is to get the key-get indicator of object type;

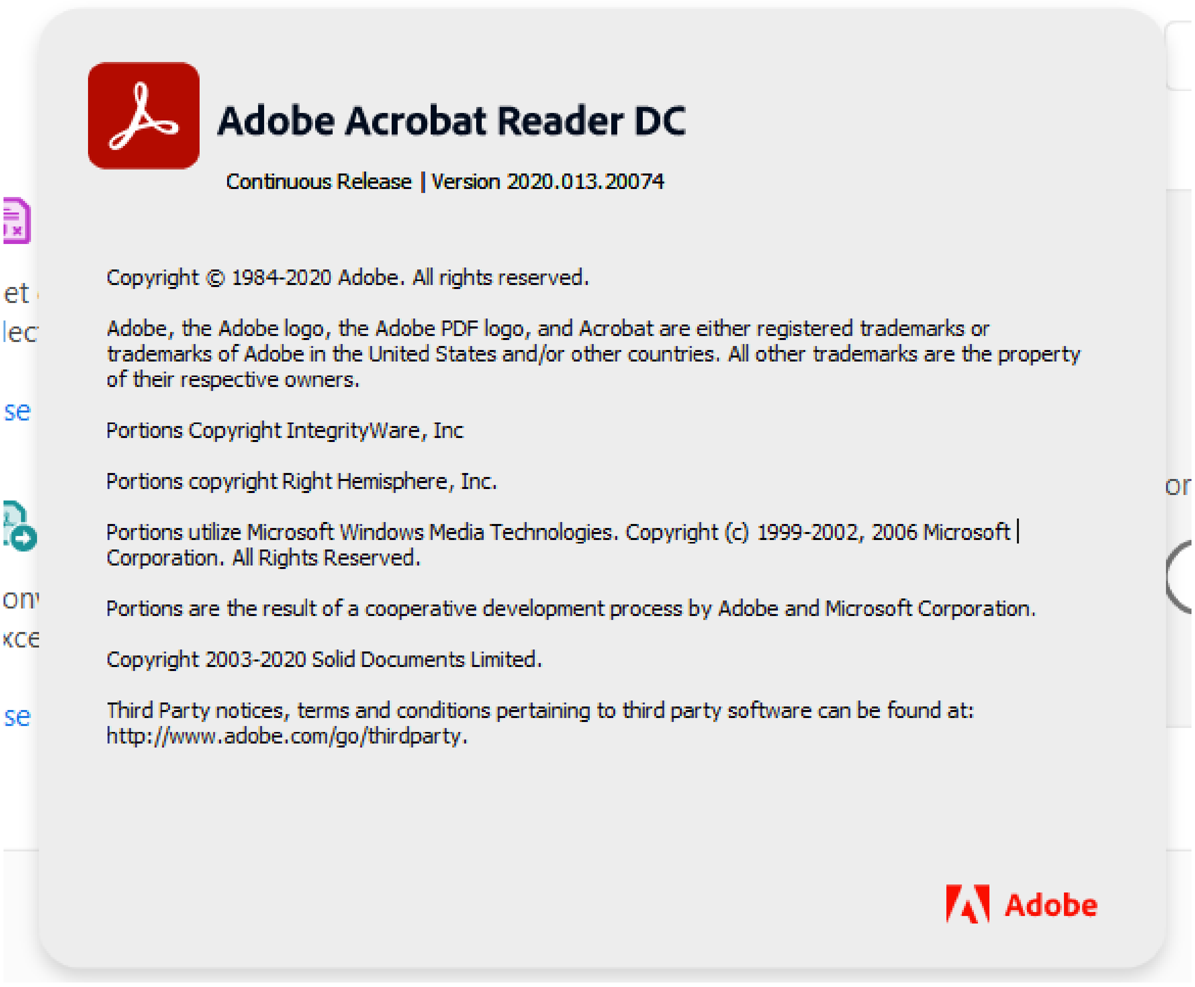
step 4 is to generate the type information of every binding calls.

The following will talk about reproduction in detail.

`Note:`

`1. for easier reproduction, the provided type inference's code is constructed based on Adobe Acrobat Reader version 2020.013.20074, please check the version before reproduction.`

`2. in our experience, the type information between different versions of Adobe Reader are usually the same, so we only need to infer type information through one version and use this infered type information to fuzz different versions of Adobe Reader(so does Foxit Reader)`


Adobe Acrobat Reader DC

Continuous Release | Version 2020.013.20074

Copyright © 1984-2020 Adobe. All rights reserved.

Adobe, the Adobe logo, the Adobe PDF logo, and Acrobat are either registered trademarks or trademarks of Adobe in the United States and/or other countries. All other trademarks are the property of their respective owners.

Portions Copyright IntegrityWare, Inc

Portions copyright Right Hemisphere, Inc.

Portions utilize Microsoft Windows Media Technologies. Copyright (c) 1999-2002, 2006 Microsoft Corporation. All Rights Reserved.

Portions are the result of a cooperative development process by Adobe and Microsoft Corporation.

Copyright 2003-2020 Solid Documents Limited.

Third Party notices, terms and conditions pertaining to third party software can be found at: http://www.adobe.com/go/thirdparty.

1. diff_collect

This step will collect the differential results. We use pin.exe to instrument the target PDF Reader to collect execution trace.

folder structure

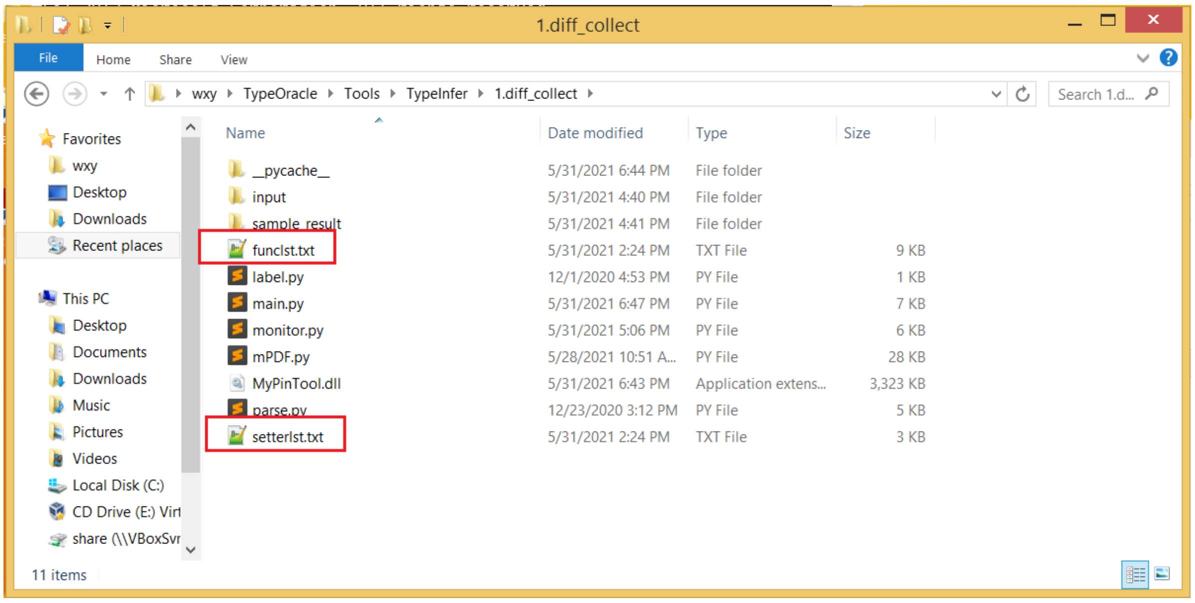
```
- sample_input
  - funclst.txt (list of all method names)
  - setterlst.txt (list of all accessors)

- sample_result (the folder store the sample result of this step)
  - save

- main.py

- monitor.py (to minitor the PDF Reader)

- mPDF.py (to embed javascript code to PDF)

- MyPinTool.dll (the instrumentation file)

- parse.py (parse the results recorded by pin.exe)
```

how to reproduce

1. make sure the Page Heap is turned off (execute following command and click yes, for more information about Page Heap, please refer to C:\Users\wxy\TypeOracle\Other\README.pdf)

```
"C:\Program Files (x86)\Windows Kits\8.1\Debuggers\x86\gflags.exe" /p /disable
"C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
```

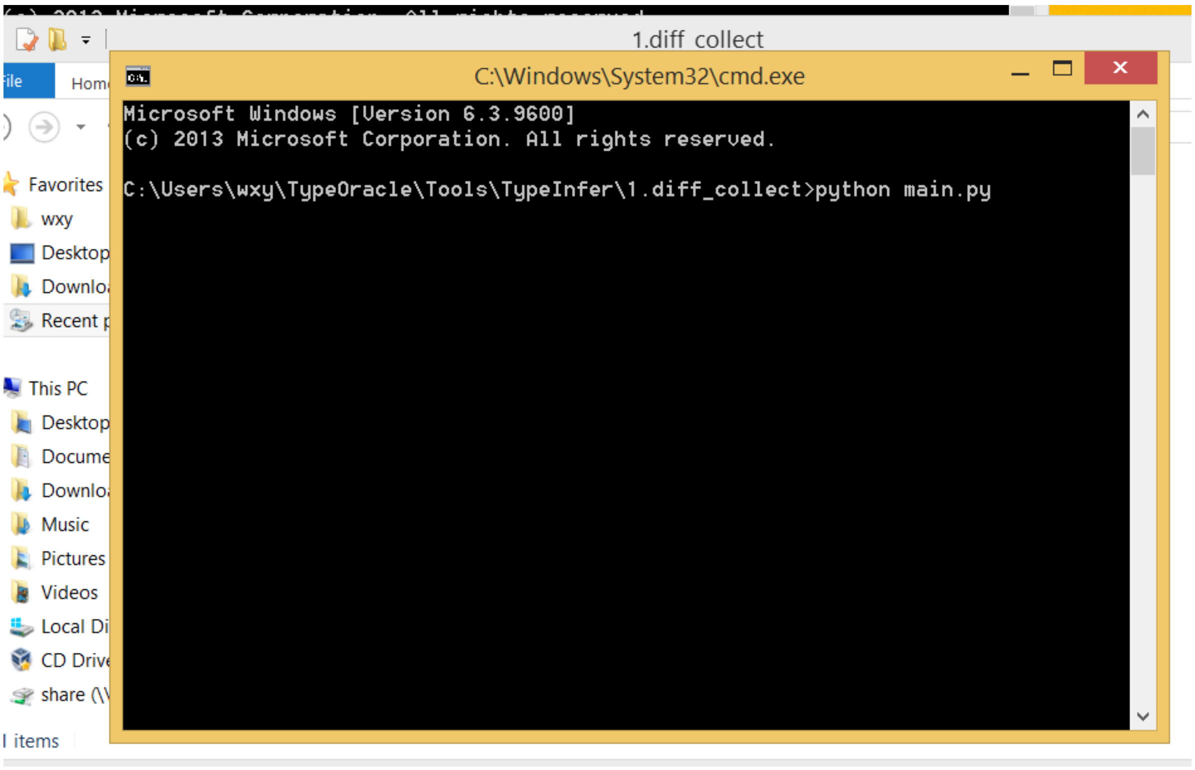
2. copy `funclst.txt` and `setterlst.txt` from `sample_input` folder to current folder



Name	Date modified	Type	Size
__pycache__	5/31/2021 6:44 PM	File folder	
input	5/31/2021 4:40 PM	File folder	
sample_result	5/31/2021 4:41 PM	File folder	
funclst.txt	5/31/2021 2:24 PM	TXT File	9 KB
label.py	12/1/2020 4:53 PM	PY File	1 KB
main.py	5/31/2021 6:47 PM	PY File	7 KB
monitor.py	5/31/2021 5:06 PM	PY File	6 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:43 PM	Application extens...	3,323 KB
parse.py	12/23/2020 3:12 PM	PY File	5 KB
setterlst.txt	5/31/2021 2:24 PM	TXT File	3 KB

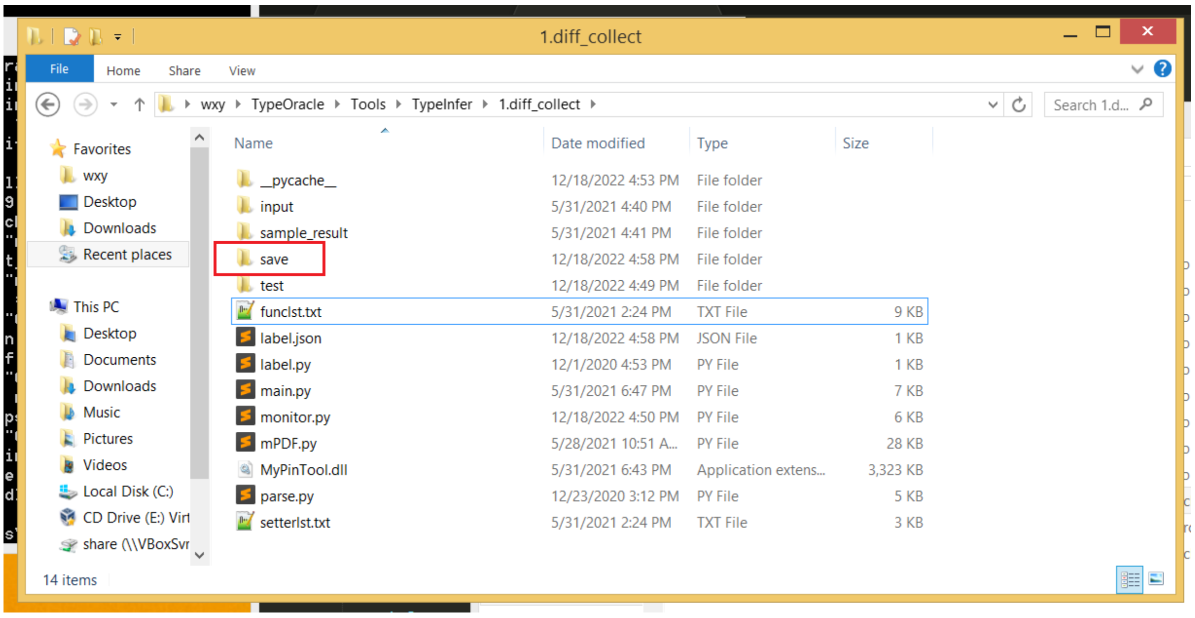
3. execute `main.py` at the current folder to start collect the differential results of every binding call (this step will take more than ten hours)

```
python main.py
```



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\1.diff_collect>python main.py
```

4. the result will be stored at `save` folder



Name	Date modified	Type	Size
__pycache__	12/18/2022 4:53 PM	File folder	
input	5/31/2021 4:40 PM	File folder	
sample_result	5/31/2021 4:41 PM	File folder	
save	12/18/2022 4:58 PM	File folder	
test	12/18/2022 4:49 PM	File folder	
funclst.txt	5/31/2021 2:24 PM	TXT File	9 KB
label.json	12/18/2022 4:58 PM	JSON File	1 KB
label.py	12/1/2020 4:53 PM	PY File	1 KB
main.py	5/31/2021 6:47 PM	PY File	7 KB
monitor.py	12/18/2022 4:50 PM	PY File	6 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:43 PM	Application extens...	3,323 KB
parse.py	12/23/2020 3:12 PM	PY File	5 KB
setterlst.txt	5/31/2021 2:24 PM	TXT File	3 KB

2. parse_diffresult

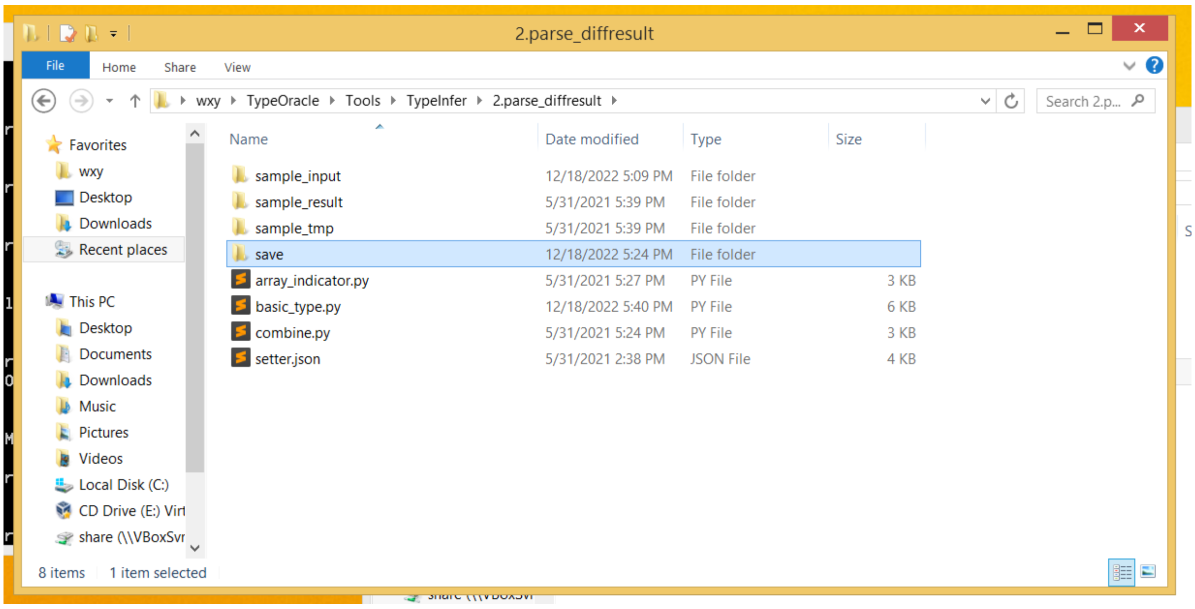
This step is to parse the differential results collected in step 1, and get the type indicator of bool, string , number and array type.

folder structure

```
- sample_input
  - save (differential results collected in step 1)
  - setter.json (list of all accessors and its type)

- sample_result (the folder store the sample result of this step)
  - basic.txt
  - array.txt

- sample_tmp (store sample intermediate results)

- array_indicator.py (get the array indicator)

- basic_type.py (get basic types' indicator)

- combine.py (combine step 1's result)
```

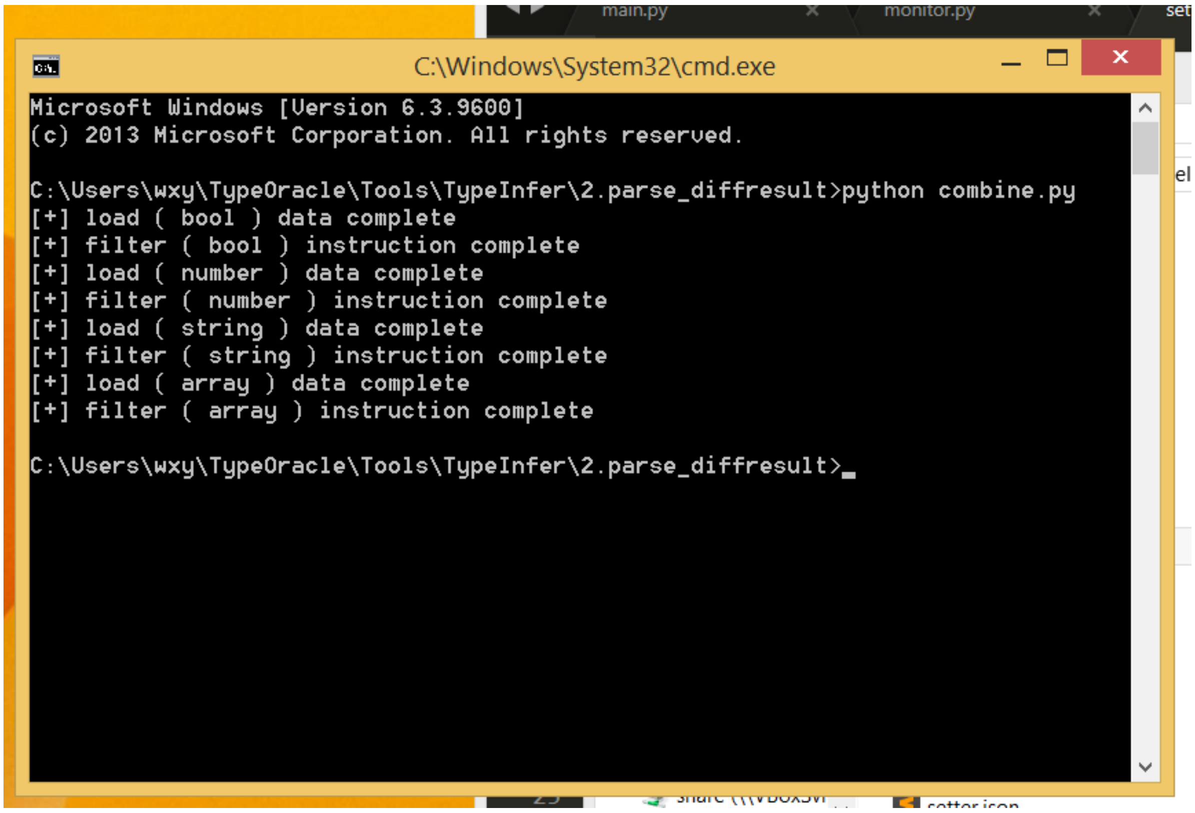
how to reproduce

1. copy `save` folder in step 1 to this folder (you can also copy `save` from `sample_input` folder) , and copy setter.json in `sample_input` folder to this folder

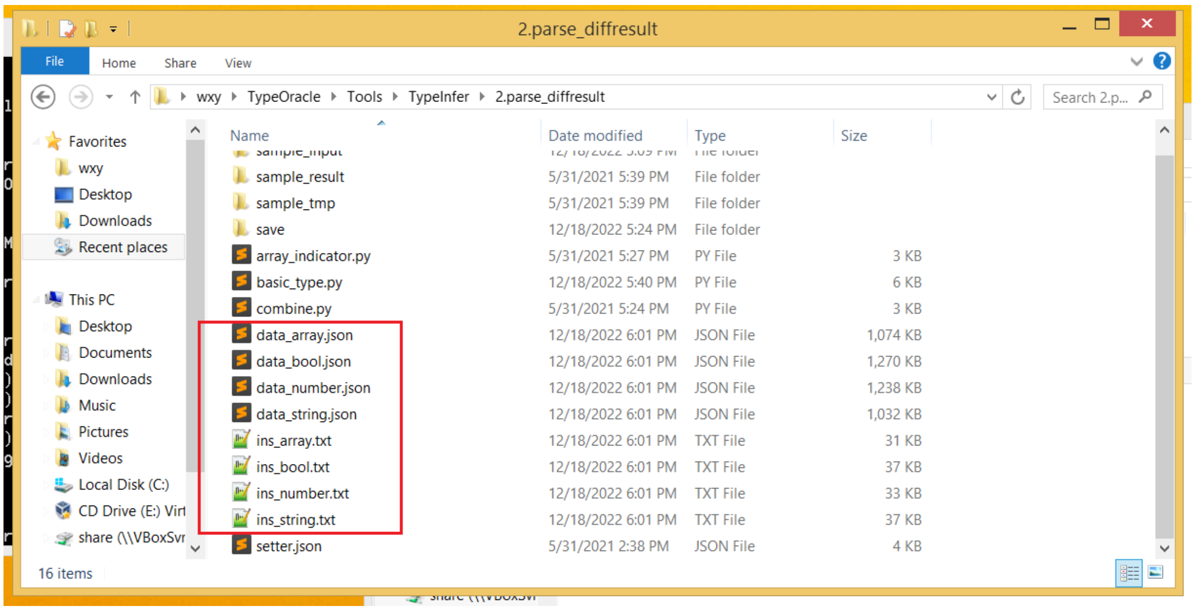


```
2.parse_diffresult
wxy > TypeOracle > Tools > TypeInfer > 2.parse_diffresult

Name                 Date modified        Type         Size
sample_input         12/18/2022 5:09 PM   File folder
sample_result        5/31/2021 5:39 PM    File folder
sample_tmp           5/31/2021 5:39 PM    File folder
save                 12/18/2022 5:24 PM   File folder
array_indicator.py   5/31/2021 5:27 PM    PY File      3 KB
basic_type.py        12/18/2022 5:40 PM   PY File      6 KB
combine.py           5/31/2021 5:24 PM    PY File      3 KB
setter.json          5/31/2021 2:38 PM    JSON File    4 KB

8 items    1 item selected
```

2. execute combine.py, this will generate some intermediate result files(data_x.json ins_x.json)



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\2.parse_diffresult>python combine.py
[+] load ( bool ) data complete
[+] filter ( bool ) instruction complete
[+] load ( number ) data complete
[+] filter ( number ) instruction complete
[+] load ( string ) data complete
[+] filter ( string ) instruction complete
[+] load ( array ) data complete
[+] filter ( array ) instruction complete

C:\Users\wxy\TypeOracle\Tools\TypeInfer\2.parse_diffresult>_
```


2.parse_diffresult

wxy ▸ TypeOracle ▸ Tools ▸ TypeInfer ▸ 2.parse_diffresult

Name	Date modified	Type	Size
sample_input	12/18/2022 5:09 PM	File folder	
sample_result	5/31/2021 5:39 PM	File folder	
sample_tmp	5/31/2021 5:39 PM	File folder	
save	12/18/2022 5:24 PM	File folder	
array_indicator.py	5/31/2021 5:27 PM	PY File	3 KB
basic_type.py	12/18/2022 5:40 PM	PY File	6 KB
combine.py	5/31/2021 5:24 PM	PY File	3 KB
data_array.json	12/18/2022 6:01 PM	JSON File	1,074 KB
data_bool.json	12/18/2022 6:01 PM	JSON File	1,270 KB
data_number.json	12/18/2022 6:01 PM	JSON File	1,238 KB
data_string.json	12/18/2022 6:01 PM	JSON File	1,032 KB
ins_array.txt	12/18/2022 6:01 PM	TXT File	31 KB
ins_bool.txt	12/18/2022 6:01 PM	TXT File	37 KB
ins_number.txt	12/18/2022 6:01 PM	TXT File	33 KB
ins_string.txt	12/18/2022 6:01 PM	TXT File	37 KB
setter.json	5/31/2021 2:38 PM	JSON File	4 KB

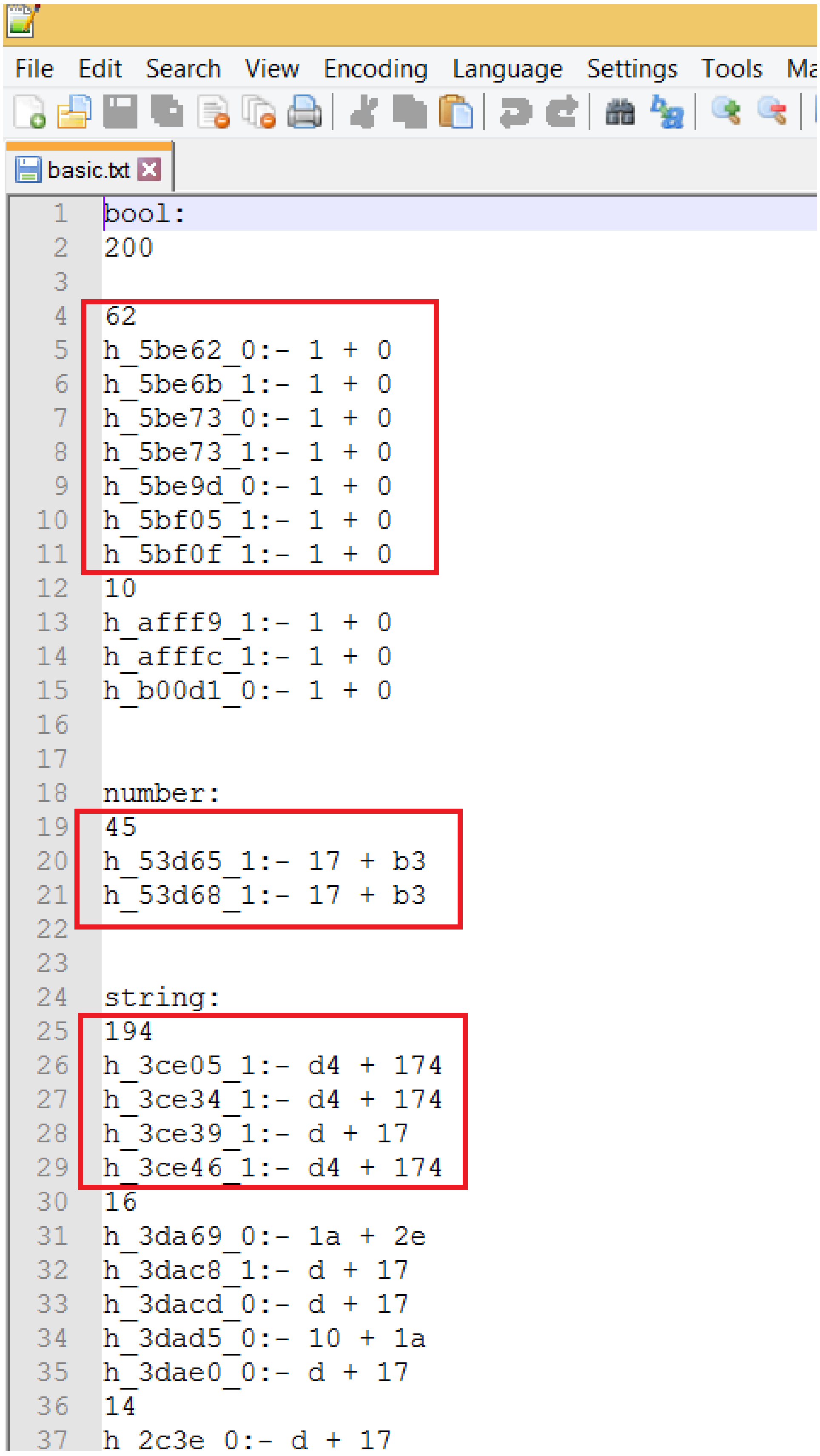
16 items

3. execute basic.py to get basic types' type indicator, redirect the results to basic.txt

```
python basic_type.py > basic.txt
```

The instructions that show up the most times is the type indicator, from the following snapshot, we can infer following type indicators.

```
type indicator:(h means this instruction belongs to Escript.api module, for other
modules, please refer tp C:\Users\wxy\TypeOracle\Other\label.json)
bool: 0x5be62, 0x5be6b,0x5be9d,0x5bf05,0x5bf0f
number: 0x53d65,0x53d68
string: 0x3ce05,0x3ce34,0x3ce39,0x3ce46
```



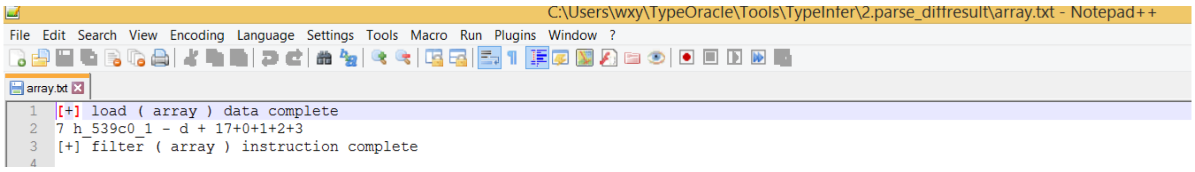
```
bool:
200

62
h_5be62_0:- 1 + 0
h_5be6b_1:- 1 + 0
h_5be73_0:- 1 + 0
h_5be73_1:- 1 + 0
h_5be9d_0:- 1 + 0
h_5bf05_1:- 1 + 0
h_5bf0f_1:- 1 + 0
10
h_afff9_1:- 1 + 0
h_afffc_1:- 1 + 0
h_b00d1_0:- 1 + 0


number:
45
h_53d65_1:- 17 + b3
h_53d68_1:- 17 + b3


string:
194
h_3ce05_1:- d4 + 174
h_3ce34_1:- d4 + 174
h_3ce39_1:- d + 17
h_3ce46_1:- d4 + 174
16
h_3da69_0:- 1a + 2e
h_3dac8_1:- d + 17
h_3dacd_0:- d + 17
h_3dad5_0:- 10 + 1a
h_3dae0_0:- d + 17
14
h_2c3e_0:- d + 17
```

4. execute array_indicator.py， redirect the results to array.txt

```
python array_indicator.py > array.txt
```

we can get the following array type indicator

```
type indicator:
array: 0x539c0
```


C:\Users\wxy\TypeOracle\Tools\TypeInfer\2.parse_diffresult\array.txt - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
array.txt

```
[+] load ( array ) data complete
7 h_539c0_1 - d + 17+0+1+2+3
[+] filter ( array ) instruction complete
```

3. generic_object

An object contains key-value pairs, and one can access the values by referring to the keys. The keys can not be obtained by blind guessing. Observe that the binding calls generally get the desired key of an object from the data segment of the program. Hence the strings read from the data segment during the binding call execution are considered as possible keys.

The script engines of Adobe Reader and Foxit Reader both have two important APIs for handling keys in objects, similar to hasKeyInObject() and getValueByKeyInObject(). The former is to check whether the key exists, and the latter is to get the value from the object according to the key. This is our main idea to get key-get indicator.

folder structure

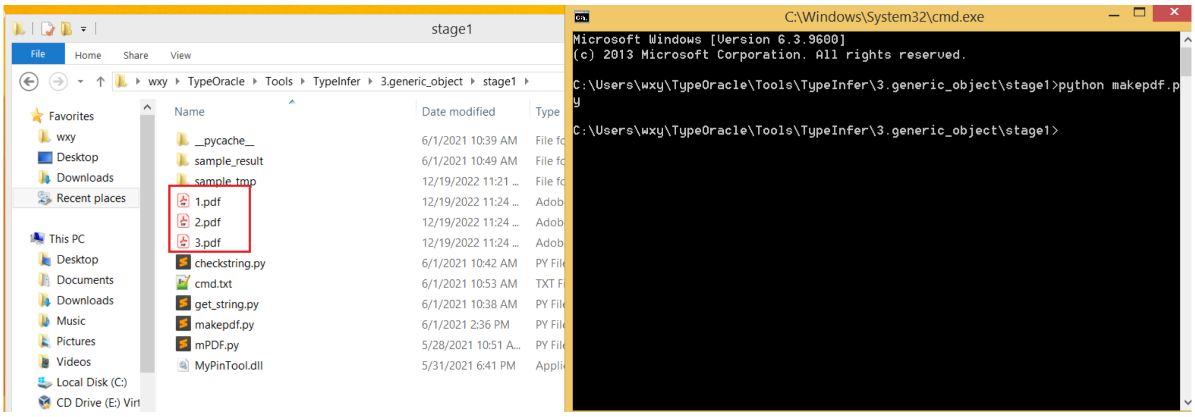
```
- stage1
  - sample_result  (the folder store the sample result of this step)

  - sample_tmp (the folder store sample intermediate results)

  - checkstring.py (script file executed in IDA pro)

  - cmd.txt (some commands to start Adobe Reader with pin.exe)

  - get_string.py (get the string in execution strace)

  - label.py

  - makepdf.py (generate PDFs with special javascript codes)

  - mPDF.py (to embed javascript code to PDF)

  - MyPinTool.dll (the instrumentation file)

- stage2
  - sample_result  (the folder store the sample result of this step)

  - sample_tmp (the folder store sample intermediate results)

  - cmd.txt (some commands to start Adobe Reader with pin.exe)

  - idapush.py (script file executed in IDA pro)

  - label.py

  - makepdf.py (generate PDFs with special javascript codes)

  - mPDF.py (to embed javascript code to PDF)

  - MyPinTool.dll (the instrumentation file)
  - parse.py

  - parse2.py

  - record.py (script file executed in windbg)

- EScript.api.idb (the result database of IDA pro to parse EScript.api)
```

how to reproduce

stage 1

1. execute `makepdf.py` to generate three test PDFs

```
python makepdf.py
```

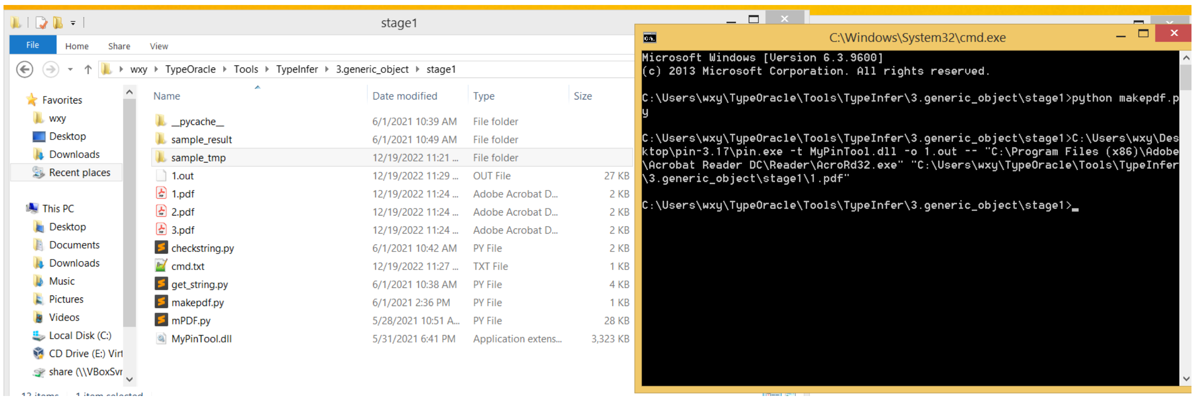


Name	Date modified	Type
__pycache__	6/1/2021 10:39 AM	File fo
sample_result	6/1/2021 10:49 AM	File fo
sample_tmp	12/19/2022 11:21 ...	File fo
1.pdf	12/19/2022 11:24 ...	Adob
2.pdf	12/19/2022 11:24 ...	Adob
3.pdf	12/19/2022 11:24 ...	Adob
checkstring.py	6/1/2021 10:42 AM	PY File
cmd.txt	6/1/2021 10:53 AM	TXT F
get_string.py	6/1/2021 10:38 AM	PY File
makepdf.py	6/1/2021 2:36 PM	PY File
mPDF.py	5/28/2021 10:51 A...	PY File
MyPinTool.dll	5/31/2021 6:41 PM	Appli

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>python makepdf.p
y

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>
```

2. execute the commands in `cmd.txt`, remember to close Adobe Reader when the execution of every command is finished(execution of each command will take about 5 minutes), this will generate three files: `1.out`, `2.out` and `3.out`

```
commands in cmd.txt:
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage1\1.pdf"
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 2.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage1\2.pdf"
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 3.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage1\3.pdf"
```

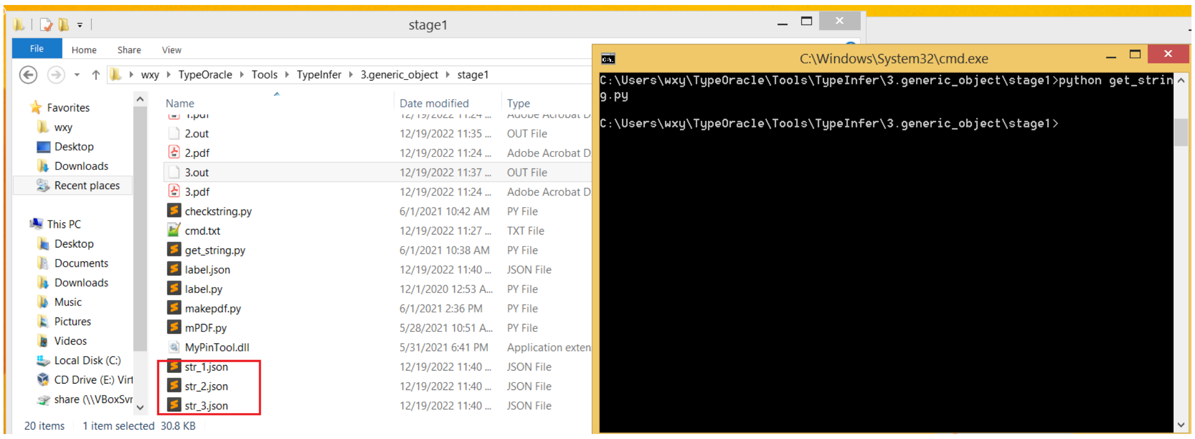


Name	Date modified	Type	Size
__pycache__	6/1/2021 10:39 AM	File folder	
sample_result	6/1/2021 10:49 AM	File folder	
sample_tmp	12/19/2022 11:21 ...	File folder	
1.out	12/19/2022 11:29 ...	OUT File	27 KB
1.pdf	12/19/2022 11:24 ...	Adobe Acrobat D...	2 KB
2.pdf	12/19/2022 11:24 ...	Adobe Acrobat D...	2 KB
3.pdf	12/19/2022 11:24 ...	Adobe Acrobat D...	2 KB
checkstring.py	6/1/2021 10:42 AM	PY File	2 KB
cmd.txt	12/19/2022 11:27 ...	TXT File	1 KB
get_string.py	6/1/2021 10:38 AM	PY File	4 KB
makepdf.py	6/1/2021 2:36 PM	PY File	1 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:41 PM	Application extens...	3,323 KB

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>python makepdf.p
y

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage1\1.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>
```

3. execute `get_string.py` to get str_x.json

```
python get_string.py
```



Name	Date modified	Type
2.out	12/19/2022 11:35 ...	OUT File
2.pdf	12/19/2022 11:24 ...	Adobe Acrobat D
3.out	12/19/2022 11:37 ...	OUT File
3.pdf	12/19/2022 11:24 ...	Adobe Acrobat D
checkstring.py	6/1/2021 10:42 AM	PY File
cmd.txt	12/19/2022 11:27 ...	TXT File
get_string.py	6/1/2021 10:38 AM	PY File
label.json	12/19/2022 11:40 ...	JSON File
label.py	12/1/2020 12:53 A...	PY File
makepdf.py	6/1/2021 2:36 PM	PY File
mPDF.py	5/28/2021 10:51 A...	PY File
MyPinTool.dll	5/31/2021 6:41 PM	Application exten
str_1.json	12/19/2022 11:40 ...	JSON File
str_2.json	12/19/2022 11:40 ...	JSON File
str_3.json	12/19/2022 11:40 ...	JSON File

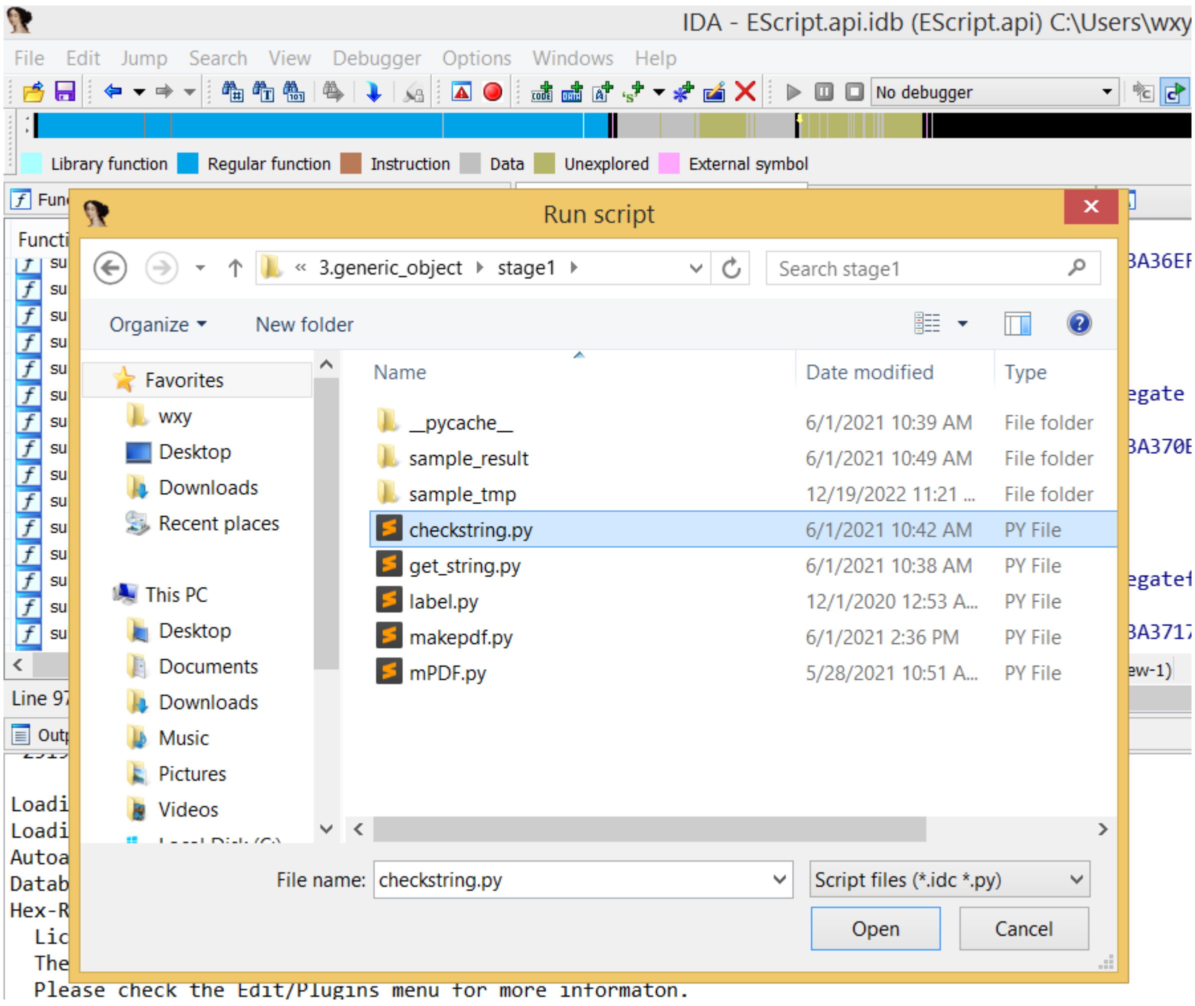
20 items   1 item selected 30.8 KB

```
C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>python get_strin
g.py

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage1>
```

4. open EScript.api.idb with IDA pro, and execute scripts `checkstring.py`

This is IDA pro

Click file->Script file (file button is in the upper left corner)

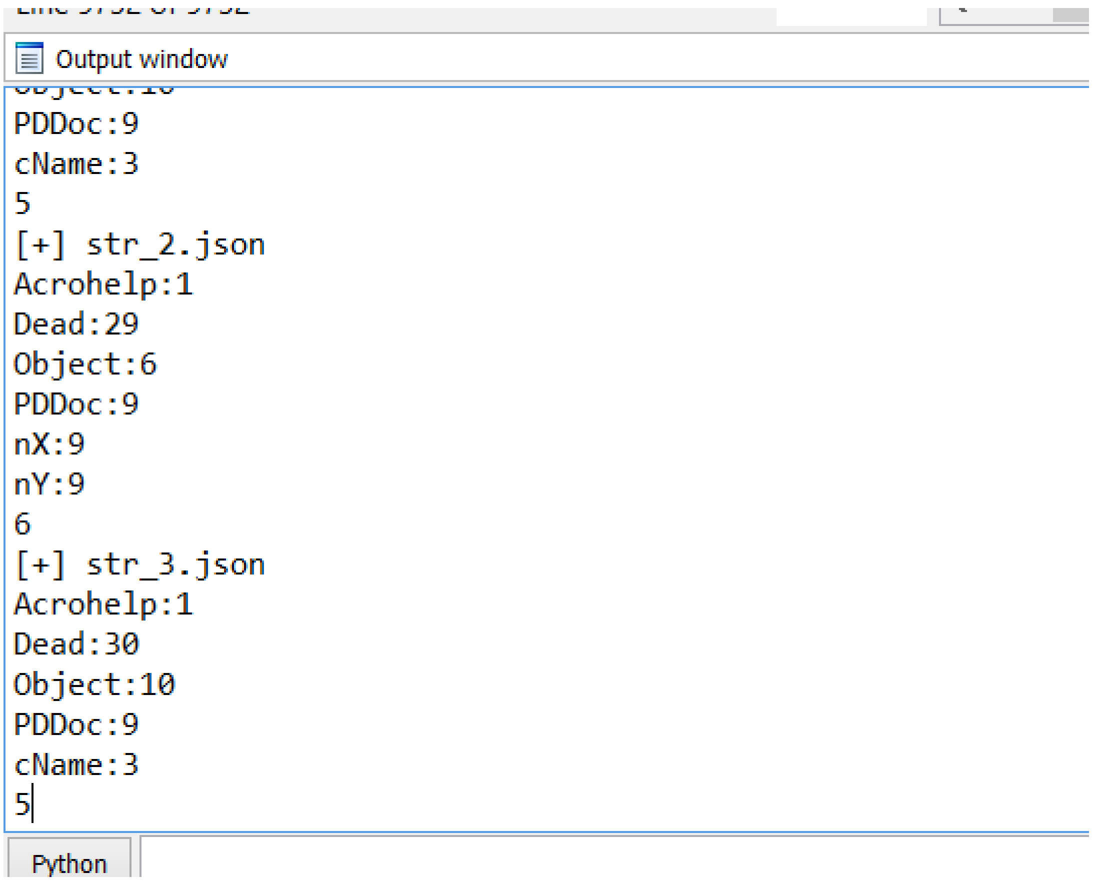


Name	Date modified	Type
__pycache__	6/1/2021 10:39 AM	File folder
sample_result	6/1/2021 10:49 AM	File folder
sample_tmp	12/19/2022 11:21 ...	File folder
checkstring.py	6/1/2021 10:42 AM	PY File
get_string.py	6/1/2021 10:38 AM	PY File
label.py	12/1/2020 12:53 A...	PY File
makepdf.py	6/1/2021 2:36 PM	PY File
mPDF.py	5/28/2021 10:51 A...	PY File

File name: checkstring.py

```
Loadi
Loadi
Autoa
Datab
Hex-R
  Lic
  The
  Please check the Edit/Plugins menu for more informaton.
```

The result will output through output windows

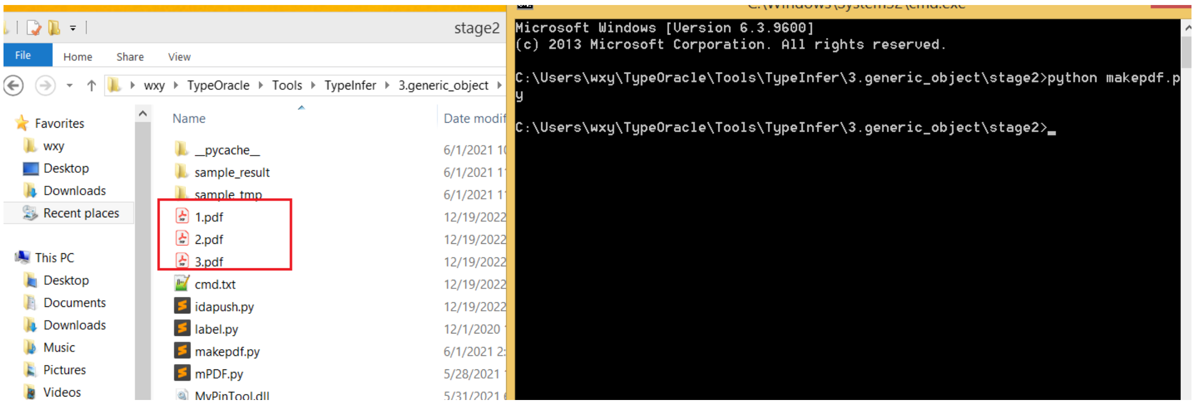


```
PDDoc:9
cName:3
5
[+] str_2.json
Acrohelp:1
Dead:29
Object:6
PDDoc:9
nX:9
nY:9
6
[+] str_3.json
Acrohelp:1
Dead:30
Object:10
PDDoc:9
cName:3
5
```

stage2

1. execute `makepdf.py` to generate three test PDFs (this step use the key infered in stage1)

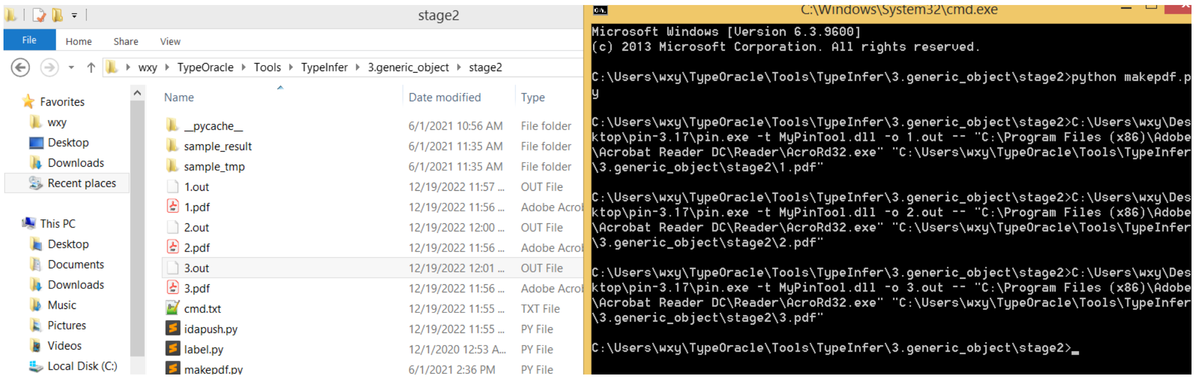
```
python makepdf.py
```



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python makepdf.p
y

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>_
```

2. execute the following three commands in `cmd.txt`, remember to close Adobe Reader when the execution of every command is finished(execution of each command will take about 5 minutes), this will generate three files: `1.out`, `2.out` and `3.out`

```
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage2\1.pdf"
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 2.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage2\2.pdf"
C:\Users\wxy\Desktop\pin-3.17\pin.exe -t MyPinTool.dll -o 3.out -- "C:\Program
Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
"C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage2\3.pdf"
```

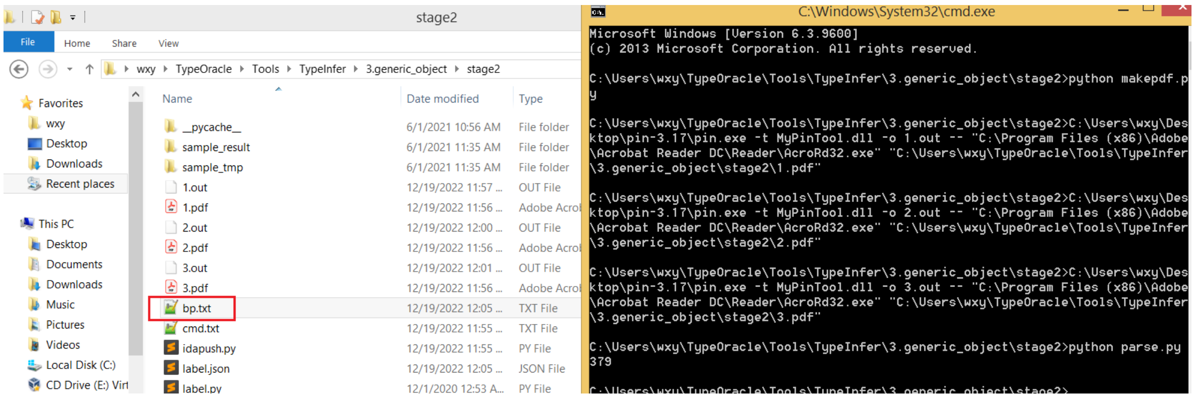


Name	Date modified	Type
__pycache__	6/1/2021 10:56 AM	File folder
sample_result	6/1/2021 11:35 AM	File folder
sample_tmp	6/1/2021 11:35 AM	File folder
1.out	12/19/2022 11:57 ...	OUT File
1.pdf	12/19/2022 11:56 ...	Adobe Acro
2.out	12/19/2022 12:00 ...	OUT File
2.pdf	12/19/2022 11:56 ...	Adobe Acro
3.out	12/19/2022 12:01 ...	OUT File
3.pdf	12/19/2022 11:56 ...	Adobe Acro
cmd.txt	12/19/2022 11:55 ...	TXT File
idapush.py	12/19/2022 11:55 ...	PY File
label.py	12/1/2020 12:53 A...	PY File
makepdf.py	6/1/2021 2:36 PM	PY File

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python makepdf.p
y

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\1.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 2.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\2.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 3.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\3.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>_
```

3. execute `parse.py` to get `bp.txt`

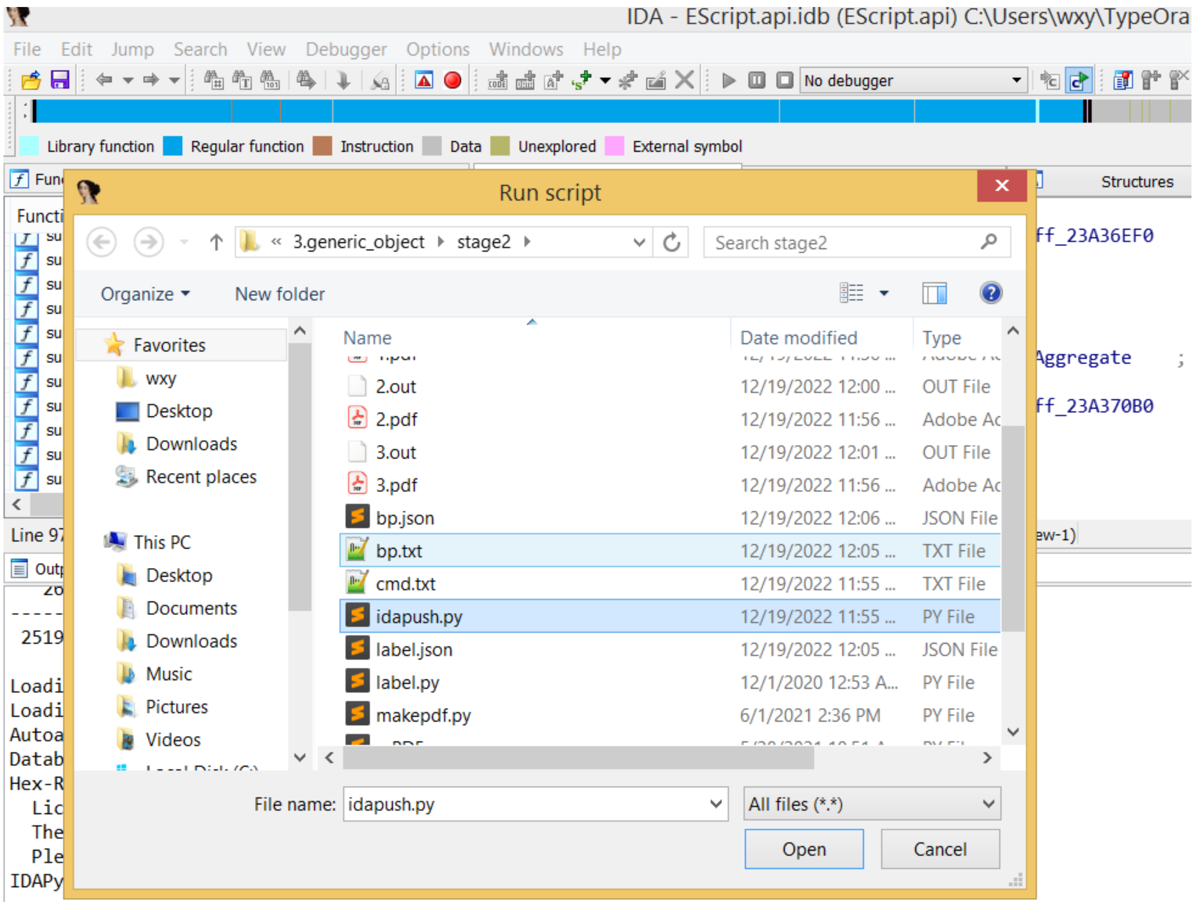
```
python parse.py
```



Name	Date modified	Type
__pycache__	6/1/2021 10:56 AM	File folder
sample_result	6/1/2021 11:35 AM	File folder
sample_tmp	6/1/2021 11:35 AM	File folder
1.out	12/19/2022 11:57 ...	OUT File
1.pdf	12/19/2022 11:56 ...	Adobe Acro
2.out	12/19/2022 12:00 ...	OUT File
2.pdf	12/19/2022 11:56 ...	Adobe Acro
3.out	12/19/2022 12:01 ...	OUT File
3.pdf	12/19/2022 11:56 ...	Adobe Acro
bp.txt	12/19/2022 12:05 ...	TXT File
cmd.txt	12/19/2022 11:55 ...	TXT File
idapush.py	12/19/2022 11:55 ...	PY File
label.json	12/19/2022 12:05 ...	JSON File
label.py	12/1/2020 12:53 A...	PY File

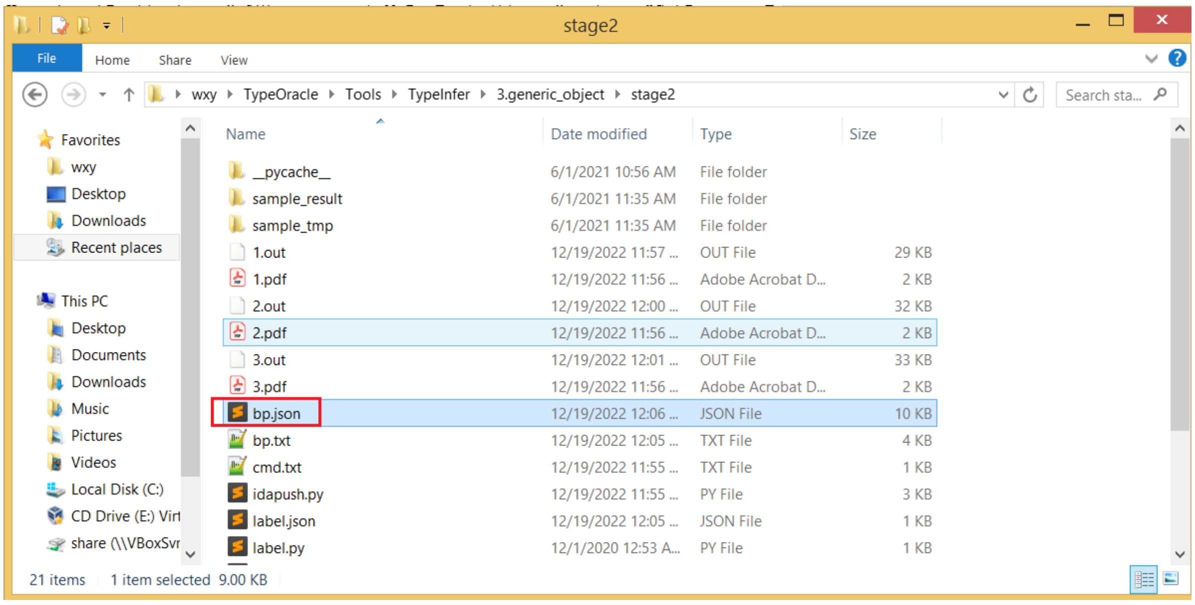
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python makepdf.p
y

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\1.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 2.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\2.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 3.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\3.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python parse.py
379

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>
```

4. open EScript.api.idb with IDA pro, and execute scripts `idapush.py`



Name	Date modified	Type
2.out	12/19/2022 12:00 ...	OUT File
2.pdf	12/19/2022 11:56 ...	Adobe Ac
3.out	12/19/2022 12:01 ...	OUT File
3.pdf	12/19/2022 11:56 ...	Adobe Ac
bp.json	12/19/2022 12:06 ...	JSON File
bp.txt	12/19/2022 12:05 ...	TXT File
cmd.txt	12/19/2022 11:55 ...	TXT File
idapush.py	12/19/2022 11:55 ...	PY File
label.json	12/19/2022 12:05 ...	JSON File
label.py	12/1/2020 12:53 A...	PY File
makepdf.py	6/1/2021 2:36 PM	PY File

File name: idapush.py
All files (*.*)

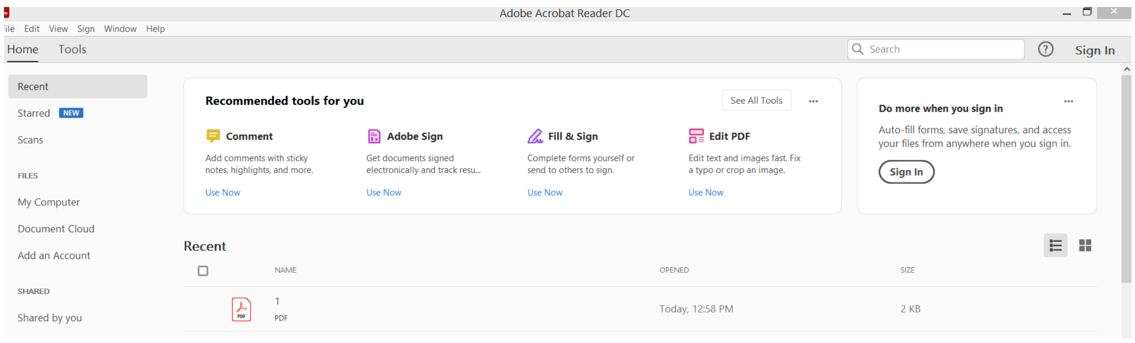


Name	Date modified	Type	Size
__pycache__	6/1/2021 10:56 AM	File folder	
sample_result	6/1/2021 11:35 AM	File folder	
sample_tmp	6/1/2021 11:35 AM	File folder	
1.out	12/19/2022 11:57 ...	OUT File	29 KB
1.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
2.out	12/19/2022 12:00 ...	OUT File	32 KB
2.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
3.out	12/19/2022 12:01 ...	OUT File	33 KB
3.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
bp.json	12/19/2022 12:06 ...	JSON File	10 KB
bp.txt	12/19/2022 12:05 ...	TXT File	4 KB
cmd.txt	12/19/2022 11:55 ...	TXT File	1 KB
idapush.py	12/19/2022 11:55 ...	PY File	3 KB
label.json	12/19/2022 12:05 ...	JSON File	1 KB
label.py	12/1/2020 12:53 A...	PY File	1 KB

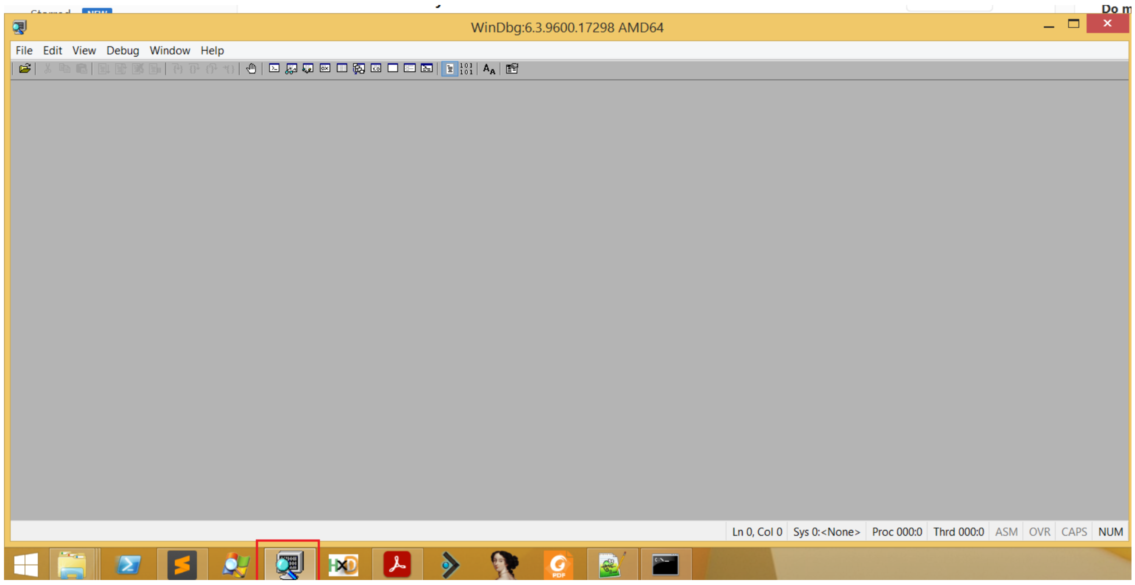
21 items 1 item selected 9.00 KB

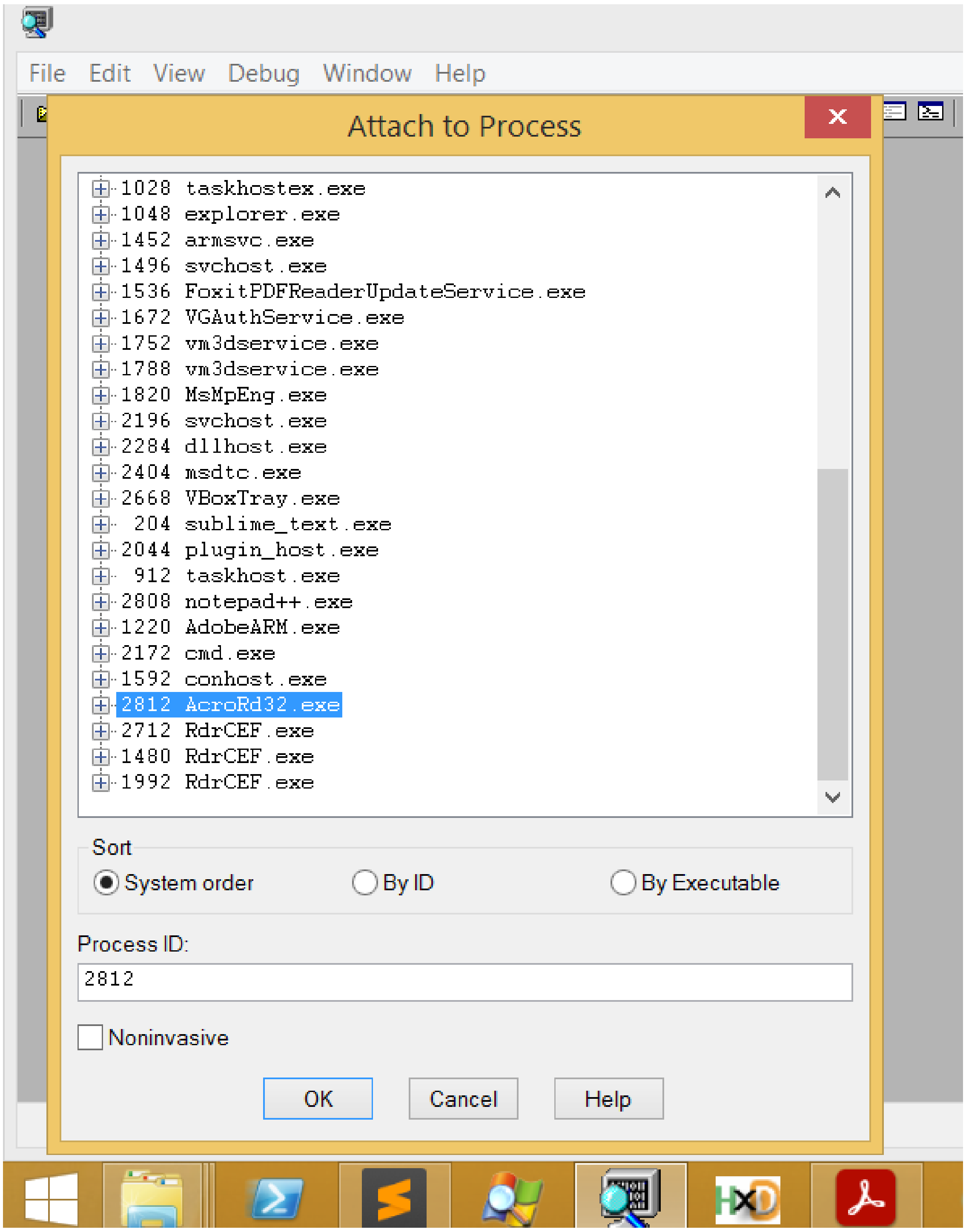
5. 打开1.pdf并关闭(载入EScript模块)，attach windbg并运行cmd.txt中的指令载入pykd脚本，然后打开1.pdf,将结果result.json重命名为1.json,打开2.pdf,重命名为2.json,打开3.pdf,重命名为3.json

5. open 1.pdf with Adobe Reader (this step is to load EScript.api module, 1.pdf will automatically close itself)


Adobe Acrobat Reader DC
File Edit View Sign Window Help
Home Tools
Search
Sign In
Recent
Starred NEW
Scans
FILES
My Computer
Document Cloud
Add an Account
SHARED
Shared by you
Recommended tools for you
See All Tools
Comment
Add comments with sticky notes, highlights, and more.
Use Now
Adobe Sign
Get documents signed electronically and track resu...
Use Now
Fill & Sign
Complete forms yourself or send to others to sign.
Use Now
Edit PDF
Edit text and images fast. Fix a typo or crop an image.
Use Now
Do more when you sign in
Auto-fill forms, save signatures, and access your files from anywhere when you sign in.
Sign In
Recent
NAME
OPENED
SIZE
1
PDF
Today, 12:58 PM
2 KB

use windbg to attach to Adobe Reader





```
1028 taskhostex.exe
1048 explorer.exe
1452 armsvc.exe
1496 svchost.exe
1536 FoxitPDFReaderUpdateService.exe
1672 VGAuthService.exe
1752 vm3dservice.exe
1788 vm3dservice.exe
1820 MsMpEng.exe
2196 svchost.exe
2284 dllhost.exe
2404 msdtc.exe
2668 VBoxTray.exe
 204 sublime_text.exe
2044 plugin_host.exe
 912 taskhost.exe
2808 notepad++.exe
1220 AdobeARM.exe
2172 cmd.exe
1592 conhost.exe
2812 AcroRd32.exe
2712 RdrCEF.exe
1480 RdrCEF.exe
1992 RdrCEF.exe
```

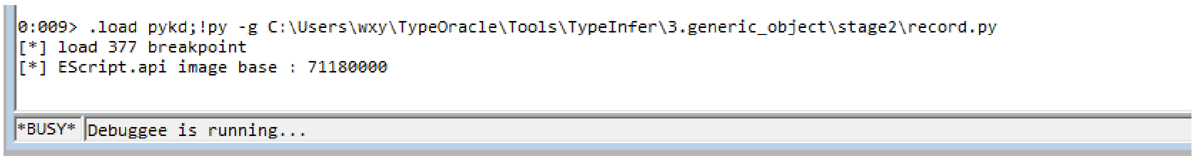
Sort: [✓ System order] By ID By Executable

Process ID: 2812

Noninvasive

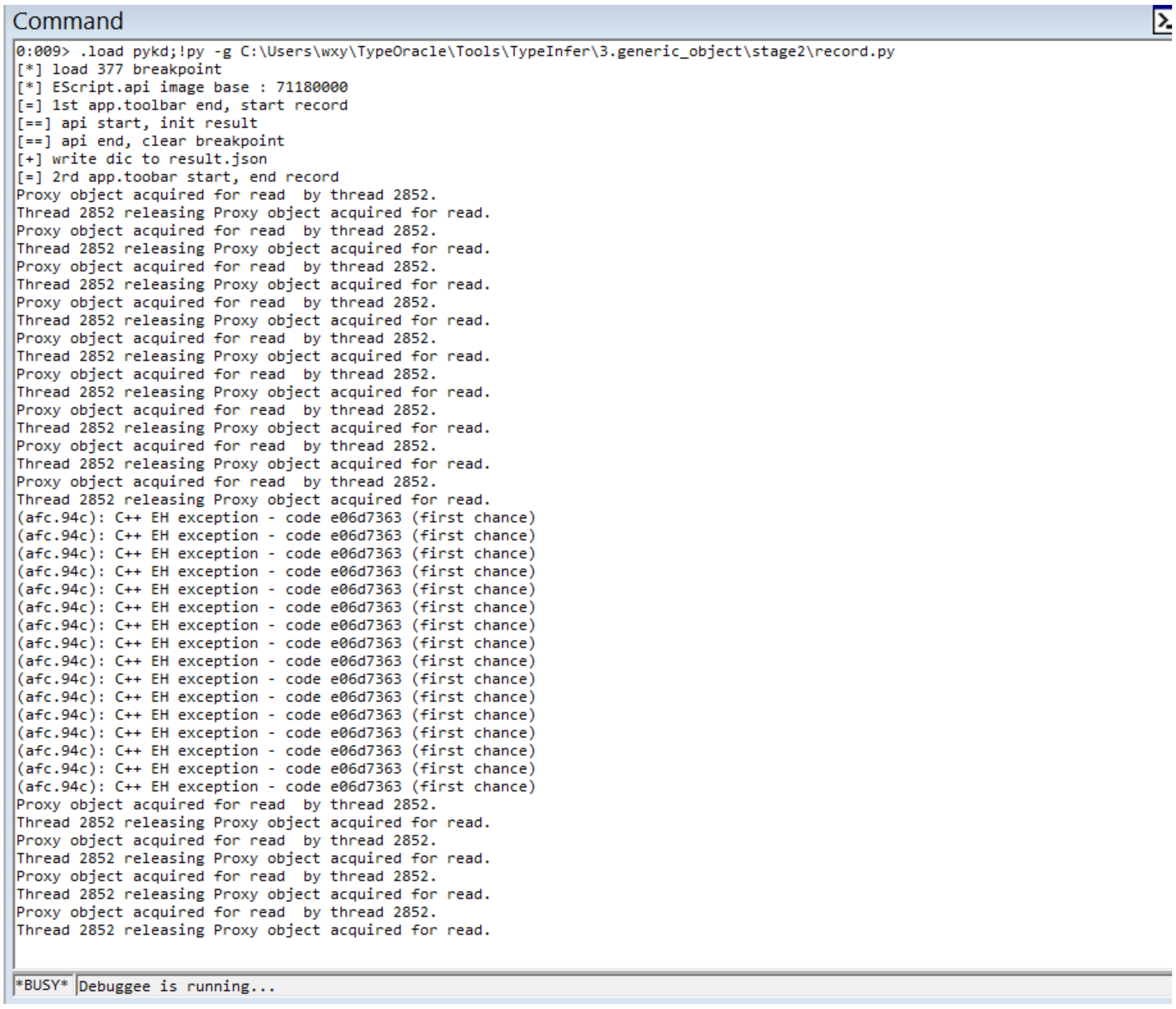
execute the following command to load record.py

```
.load pykd;!py -g
C:\Users\wxy\TypeOracle\Tools\TypeInfer\adobe\3.generic_object\stage2\record.py
```

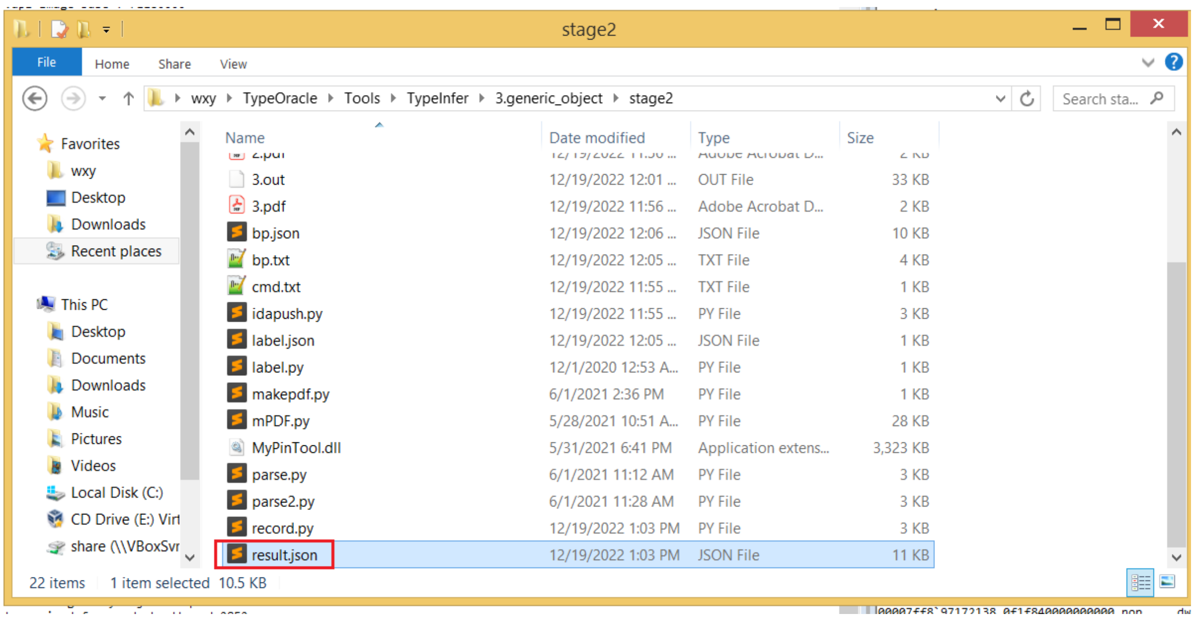


```
0:009> .load pykd;!py -g C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2\record.py
[*] load 377 breakpoint
[*] EScript.api image base : 71180000

*BUSY* Debuggee is running...
```

open 1.pdf with Adobe Reader, this will generate result.json, rename it to 1.json



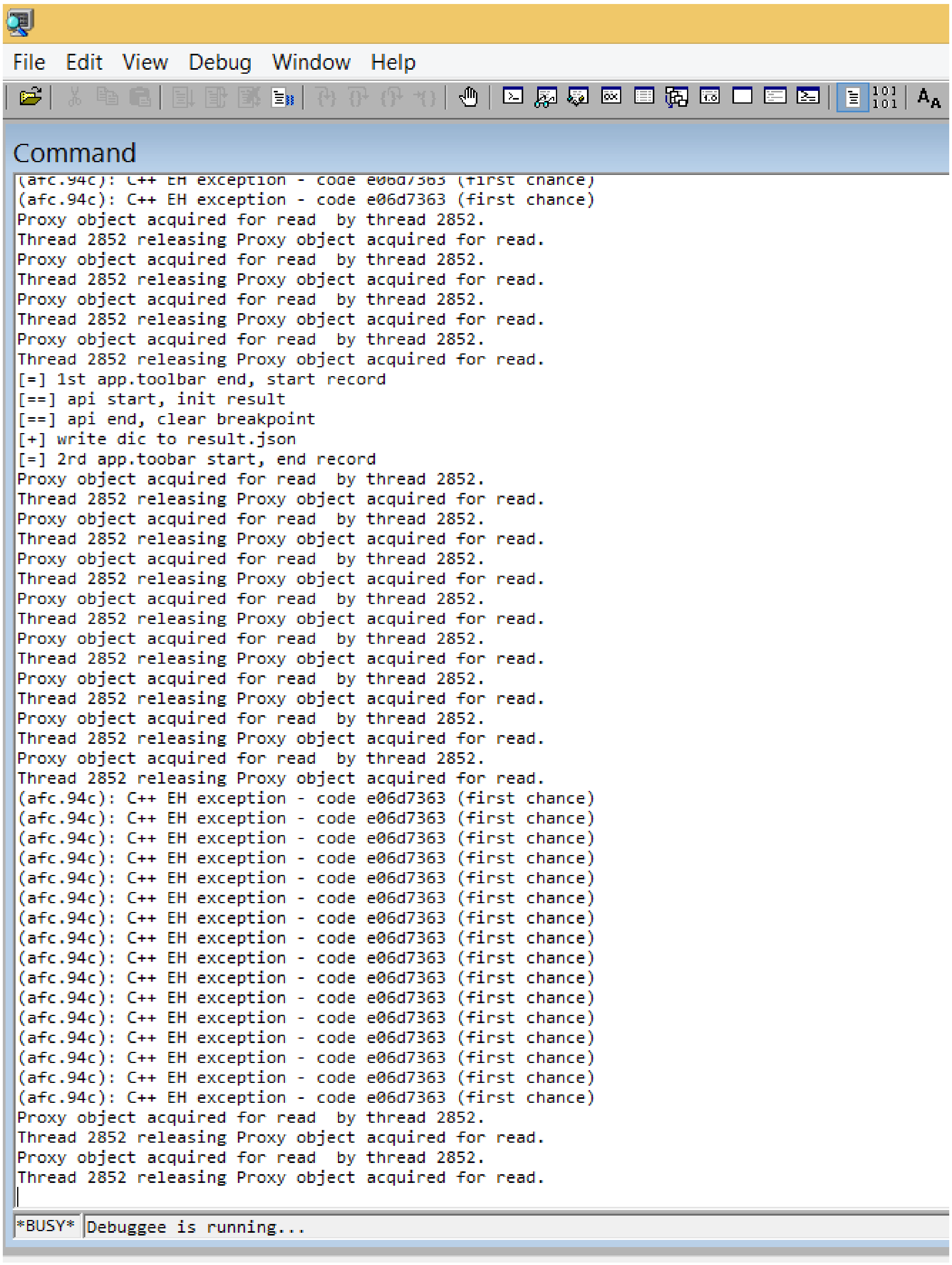
```
Command
0:009> .load pykd;!py -g C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2\record.py
[*] load 377 breakpoint
[*] EScript.api image base : 71180000
[=] 1st app.toolbar end, start record
[==] api start, init result
[==] api end, clear breakpoint
[+] write dic to result.json
[=] 2rd app.toobar start, end record
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.

*BUSY* Debuggee is running...
```



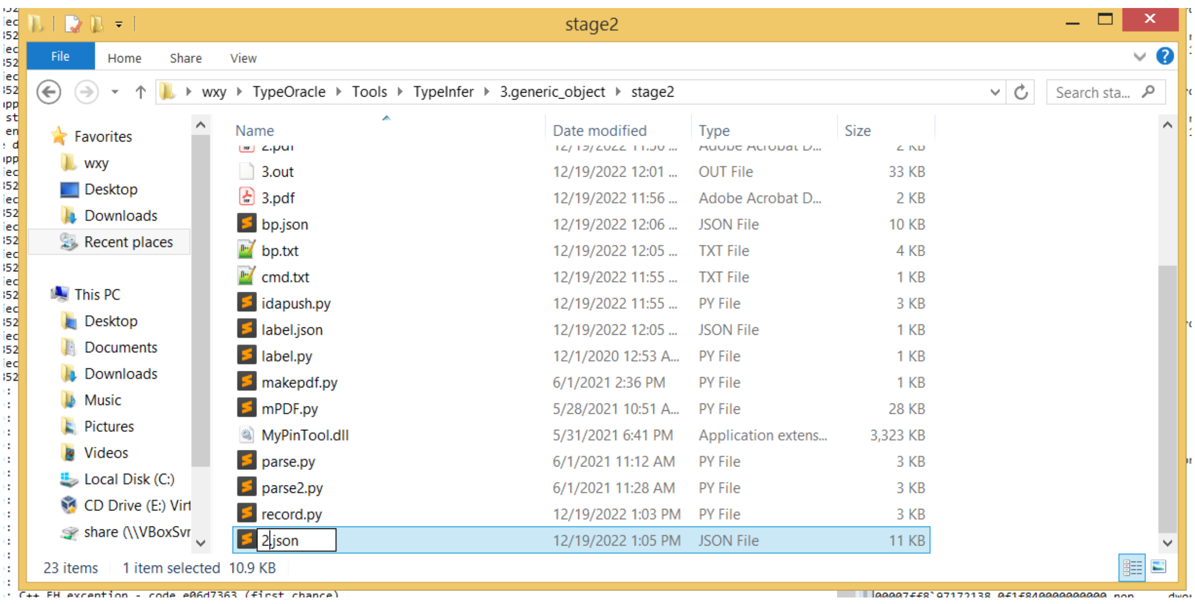
Name	Date modified	Type	Size
2.pdf	12/19/2022 11:50 ...	Adobe Acrobat D...	2 KB
3.out	12/19/2022 12:01 ...	OUT File	33 KB
3.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
bp.json	12/19/2022 12:06 ...	JSON File	10 KB
bp.txt	12/19/2022 12:05 ...	TXT File	4 KB
cmd.txt	12/19/2022 11:55 ...	TXT File	1 KB
idapush.py	12/19/2022 11:55 ...	PY File	3 KB
label.json	12/19/2022 12:05 ...	JSON File	1 KB
label.py	12/1/2020 12:53 A...	PY File	1 KB
makepdf.py	6/1/2021 2:36 PM	PY File	1 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:41 PM	Application extens...	3,323 KB
parse.py	6/1/2021 11:12 AM	PY File	3 KB
parse2.py	6/1/2021 11:28 AM	PY File	3 KB
record.py	12/19/2022 1:03 PM	PY File	3 KB
result.json	12/19/2022 1:03 PM	JSON File	11 KB

22 items   1 item selected 10.5 KB

open 2.pdf with Adobe Reader, this will generate result.json, rename it to 2.json



```
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
[=] 1st app.toolbar end, start record
[==] api start, init result
[==] api end, clear breakpoint
[+] write dic to result.json
[=] 2rd app.toobar start, end record
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.

*BUSY* Debuggee is running...
```



stage2

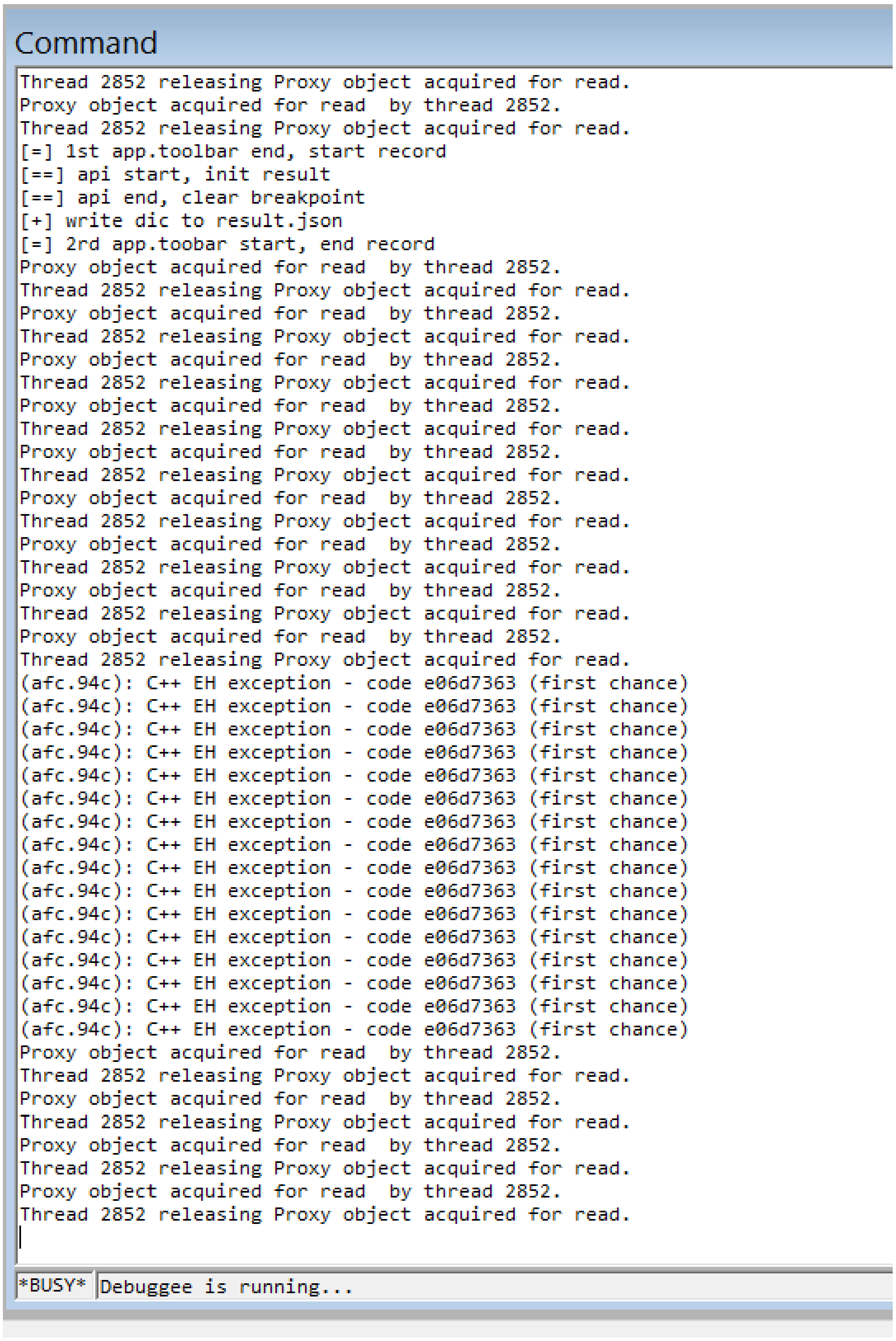
File Home Share View

wxy › TypeOracle › Tools › TypeInfer › 3.generic_object › stage2

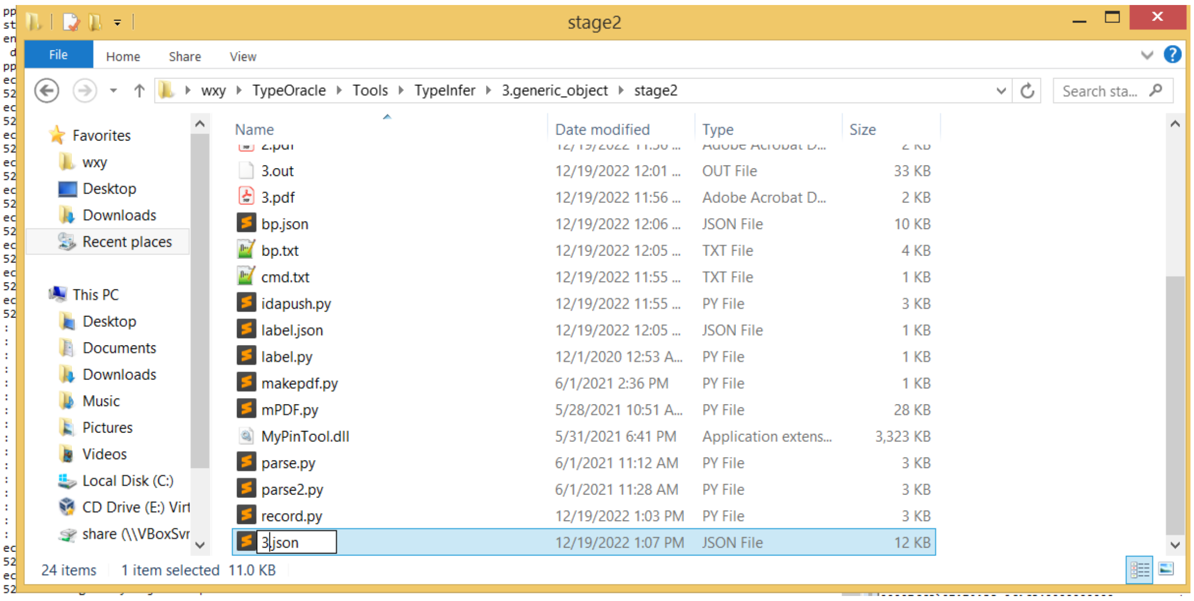
Name	Date modified	Type	Size
2.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
3.out	12/19/2022 12:01 ...	OUT File	33 KB
3.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
bp.json	12/19/2022 12:06 ...	JSON File	10 KB
bp.txt	12/19/2022 12:05 ...	TXT File	4 KB
cmd.txt	12/19/2022 11:55 ...	TXT File	1 KB
idapush.py	12/19/2022 11:55 ...	PY File	3 KB
label.json	12/19/2022 12:05 ...	JSON File	1 KB
label.py	12/1/2020 12:53 A...	PY File	1 KB
makepdf.py	6/1/2021 2:36 PM	PY File	1 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:41 PM	Application extens...	3,323 KB
parse.py	6/1/2021 11:12 AM	PY File	3 KB
parse2.py	6/1/2021 11:28 AM	PY File	3 KB
record.py	12/19/2022 1:03 PM	PY File	3 KB
2.json	12/19/2022 1:05 PM	JSON File	11 KB

23 items 1 item selected 10.9 KB

open 3.pdf with Adobe Reader, this will generate result.json, rename it to 3.json



```
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
[=] 1st app.toolbar end, start record
[==] api start, init result
[==] api end, clear breakpoint
[+] write dic to result.json
[=] 2rd app.toobar start, end record
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
(afc.94c): C++ EH exception - code e06d7363 (first chance)
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
Proxy object acquired for read  by thread 2852.
Thread 2852 releasing Proxy object acquired for read.
```



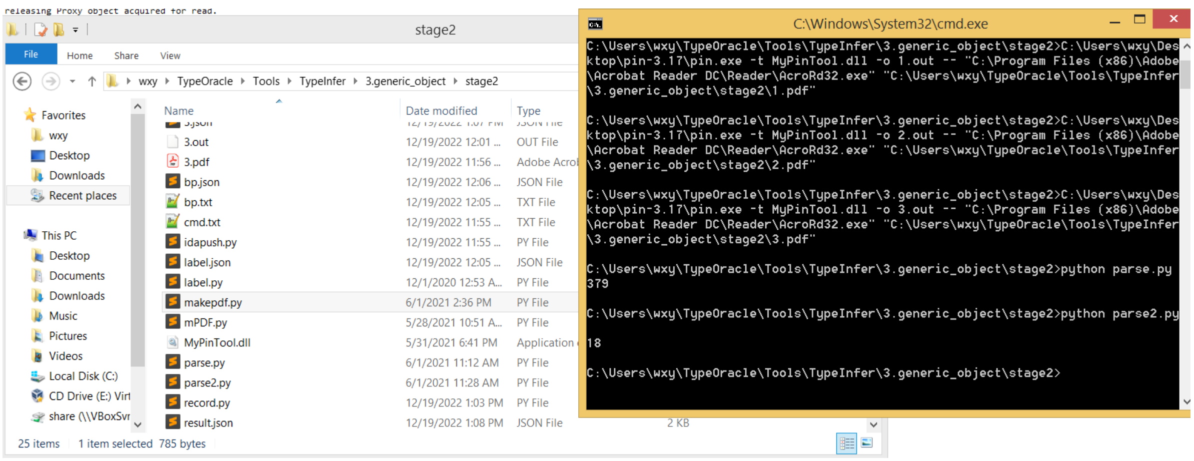
wxy › TypeOracle › Tools › TypeInfer › 3.generic_object › stage2

Name	Date modified	Type	Size
3.out	12/19/2022 12:01 ...	OUT File	33 KB
3.pdf	12/19/2022 11:56 ...	Adobe Acrobat D...	2 KB
bp.json	12/19/2022 12:06 ...	JSON File	10 KB
bp.txt	12/19/2022 12:05 ...	TXT File	4 KB
cmd.txt	12/19/2022 11:55 ...	TXT File	1 KB
idapush.py	12/19/2022 11:55 ...	PY File	3 KB
label.json	12/19/2022 12:05 ...	JSON File	1 KB
label.py	12/1/2020 12:53 A...	PY File	1 KB
makepdf.py	6/1/2021 2:36 PM	PY File	1 KB
mPDF.py	5/28/2021 10:51 A...	PY File	28 KB
MyPinTool.dll	5/31/2021 6:41 PM	Application extens...	3,323 KB
parse.py	6/1/2021 11:12 AM	PY File	3 KB
parse2.py	6/1/2021 11:28 AM	PY File	3 KB
record.py	12/19/2022 1:03 PM	PY File	3 KB
3.json	12/19/2022 1:07 PM	JSON File	12 KB

24 items 1 item selected 11.0 KB

6. execute parse2.py to get result.json, it contains the key-get indicators

```
python parse2.py
```



wxy › TypeOracle › Tools › TypeInfer › 3.generic_object › stage2

Name	Date modified	Type
3.out	12/19/2022 12:01 ...	OUT File
3.pdf	12/19/2022 11:56 ...	Adobe Acrol
bp.json	12/19/2022 12:06 ...	JSON File
bp.txt	12/19/2022 12:05 ...	TXT File
cmd.txt	12/19/2022 11:55 ...	TXT File
idapush.py	12/19/2022 11:55 ...	PY File
label.json	12/19/2022 12:05 ...	JSON File
label.py	12/1/2020 12:53 A...	PY File
makepdf.py	6/1/2021 2:36 PM	PY File
mPDF.py	5/28/2021 10:51 A...	PY File
MyPinTool.dll	5/31/2021 6:41 PM	Application
parse.py	6/1/2021 11:12 AM	PY File
parse2.py	6/1/2021 11:28 AM	PY File
record.py	12/19/2022 1:03 PM	PY File
result.json	12/19/2022 1:08 PM	JSON File

25 items 1 item selected 785 bytes

```
C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 1.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\1.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 2.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\2.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>C:\Users\wxy\Des
ktop\pin-3.17\pin.exe -t MyPinTool.dll -o 3.out -- "C:\Program Files (x86)\Adobe
\Acrobat Reader DC\Reader\AcroRd32.exe" "C:\Users\wxy\TypeOracle\Tools\TypeInfer
\3.generic_object\stage2\3.pdf"

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python parse.py
379

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>python parse2.py

18

C:\Users\wxy\TypeOracle\Tools\TypeInfer\3.generic_object\stage2>
```

4.arg_probe

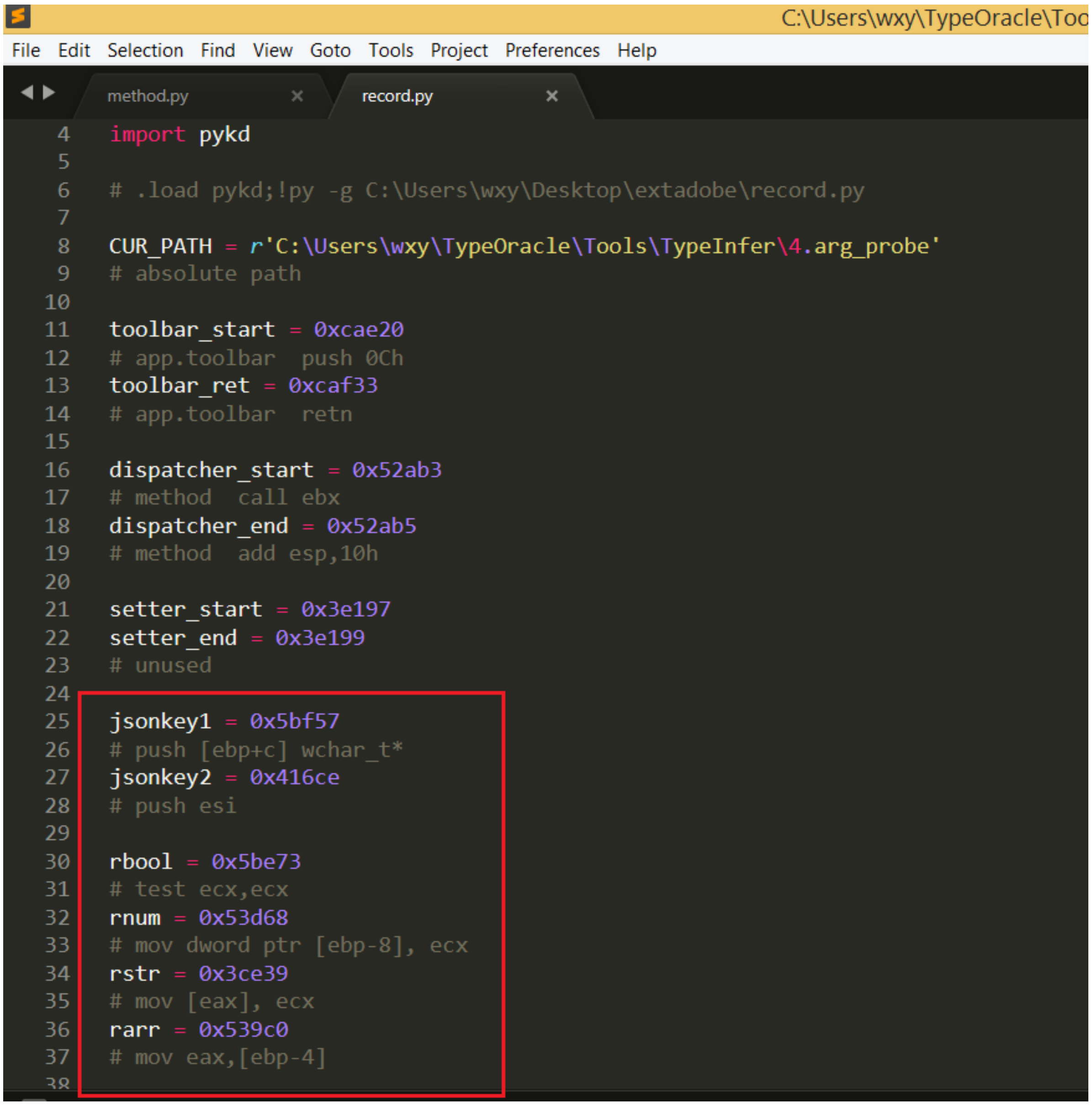
This step uses the type indicator extracted previously to obtain type information.

folder structure

```
- sample_result  (the folder store the sample result of this step)

- sample_tmp (the folder store sample intermediate results)

- 1.json-9.json

- funclst.txt (list of all methods)
- init.pdf (PDF file to init Adobe Reader)

- jstemplate.js

- method.py

- monitor.py (monitor the execution of PDF Reader)

- mPDF.py (generate PDF file)

- parse.py

- pattern.py (generate javascript code for one binding call)

- record.py (script executed in windbg)

- runtime.py

- s1.py (combine the results)

- setterlst.txt (list of all accessors)
```

how to reproduce

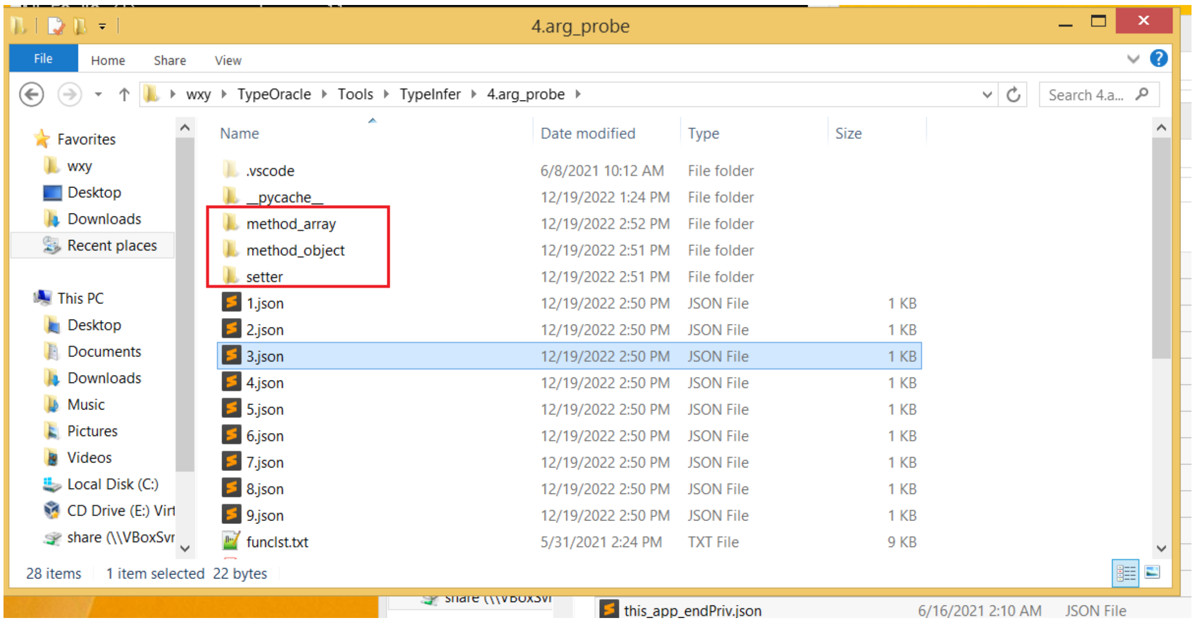
1. fill the type indicator infered before in `record.py`


C:\Users\wxy\TypeOracle\Too
File Edit Selection Find View Goto Tools Project Preferences Help
method.py
record.py

```python
import pykd

# .load pykd;!py -g C:\Users\wxy\Desktop\extadobe\record.py

CUR_PATH = r'C:\Users\wxy\TypeOracle\Tools\TypeInfer\4.arg_probe'
# absolute path

toolbar_start = 0xcae20
# app.toolbar  push 0Ch
toolbar_ret = 0xcaf33
# app.toolbar  retn

dispatcher_start = 0x52ab3
# method  call ebx
dispatcher_end = 0x52ab5
# method  add esp,10h

setter_start = 0x3e197
setter_end = 0x3e199
# unused

jsonkey1 = 0x5bf57
# push [ebp+c] wchar_t*
jsonkey2 = 0x416ce
# push esi

rbool = 0x5be73
# test ecx,ecx
rnum = 0x53d68
# mov dword ptr [ebp-8], ecx
rstr = 0x3ce39
# mov [eax], ecx
rarr = 0x539c0
# mov eax,[ebp-4]
```

2. execute `method.py` (it will take about five hours), this will generate three folders: `method_array`, `method_object`, `setter`

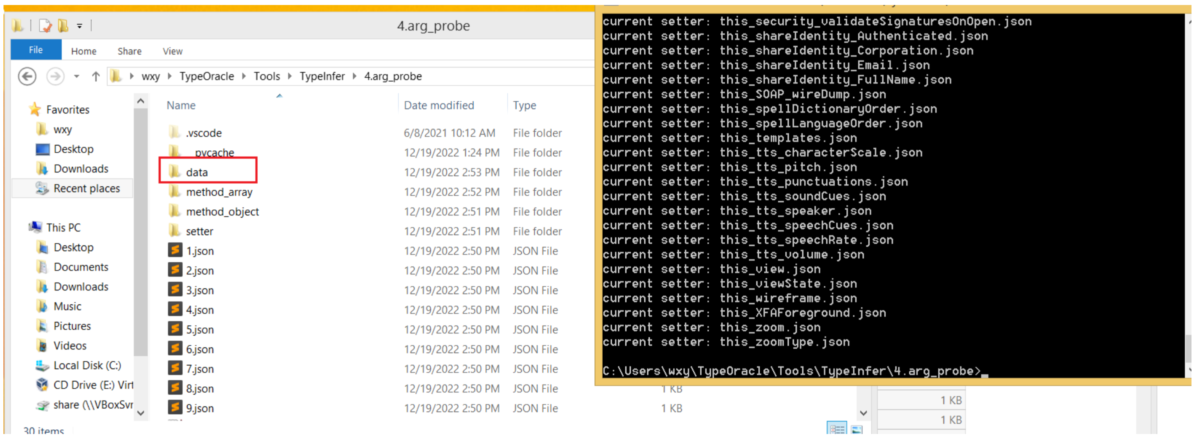
```
python method.py
```



```
4.arg_probe
wxy > TypeOracle > Tools > TypeInfer > 4.arg_probe

Name               Date modified        Type         Size
.vscode            6/8/2021 10:12 AM    File folder
__pycache__        12/19/2022 1:24 PM   File folder
method_array       12/19/2022 2:52 PM   File folder
method_object      12/19/2022 2:51 PM   File folder
setter             12/19/2022 2:51 PM   File folder
1.json             12/19/2022 2:50 PM   JSON File    1 KB
2.json             12/19/2022 2:50 PM   JSON File    1 KB
3.json             12/19/2022 2:50 PM   JSON File    1 KB
4.json             12/19/2022 2:50 PM   JSON File    1 KB
5.json             12/19/2022 2:50 PM   JSON File    1 KB
6.json             12/19/2022 2:50 PM   JSON File    1 KB
7.json             12/19/2022 2:50 PM   JSON File    1 KB
8.json             12/19/2022 2:50 PM   JSON File    1 KB
9.json             12/19/2022 2:50 PM   JSON File    1 KB
funclst.txt        5/31/2021 2:24 PM    TXT File     9 KB

28 items    1 item selected  22 bytes
```

3. execute s1.py to combine all results, the type information is store in `data` folder

```
python s1.py
```



```
4.arg_probe
wxy > TypeOracle > Tools > TypeInfer > 4.arg_probe

Name               Date modified        Type
.vscode            6/8/2021 10:12 AM    File folder
__pycache__        12/19/2022 1:24 PM   File folder
data               12/19/2022 2:53 PM   File folder
method_array       12/19/2022 2:52 PM   File folder
method_object      12/19/2022 2:51 PM   File folder
setter             12/19/2022 2:51 PM   File folder
1.json             12/19/2022 2:50 PM   JSON File
2.json             12/19/2022 2:50 PM   JSON File
3.json             12/19/2022 2:50 PM   JSON File
4.json             12/19/2022 2:50 PM   JSON File
5.json             12/19/2022 2:50 PM   JSON File
6.json             12/19/2022 2:50 PM   JSON File
7.json             12/19/2022 2:50 PM   JSON File
8.json             12/19/2022 2:50 PM   JSON File
9.json             12/19/2022 2:50 PM   JSON File

30 items
```

```
current setter: this_security_validateSignaturesOnOpen.json
current setter: this_shareIdentity_Authenticated.json
current setter: this_shareIdentity_Corporation.json
current setter: this_shareIdentity_Email.json
current setter: this_shareIdentity_FullName.json
current setter: this_SOAP_wireDump.json
current setter: this_spellDictionaryOrder.json
current setter: this_spellLanguageOrder.json
current setter: this_templates.json
current setter: this_tts_characterScale.json
current setter: this_tts_pitch.json
current setter: this_tts_punctuations.json
current setter: this_tts_soundCues.json
current setter: this_tts_speaker.json
current setter: this_tts_speechCues.json
current setter: this_tts_speechRate.json
current setter: this_tts_volume.json
current setter: this_view.json
current setter: this_viewState.json
current setter: this_wireframe.json
current setter: this_XFAForeground.json
current setter: this_zoom.json
current setter: this_zoomType.json

C:\Users\wxy\TypeOracle\Tools\TypeInfer\4.arg_probe>
```

illustration of type information

api: api name

apitype: 0-method 1-accessor

root: root object's index

info:

23x generic object

22x array

  req_type: required argument

  opt_type: optional argument

5x value: multiple types of parameters are allowed

0 bool

1 number

3 string

2 built-in object

5 any type