

This toolkit was developed since clinicaltrial.gov doesn't provide any access for searching in the main text of trial documents. The main idea of this tool is to download the trial documents via the XML files provided by clinical.gov which contain the area name of trial documents.

Therefore, the working process includes three main parts:

1. Search the clinicaltrial.gov with some sketchy conditions and download XML files of all potential qualified cases via clinicaltrial.gov manually.

42275 Studies found for: **Interventional Studies | Phase 3**

Applied Filters: ☒ Interventional ☒ Phase 3

Map Search Details

Showing: 1-10 of 42,275 studies 10 studies per page

Download Subscribe to RSS Show/Hide Columns

Row	Saved	Study Title	Sponsor/Collaborators	NCT Number	Study Start	First Posted
1	<input type="checkbox"/>	A Long-term Trial of ETC-1002 in Patients With Hyper-LDL Cholesterolemia	• Otsuka Pharmaceutical Co., Ltd.	NCT05687071	February 2023	January 16, 2023
2	<input type="checkbox"/>	A Study to Investigate the Safety and Efficacy of Undiluted Intravenous Infusion of I.V.-Hepabig In	• GC Biopharma Corp	NCT05686759	January 2023	January 16, 2023
3	<input type="checkbox"/>	mRNA Booster Vaccine(SW-BIC-213) Compared With Pfizer and Sinopharm Against Emerging VOCs	• Stemima Therapeutics	NCT05686161	October 19, 2022	January 16, 2023

*(all interventional phase 3 study)*

Download the search results for:  
**Interventional Studies | Phase 3 (42275 records)**

Need help? See [Downloading Content for Analysis](#)

Number of Studies:   
(PDF limited to 100 studies)

**Download Table Contents (1 row per study)**

Select table columns:

Select file format:

**For Advanced Users: Full Study Record XML Download**

- **Download** a zip file of the full study records in XML for all studies in the search results table (max 10000)
- To download the XML for all public study records in our database; See [Downloading All Study Content for Analysis](#)

*(Download a zip for all XML files, then unzip these files to another folder.)*

2. Read and parse the XML files with the python script titled "download PDF from XML file.py", the targeted PDF files will be downloaded through which.

```

def get_file_from_url(dire):
    from pathlib import Path
    import os
    import requests
    import io
    key = 'document_url'
    p = Path(dire)
    file_name = p.glob('*/*.xml')
    url_list = []
    title_list = []
    for i in file_name:
        response = requests.get(i)
        title = response.headers.get('Content-Disposition')
        title = title[0:11]
        title_list.append(title)
        for line in response.iter_lines():
            text = line.decode('utf-8')
            if key in text and "http" in text:
                start_text = text.find('http')
                end_text = text.find('.pdf')
    
```

Process finished with `exit` code 0

venv	2022/12/16 18:56	File folder	
.idea	2023/1/18 10:55	File folder	
searching key words in PDF file.py	2023/1/18 10:18	PY File	6 KB
search using pdf2p.py	2022/5/29 4:07	PY File	6 KB
download PDF from xml file.py	2022/12/16 17:19	PY File	3 KB
User guide for clinicaltrial.gov searching ...	2023/1/18 10:19	Microsoft Word ...	0 KB
sss	2022/12/17 15:37	File	1 KB
i+key	2022/3/25 3:23	File	1,106 KB
dsds	2022/12/17 14:39	File	0 KB
NCT03886025.pdf	2023/1/18 10:55	Adobe Acrobat 文...	309 KB
NCT03877627.pdf	2023/1/18 10:55	Adobe Acrobat 文...	290 KB
NCT03617445.pdf	2023/1/18 10:55	Adobe Acrobat 文...	8,053 KB
NCT03486444.pdf	2023/1/18 10:55	Adobe Acrobat 文...	2,923 KB
NCT03473483.pdf	2023/1/18 10:55	Adobe Acrobat 文...	793 KB
NCT03430973.pdf	2023/1/18 10:55	Adobe Acrobat 文...	357 KB
NCT01917604.pdf	2023/1/18 10:55	Adobe Acrobat 文...	1,891 KB

(Execute the python script, if it works, a similar log will appear, and you will find the downloaded PDF file in the same pathway as the script.)

- Comprehensive searching by the two scripts titled '*searching keywords in PDF file.py*' and '*search using pdf2p.py*', the corresponding searching results will be automatically sorted and stored.

*Notice: the search terms are input by list variable 'lisx', users could customize the search by modifying the search list.*

*Before the execution, please modify the absolute pathway variable 'drx' to customize the output.*

*The function of the two scripts '*searching keywords in PDF file.py*' and '*search using pdf2p.py*' are similar but were realized by different packages. Since PDF is not a stable data format, the author will recommend using the two scripts twice for quality control.*

```

import shutil

logging.propagate = False
logging.getLogger().setLevel(logging.ERROR)

lis1=['group sequential','conditional power','predictive power','conditional error','alpha spending','interim ana
lis2=['multi-arm multi-stage','multiarm multistage','multi stage']
lis3=['adaptive platform','adaptive platform']
lis4=['pick the winner','drop the loser','adaptive dose selection','seamless','adaptive design']
lis5=['adaptive enrichment','population enrichment','patient enrichment','enrichment design','biomarker adaptive
lis6=['response adaptive','adaptive randomisation','adaptive randomization','outcome adaptive']
lis7=['adaptive hypothesis','adaptive hypotheses']
lis8=['sample size adjustment','sample size re estimation','sample size reestimation','sample size modification','
        'sample size re assessment','sample size re-calculation','sample size recalculation']

dir = r'D:\search'
dr1 = r'D:\results\GSD'
dr2 = r'D:\results\MAMS'
dr3 = r'D:\results\Platform'
dr4 = r'D:\results\Dose selection'
dr5 = r'D:\results\Enrichment'
dr6 = r'D:\results\ARD'
dr7 = r'D:\results\Adaptive hypo'
dr8 = r'D:\results\sample size reestimation'
dr9 = r'D:\results\Not adaptive'

```

```

search using pdf2p
range(1, 167)
D:\search\NCT02422615.pdf
GSD was found
range(1, 69)
D:\search\NCT02509078.pdf
sample size reestimation was found
D:\search\NCT02509078.pdf
range(1, 163)
D:\search\NCT02569398.pdf
sample size reestimation was found
D:\search\NCT02569398.pdf
range(1, 73)
D:\search\NCT02586155.pdf

```

(an example of successful execution of 'search using pdf2p.py')

```

alpha spending
os.remove(i)
print('sample size reestimation was found')
break
else:
    shutil.copy(i, dr9)
    os.remove(i)
    print('key word was not found')
    break
except Exception as e:
    pass

```

```

for i in file_name:
    try:
        for key in lis1:
            else:
                for key in lis2:
                    else:
                        for key in lis3:
                            else:
                                for key in lis4:
                                    else:
                                        for key in lis5:
                                            else:
                                                for key in lis6:
                                                    else:
                                                        for key in lis7:
                                                            else:
                                                                for key in lis8:
                                                                    GSD was found
D:\search\NCT02358031.pdf
searching in lis1
GSD was found
D:\search\NCT02422186.pdf
searching in lis1
searching in lis2
searching in lis3
searching in lis4
searching in lis5
searching in lis6
searching in lis7
searching in lis8
D:\search\NCT02422615.pdf
searching in lis1
GSD was found
D:\search\NCT02431702.pdf
searching in lis1
searching in lis2
searching in lis3
searching in lis4
searching in lis5
searching in lis6
searching in lis7
searching in lis8
key word was not found

```

(an example of successful execution of 'searching keywords in PDF file.py')