

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな名前で置き換える
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*pdf')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217 |
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```

```
page = 0
"staring...")
in range(num_pages):
y:
    time.sleep(time_sleep)
# Take and save a screenshot
sc = pyautogui.screenshot()
sc.save("./" + title + '/page_{}.png'.format(i))
# Turn page
pyautogui.press(right_left)
# ページ数カウント
count_page += 1
percent = (count_page / num_pages) * 100
if percent % 10 == 0:
    print("saving... " + str(percent) + '%' + 
          str(count_page) + "/" + str(num_pages))
else:
    continue
cept KeyboardInterrupt:
    print("Ctrl+Cで停止しました")
    break
"test")
184     # pdfファイルを書き込む
185     file.write(cov_pdf)
186
187     # 開いているファイルを閉じる
188     img.close()
189     file.close()
190
191     # 複数のpdfファイルを結合する
192     pdf_path = "./" + title + "/pdf/"
193
194     merge = PyPDF2.PdfFileMerger()
195     for j in sorted(os.listdir(pdf_path), key=lambda s: int(s)):
196         merge.append(pdf_path + "/" + j)
197     merge.write("./" + title + '/' + title + '.pdf')
198     merge.close()
199     print("変換終了")
200
201     # すべて終了後、all.pdf以外のファイル、ディレクトリを削除する
202
203     # pdfの同じ名前のファイルが在ればリネームする
204     list_file_name = []
205     p_temp = pathlib.Path(path_dir).glob('*.*')
206     for p in p_temp:
207         list_file_name.append(p.name)
208     title_pdf = title + ".pdf"
209     if title_pdf in list_file_name: # 存在していれば、ランダムな
210         title_pdf = randomname(10) + ".pdf"
211     else:
212         pass
213
214     shutil.move("./" + title + '/' + title_pdf, path_dir)
215
216     shutil.rmtree("./" + title)
217
```