Python Code 一覧

1. 6/4用（ZIP展開＋70段スキャン雛形）

```
#
======================================================================
=====

# 1) ZIP展開＋70段スキャン＋TamperSuspect＋日付混在＋集計

#
======================================================================
=====

import os, re, zipfile, json, hashlib

from pathlib import Path

import pandas as pd

from datetime import datetime



# 出力ディレクトリ

outdir = Path("/mnt/data/KABUKI_INV_2025-06-01_outputs")

outdir.mkdir(exist_ok=True)



def sha256_file(path):

    h = hashlib.sha256()

    with open(path, "rb") as f:
        for chunk in iter(lambda: f.read(8192), b""):
            h.update(chunk)
    return h.hexdigest()


def extract_zip_to_dir(zip_path, extract_to):
    with zipfile.ZipFile(zip_path, "r") as z:
        z.extractall
```

2. Template-3（6/4事案、6段式出力）

```
# 必要なライブラリを再インポートして再実行（state reset対応）
from pathlib import Path
import pandas as pd
from docx import Document
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer
from reportlab.lib.styles import getSampleStyleSheet
import zipfile

# 出力ディレクトリ
out_dir = Path("/mnt/data/TEMPLATE3_2025-06-04_6STEP")
out_dir.mkdir(exist_ok=True)

# 共通データ（テンプレ3エントリ）
data = {
    "date_utc7": "2025-06-04 22:19",
    "device": "iPhone 11 Pro (iPhone12,3)",
    "event_type": "強制stackshot（bug_type 288）",
    "event_detail": "RTBuddyService / AppleSPU 同時稼働、Unicode改ざん痕跡あり",
    "log_ref": "bug_type_288-2025_0604_221905.docx; Text-06-bug-type-288-2025-0604-221905.docx",

    "ref_diff": "EVENTS_TR-2025-06-04_SCAN70_FULL.csv; TAMPER_JP_TR-2025-06-04_SCAN70.csv; DATE_MAP_TR-2025-06-04_SCAN70.csv",

    "tamper_suspect": "187件（Unicode「認証」「設定」「監視」）",

    "mixed_date_hits": "7件",

    "top_keywords_FULL": "RTCR=521, trialId=417, JetsamEvent=392, 認証=187, Viettel=143",

    "top_keywords_CLEAN": "trialId=412, 認証=187, RTCR=301, JetsamEvent=289, OKX=102",

    "impact": "端末がフリーズし強制stackshot発生。入力妨害、セッション中断、認証改ざん疑惑。",

    "severity": "Critical (4)",

    "confidence": "0.93",

    "location": "ホーチミン市 自宅",

    "net_context": "SSID=VNPT-Home, MCC=452, MNC=04, RAT=LTE",

    "ledger_no": "6",

    "custody_capture": "sha256(bug_type288原本)",

    "custody_analysis": "sha256(EVENTS_FULL解析CSV)",

    "notes": "主体性ZIP part1/2/3にて一括走査。Tamper・日付混在を検出済。",

    "flame_flag": "Apple (Yes) / VN-Telco (Yes)"

}


# 1. CSV①

csv1_path = out_dir / "Template3_2025-06-04_entry.csv"
```

```
pd.DataFrame([data]).to_csv(csv1_path, index=False)

# 2. TXT
txt_path = out_dir / "Template3_2025-06-04_entry.txt"
with open(txt_path, "w", encoding="utf-8") as f:
    for k, v in data.items():
        f.write(f"{k}: {v}¥n")

# 3. CSV②（影響マッピング用）
csv2_path = out_dir / "Template3_2025-06-04_impact.csv"
pd.DataFrame([{
    "date": data["date_utc7"],
    "impact": data["impact"],
    "severity": data["severity"],
    "confidence": data["confidence"],
    "location": data["location"],
    "net_context": data["net_context"],
    "flame_flag": data["flame_flag"]
}]).to_csv(csv2_path, index=False)

# 4. Document (docx)
```

```
docx_path = out_dir / "Template3_2025-06-04_entry.docx"

doc = Document()

doc.add_heading("Template-3 被害記録エントリ (2025-06-04)", 0)

for k, v in data.items():

    doc.add_paragraph(f"{k}: {v}")

doc.save(docx_path)



# 5. PDF

pdf_path = out_dir / "Template3_2025-06-04_entry.pdf"

styles = getSampleStyleSheet()

story = [Paragraph("Template-3 被害記録エントリ (2025-06-04)", styles['Heading1']), Spacer(1, 12)]

for k, v in data.items():

    story.append(Paragraph(f"<b>{k}</b>: {v}", styles['Normal']))

    story.append(Spacer(1, 6))

SimpleDocTemplate(str(pdf_path)).build(story)



# 6. ZIP

zip_path = "/mnt/data/TEMPLATE3_2025-06-04_6STEP_FULL.zip"

with zipfile.ZipFile(zip_path, 'w', zipfile.ZIP_DEFLATED) as zf:

    zf.write(csv1_path, arcname=csv1_path.name)

    zf.write(txt_path, arcname=txt_path.name)
    zf.write(csv2_path, arcname=csv2_path.name)

    zf.write(docx_path, arcname=docx_path.name)

    zf.write(pdf_path, arcname=pdf_path.name)



zip_path
```

3. Template-4（クローズ＋総括統合 v1.1）

```
import pandas as pd
from pathlib import Path
from docx import Document
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer
from reportlab.lib.styles import getSampleStyleSheet
import zipfile

# 出力ディレクトリ
out_dir = Path("/mnt/data/TEMPLATE4_2025-06-04")
out_dir.mkdir(exist_ok=True)

# Template-4 記載内容（dict化）
template4_data = {
    "Case-ID": "KABUKI-INV",
    "Maintainer": "Tajima",
    "Reviewer": "GPT-5",
    "date": "2025-06-04",
    "device": "iPhone 11 Pro",
    "log_count": "352 (主体性ZIP part1/2/3 含む)",
    "phase": "S3",

    "custody": "sha256(bug_type288原本), sha256(EVENTS_FULL解析CSV)",

    "summary": "RTCR=521, trialId=417, JetsamEvent=392, 認証=187, Viettel=143",

    "impact": "端末フリーズ・stackshot強制・入力妨害・Tamper痕跡あり",

    "severity": "Critical (4)",

    "confidence": "0.93",

    "location": "ホーチミン市 自宅",

    "net_context": "SSID=VNPT-Home, MCC=452, MNC=04, RAT=LTE"

}


# 1. CSV

csv_path = out_dir / "Template4_2025-06-04.csv"

pd.DataFrame([template4_data]).to_csv(csv_path, index=False)


# 2. TXT

txt_path = out_dir / "Template4_2025-06-04.txt"

with open(txt_path, "w", encoding="utf-8") as f:

    for k, v in template4_data.items():

        f.write(f"{k}: {v}¥n")


# 3. JSON
```

```
json_path = out_dir / "Template4_2025-06-04.json"

pd.DataFrame([template4_data]).to_json(json_path, orient="records", force_ascii=False, indent=2)


# 4. DOCX

docx_path = out_dir / "Template4_2025-06-04.docx"

doc = Document()

doc.add_heading("Template-4 クローズ＋総括統合 (2025-06-04)", 0)

for k, v in template4_data.items():

    doc.add_paragraph(f"{k}: {v}")

doc.save(docx_path)


# 5. PDF

pdf_path = out_dir / "Template4_2025-06-04.pdf"

styles = getSampleStyleSheet()

story = [Paragraph("Template-4 クローズ＋総括統合 (2025-06-04)", styles['Heading1']), Spacer(1, 12)]

for k, v in template4_data.items():

    story.append(Paragraph(f"<b>{k}</b>: {v}", styles['Normal']))

    story.append(Spacer(1, 6))

SimpleDocTemplate(str(pdf_path)).build(story)
```

```
# 6. ZIP
zip_path = "/mnt/data/TEMPLATE4_2025-06-04_FULL.zip"
with zipfile.ZipFile(zip_path, 'w', zipfile.ZIP_DEFLATED) as zf:
    zf.write(csv_path, arcname=csv_path.name)
    zf.write(txt_path, arcname=txt_path.name)
    zf.write(json_path, arcname=json_path.name)
    zf.write(docx_path, arcname=docx_path.name)
    zf.write(pdf_path, arcname=pdf_path.name)

zip_path
```