

```

import pandas as pd
import sys
import re

# -----
# Normalisation helpers
# -----

def clean_title(title):
    if pd.isna(title):
        return ""
    t = title.lower()
    t = re.sub(r'[^a-z0-9 ]+', ' ', t)
    t = re.sub(r'\s+', ' ', t).strip()
    return t

def clean_doi(doi):
    if pd.isna(doi):
        return ""
    doi = doi.strip().lower()
    doi = doi.replace("https://doi.org/", "")
    doi = doi.replace("http://dx.doi.org/", "")
    return doi

# -----
# Load and normalise a CSV
# -----

def load_db_csv(path):
    df = pd.read_csv(path)

    # Try common column names across databases
    title_cols = [c for c in df.columns if c.lower() in ["title",
    "article_title", "document title"]]
    doi_cols = [c for c in df.columns if "doi" in c.lower()]
    pmid_cols = [c for c in df.columns if "pmid" in c.lower()]

    df["TITLE_RAW"] = df[title_cols[0]] if title_cols else ""
    df["DOI_RAW"] = df[doi_cols[0]] if doi_cols else ""
    df["PMID_RAW"] = df[pmid_cols[0]] if pmid_cols else ""

    # Clean
    df["TITLE_CLEAN"] = df["TITLE_RAW"].apply(clean_title)
    df["DOI_CLEAN"] = df["DOI_RAW"].apply(clean_doi)
    df["PMID_CLEAN"] = df["PMID_RAW"].fillna("").astype(str)

    # Build a robust dedupe key:
    # Priority = DOI → PMID → Title
    df["DEDUP_KEY"] = (
        df["DOI_CLEAN"].replace("", pd.NA)
        .fillna(df["PMID_CLEAN"].replace("", pd.NA))
        .fillna(df["TITLE_CLEAN"])
    )

    return df

# -----
# Main merging + deduping logic

```

```

# -----
def main(pubmed_csv, cinahl_csv, wos_csv, output_csv):
    print("Loading files...")
    df_pm = load_db_csv(pubmed_csv)
    df_cn = load_db_csv(cinahl_csv)
    df_ws = load_db_csv(wos_csv)

    print("Merging...")
    merged = pd.concat([df_pm, df_cn, df_ws], ignore_index=True)

    print(f"Total before de-duplication: {len(merged)}")

    deduped = merged.drop_duplicates(subset=["DEDUP_KEY"], keep="first")

    print(f"Total after de-duplication: {len(deduped)}")

    # Export the full frame (not just clean columns)
    deduped.to_csv(output_csv, index=False)

    print(f"Saved de-duplicated file to {output_csv}")

# -----
# CLI entry
# -----
if __name__ == "__main__":
    if len(sys.argv) != 5:
        print("\nUsage:")
        print("  python merge_and_dedupe.py PubMed.csv CINAHL.csv WoS.csv")
        print("  output.csv\n")
        sys.exit(1)

    main(sys.argv[1], sys.argv[2], sys.argv[3], sys.argv[4])

```