import os

import xml.etree.ElementTree as ET

import json

def read\_references(input\_path, input\_format):

"""Reads references from different formats."""

references = []

if input\_format == "bibtex":

with open(input\_path, "r", encoding="utf-8") as file:

references = parse\_bibtex(file.read())

elif input\_format == "json":

with open(input\_path, "r", encoding="utf-8") as file:

references = json.load(file)

elif input\_format == "ris":

with open(input\_path, "r", encoding="utf-8") as file:

references = parse\_ris(file.read())

else:

raise ValueError("Unsupported format: " + input\_format)

return references

def parse\_bibtex(content):

"""Parse BibTeX content to a reference list."""

# Dummy parser - real implementation required

return [{"title": "Sample BibTeX Entry", "author": "John Doe"}]

def parse\_ris(content):

"""Parse RIS content to a reference list."""

# Dummy parser - real implementation required

return [{"title": "Sample RIS Entry", "author": "Jane Doe"}]

def convert\_to\_endnote\_format(references):

"""Converts a list of references to EndNote XML format."""

root = ET.Element("xml")

records = ET.SubElement(root, "records")

for ref in references:

record = ET.SubElement(records, "record")

for key, value in ref.items():

field = ET.SubElement(record, key)

field.text = value

return ET.tostring(root, encoding="utf-8", method="xml").decode("utf-8")

def write\_endnote\_file(output\_path, xml\_content):

"""Writes the EndNote XML content to a file."""

with open(output\_path, "w", encoding="utf-8") as file:

file.write(xml\_content)

def main():

input\_path = input("Enter the path of the reference file: ")

input\_format = input("Enter the input format (bibtex/json/ris): ").lower()

output\_path = input("Enter the output path for the EndNote file: ")

if not os.path.exists(input\_path):

print("Input file does not exist.")

return

try:

references = read\_references(input\_path, input\_format)

endnote\_xml = convert\_to\_endnote\_format(references)

write\_endnote\_file(output\_path, endnote\_xml)

print("Conversion complete! EndNote file saved at:", output\_path)

except Exception as e:

print("An error occurred:", e)

if \_\_name\_\_ == "\_\_main\_\_":

main()