Padronização e Consolidação dos Microdados do CENSUP

Este código realiza a primeira etapa no processo de padronização dos microdados do CENSUP, filtrando e organizando dados de cursos de uma instituição específica. Primeiro, ele identifica arquivos CSV em uma pasta de entrada chamada "Pasta MicroDados Cursos". Para cada arquivo, ele carrega o conteúdo usando o pandas, mantendo todos os dados como strings. Em seguida, ele filtra apenas as linhas em que o valor da coluna CO_IES é igual a '719', representando uma instituição de interesse específica. Se houver linhas correspondentes, o código salva o conteúdo filtrado em uma nova pasta, "Linhas Filtradas", garantindo que somente os dados que o usuário deseja sejam mantidos e facilitando o próximo passo na padronização.

```python
import pandas as pd
import os
from tqdm import tqdm

pasta_planilhas = 'Pasta MicroDados Cursos'
pasta_saida = 'Aquivos Contatenados'
os.makedirs(pasta_saida, exist_ok=True)

arquivos = [f for f in os.listdir(pasta_planilhas) if f.endswith('.xlsx')]

for arquivo in tqdm(arquivos, desc='Processando arquivos', unit='arquivo'):
    caminho_arquivo = os.path.join(pasta_planilhas, arquivo)

    df = pd.read_excel(caminho_arquivo, dtype=str)

    df_filtrado = df[df['CO_IES'] == '719']

    if not df_filtrado.empty:
        caminho_arquivo_saida = os.path.join(pasta_saida, arquivo)

        df_filtrado.to_excel(caminho_arquivo_saida, index=False)

        print(f"Arquivo filtrado salvo: {caminho_arquivo_saida}")


print("Filtragem concluída!")
```

Este código executa a segunda etapa no processo de padronização dos microdados do CENSUP, consolidando os dados previamente filtrados em um único arquivo. A partir da

pasta "Linhas Filtradas", o código identifica todos os arquivos CSV e abre cada um usando o pandas, armazenando-os em uma lista de dataframes. Em seguida, ele concatena esses dataframes em um único conjunto de dados, mantendo a sequência das linhas ao desconsiderar os índices originais. Por fim, o código salva o dataframe resultante em um arquivo Excel chamado "planilha_concatenada.xlsx", facilitando a análise e a integração dos microdados em um formato padronizado.

```python
import pandas as pd
import os

caminho = 'Linhas Filtradas'

dataframes = []

for arquivo in os.listdir(caminho):
    if arquivo.endswith('.csv'):
        caminho_arquivo = os.path.join(caminho, arquivo)
        df = pd.read_excel(caminho_arquivo)
        dataframes.append(df)
    df_concatenado = pd.concat(dataframes, ignore_index=True)

df_concatenado.to_excel('planilha_concatenada.xlsx', index=False)
```

Este código executa a terceira etapa do processo de padronização dos microdados do CENSUP, aplicando uma normalização textual para facilitar a análise e padronizar os dados. Ele abre o arquivo "planilha_concatenada.xlsx" e seleciona colunas específicas relacionadas a nomes de municípios, cursos e áreas de conhecimento. Para cada uma dessas colunas, o código remove os acentos e converte os textos para letras maiúsculas, usando a biblioteca `unidecode`, que facilita a remoção de acentos e caracteres especiais. Por fim, o arquivo atualizado é salvo como "planilha_sem_acentuacao.xlsx", garantindo que todos os textos estejam padronizados para futuras etapas de análise e integração.

```python
import pandas as pd
from unidecode import unidecode

df = pd.read_excel('planilha_concatenada.xlsx')

colunas_para_modificar = [
    'NO_MUNICIPIO',
    'NO_CURSO',
    'NO_CINE_ROTULO',
    'NO_CINE_AREA_GERAL',
    'NO_CINE_AREA_ESPECIFICA',
```

```
    'NO_CINE_AREA_DETALHADA'
]

for coluna in colunas_para_modificar:
    df[coluna] = df[coluna].apply(lambda x: unidecode(str(x)).upper())

df.to_excel('planilha_sem_acentuacao.xlsx', index=False)
```

Este código implementa a quarta etapa do processo de padronização dos microdados do CENSUP, reduzindo o conjunto de dados às colunas de interesse para simplificar a análise e focar nas variáveis relevantes. Ele carrega o arquivo "'planilha_sem_acentuacao.xlsx" e define uma lista de colunas desejadas, que incluem informações como ano do censo, município, curso, modalidades de ensino e outras métricas específicas de vagas, inscrições, matrículas e conclusões, além de dados sobre diversidade e acessibilidade. A seguir, o código filtra o DataFrame para conter apenas essas colunas e salva o resultado em um novo arquivo Excel, "planilha_colunas_filtrada.xlsx", otimizando o arquivo final para as próximas etapas do processo.

Python

```
import pandas as pd

df = pd.read_excel('planilha_sem_acentuacao.xlsx')

colunas_desejadas = [
    'NU_ANO_CENSO', 'NO_MUNICIPIO', 'NO_CURSO', 'CO_CURSO',
'NO_CINE_ROTULO',
    'NO_CINE_AREA_GERAL', 'TP_MODALIDADE_ENSINO', 'TP_GRAU_ACADEMICO',
'QT_CURSO',
    'QT_VG_TOTAL', 'QT_VG_TOTAL_DIURNO', 'QT_VG_TOTAL_NOTURNO',
'QT_VG_TOTAL_EAD',
    'QT_VG_NOVA', 'QT_VG_PROC_SELETIVO', 'QT_VG_REMANESC',
'QT_VG_PROG_ESPECIAL',
    'QT_INSCRITO_TOTAL', 'QT_INSCRITO_TOTAL_DIURNO',
'QT_INSCRITO_TOTAL_NOTURNO',
    'QT_INSCRITO_TOTAL_EAD', 'QT_INSC_VG_NOVA', 'QT_INSC_PROC_SELETIVO',
    'QT_INSC_VG_REMANESC', 'QT_INSC_VG_PROG_ESPECIAL', 'QT_ING',
'QT_ING_FEM',
    'QT_ING_MASC', 'QT_ING_DIURNO', 'QT_ING_NOTURNO', 'QT_ING_VG_NOVA',
    'QT_ING_VESTIBULAR', 'QT_ING_ENEM', 'QT_ING_AVALIACAO_SERIADA',
    'QT_ING_SELECAO_SIMPLIFICA', 'QT_ING_EGR', 'QT_ING_OUTRO_TIPO_SELECAO',
    'QT_ING_PROC_SELETIVO', 'QT_ING_VG_REMANESC', 'QT_ING_VG_PROG_ESPECIAL',
    'QT_ING_OUTRA_FORMA', 'QT_ING_0_17', 'QT_ING_18_24', 'QT_ING_25_29',
    'QT_ING_30_34', 'QT_ING_35_39', 'QT_ING_40_49', 'QT_ING_50_59',
'QT_ING_60_MAIS',
    'QT_ING_BRANCA', 'QT_ING_PRETA', 'QT_ING_PARDA', 'QT_ING_AMARELA',
'QT_ING_INDIGENA',
```

```
    'QT_ING_CORND', 'QT_MAT', 'QT_MAT_FEM', 'QT_MAT_MASC', 'QT_MAT_DIURNO',
    'QT_MAT_NOTURNO', 'QT_MAT_0_17', 'QT_MAT_18_24', 'QT_MAT_25_29',
'QT_MAT_30_34',
    'QT_MAT_35_39', 'QT_MAT_40_49', 'QT_MAT_50_59', 'QT_MAT_60_MAIS',
'QT_MAT_BRANCA',
    'QT_MAT_PRETA', 'QT_MAT_PARDA', 'QT_MAT_AMARELA', 'QT_MAT_INDIGENA',
'QT_MAT_CORND',
    'QT_CONC', 'QT_CONC_FEM', 'QT_CONC_MASC', 'QT_CONC_DIURNO',
'QT_CONC_NOTURNO',
    'QT_CONC_0_17', 'QT_CONC_18_24', 'QT_CONC_25_29', 'QT_CONC_30_34',
'QT_CONC_35_39',
    'QT_CONC_50_59', 'QT_CONC_60_MAIS', 'QT_CONC_BRANCA', 'QT_CONC_PRETA',
    'QT_CONC_PARDA', 'QT_CONC_AMARELA', 'QT_CONC_INDIGENA', 'QT_CONC_CORND',
    'QT_ALUNO_DEFICIENTE', 'QT_ING_DEFICIENTE', 'QT_MAT_DEFICIENTE',
'QT_CONC_DEFICIENTE',
    'QT_ING_FINANC', 'QT_ING_PROUNII', 'QT_ING_PROUNIP',
'QT_ING_RESERVA_VAGA',
    'QT_ING_RVREDEPUBLICA', 'QT_ING_RVETNICO', 'QT_ING_RVPDEF',
'QT_ING_RVSOCIAL_RF',
    'QT_ING_RVOUTROS', 'QT_MAT_RESERVA_VAGA', 'QT_MAT_RVREDEPUBLICA',
'QT_MAT_RVETNICO',
    'QT_MAT_RVPDEF', 'QT_MAT_RVSOCIAL_RF', 'QT_MAT_RVOUTROS',
'QT_CONC_RESERVA_VAGA',
    'QT_CONC_RVREDEPUBLICA', 'QT_CONC_RVETNICO', 'QT_CONC_RVPDEF',
'QT_CONC_RVSOCIAL_RF',
    'QT_CONC_RVOUTROS', 'QT_SIT_TRANCADA', 'QT_SIT_DESVINCULADO',
'QT_SIT_TRANSFERIDO',
    'QT_SIT_FALECIDO', 'QT_ING_PROCESCPUBLICA', 'QT_ING_PROCESCPRIVADA',
    'QT_ING_PROCNAOINFORMADA', 'QT_MAT_PROCESCPUBLICA',
'QT_MAT_PROCESCPRIVADA',
    'QT_MAT_PROCNAOINFORMADA', 'QT_CONC_PROCESCPUBLICA',
'QT_CONC_PROCESCPRIVADA',
    'QT_CONC_PROCNAOINFORMADA', 'QT_APOIO_SOCIAL', 'QT_ING_APOIO_SOCIAL',
    'QT_MAT_APOIO_SOCIAL', 'QT_CONC_APOIO_SOCIAL'
]

df_filtrado = df[colunas_desejadas]
df_filtrado.to_excel('planilha_colunas_filtrada.xlsx', index=False)
```

Na quinta etapa do processo de padronização dos microdados do CENSUP, este código realiza uma limpeza essencial ao remover linhas sem identificação de município. Ele carrega o arquivo "'planilha_colunas_filtrada.xlsx" e filtra o DataFrame, mantendo apenas as linhas em que a coluna `NO_MUNICIPIO` não está vazia. Essa filtragem é necessária porque linhas sem valor nesta coluna não podem ser identificadas, prejudicando a integridade dos dados. O DataFrame resultante, agora sem linhas inválidas, é então salvo no mesmo arquivo Excel, garantindo uma versão atualizada e mais precisa para uso nas etapas subsequentes.

```python
import pandas as pd

df = pd.read_excel('planilha_colunas_filtrada.xlsx')

df_limpo = df[df['NO_MUNICIPIO'].notna()]

df_limpo.to_excel('planilha_sem_linhas_NO_MUNICIPIO_vazias.xlsx',
index=False)
```

Na sexta última etapa do processo de padronização dos microdados do CENSUP, este código substitui valores ausentes nas colunas especificadas por zeros, padronizando a presença de dados. Primeiramente, ele carrega o arquivo "planilha_sem_linhas_NO_MUNICIPIO_vazias.xlsx" e, em seguida, verifica uma lista extensa de colunas consideradas relevantes. Qualquer valor ausente nessas colunas é substituído por zero, garantindo consistência nos dados numéricos, que ficam sem lacunas. Após esse preenchimento, o DataFrame atualizado é salvo no mesmo arquivo Excel.

```python
import pandas as pd
from openpyxl import load_workbook

df = pd.read_excel('planilha_sem_linhas_NO_MUNICIPIO_vazias.xlsx')

colunas_para_verificar = [
    'NU_ANO_CENSO', 'NO_MUNICIPIO', 'NO_CURSO', 'CO_CURSO',
'NO_CINE_ROTULO',
    'NO_CINE_AREA_GERAL', 'TP_MODALIDADE_ENSINO', 'TP_GRAU_ACADEMICO',
'QT_CURSO',
    'QT_VG_TOTAL', 'QT_VG_TOTAL_DIURNO', 'QT_VG_TOTAL_NOTURNO',
'QT_VG_TOTAL_EAD',
    'QT_VG_NOVA', 'QT_VG_PROC_SELETIVO', 'QT_VG_REMANESC',
'QT_VG_PROG_ESPECIAL',
    'QT_INSCRITO_TOTAL', 'QT_INSCRITO_TOTAL_DIURNO',
'QT_INSCRITO_TOTAL_NOTURNO',
    'QT_INSCRITO_TOTAL_EAD', 'QT_INSC_VG_NOVA', 'QT_INSC_PROC_SELETIVO',
    'QT_INSC_VG_REMANESC', 'QT_INSC_VG_PROG_ESPECIAL', 'QT_ING',
'QT_ING_FEM',
    'QT_ING_MASC', 'QT_ING_DIURNO', 'QT_ING_NOTURNO', 'QT_ING_VG_NOVA',
    'QT_ING_VESTIBULAR', 'QT_ING_ENEM', 'QT_ING_AVALIACAO_SERIADA',
```

```
    'QT_ING_SELECAO_SIMPLIFICA', 'QT_ING_EGR', 'QT_ING_OUTRO_TIPO_SELECAO',
    'QT_ING_PROC_SELETIVO', 'QT_ING_VG_REMANESC', 'QT_ING_VG_PROG_ESPECIAL',
    'QT_ING_OUTRA_FORMA', 'QT_ING_0_17', 'QT_ING_18_24', 'QT_ING_25_29',
    'QT_ING_30_34', 'QT_ING_35_39', 'QT_ING_40_49', 'QT_ING_50_59',
'QT_ING_60_MAIS',
    'QT_ING_BRANCA', 'QT_ING_PRETA', 'QT_ING_PARDA', 'QT_ING_AMARELA',
'QT_ING_INDIGENA',
    'QT_ING_CORND', 'QT_MAT', 'QT_MAT_FEM', 'QT_MAT_MASC', 'QT_MAT_DIURNO',
    'QT_MAT_NOTURNO', 'QT_MAT_0_17', 'QT_MAT_18_24', 'QT_MAT_25_29',
'QT_MAT_30_34',
    'QT_MAT_35_39', 'QT_MAT_40_49', 'QT_MAT_50_59', 'QT_MAT_60_MAIS',
'QT_MAT_BRANCA',
    'QT_MAT_PRETA', 'QT_MAT_PARDA', 'QT_MAT_AMARELA', 'QT_MAT_INDIGENA',
'QT_MAT_CORND',
    'QT_CONC', 'QT_CONC_FEM', 'QT_CONC_MASC', 'QT_CONC_DIURNO',
'QT_CONC_NOTURNO',
    'QT_CONC_0_17', 'QT_CONC_18_24', 'QT_CONC_25_29', 'QT_CONC_30_34',
'QT_CONC_35_39',
    'QT_CONC_50_59', 'QT_CONC_60_MAIS', 'QT_CONC_BRANCA', 'QT_CONC_PRETA',
    'QT_CONC_PARDA', 'QT_CONC_AMARELA', 'QT_CONC_INDIGENA', 'QT_CONC_CORND',
    'QT_ALUNO_DEFICIENTE', 'QT_ING_DEFICIENTE', 'QT_MAT_DEFICIENTE',
'QT_CONC_DEFICIENTE',
    'QT_ING_FINANC', 'QT_ING_PROUNII', 'QT_ING_PROUNIP',
'QT_ING_RESERVA_VAGA',
    'QT_ING_RVREDEPUBLICA', 'QT_ING_RVETNICO', 'QT_ING_RVPDEF',
'QT_ING_RVSOCIAL_RF',
    'QT_ING_RVOUTROS', 'QT_MAT_RESERVA_VAGA', 'QT_MAT_RVREDEPUBLICA',
'QT_MAT_RVETNICO',
    'QT_MAT_RVPDEF', 'QT_MAT_RVSOCIAL_RF', 'QT_MAT_RVOUTROS',
'QT_CONC_RESERVA_VAGA',
    'QT_CONC_RVREDEPUBLICA', 'QT_CONC_RVETNICO', 'QT_CONC_RVPDEF',
'QT_CONC_RVSOCIAL_RF',
    'QT_CONC_RVOUTROS', 'QT_SIT_TRANCADA', 'QT_SIT_DESVINCULADO',
'QT_SIT_TRANSFERIDO',
    'QT_SIT_FALECIDO', 'QT_ING_PROCESCPUBLICA', 'QT_ING_PROCESCPRIVADA',
    'QT_ING_PROCNAOINFORMADA', 'QT_MAT_PROCESCPUBLICA',
'QT_MAT_PROCESCPRIVADA',
    'QT_MAT_PROCNAOINFORMADA', 'QT_CONC_PROCESCPUBLICA',
'QT_CONC_PROCESCPRIVADA',
    'QT_CONC_PROCNAOINFORMADA', 'QT_APOIO_SOCIAL', 'QT_ING_APOIO_SOCIAL',
    'QT_MAT_APOIO_SOCIAL', 'QT_CONC_APOIO_SOCIAL'
]

df[colunas_para_verificar] = df[colunas_para_verificar].fillna(0)

df.to_excel('planilha_sem_linhas_NO_MUNICIPIO_vazias.xlsx', index=False)

wb = load_workbook('planilha_sem_linhas_NO_MUNICIPIO_vazias.xlsx')
```

```
ws = wb.active
wb.save('planilha_com_dados_vazios_substituidos.xlsx')
```