

```

In [18]: import requests
import pandas as pd
from bs4 import BeautifulSoup
import os
import time # Agregamos la importación del módulo time para pausas

# URL de la página
url = 'https://www.sinhumo.net/1640-vaper'

# Realizar la solicitud GET a la página
response = requests.get(url)

# Verificar si la solicitud fue exitosa (código de estado 200)
if response.status_code == 200:
    # Parsear el contenido HTML con BeautifulSoup
    soup = BeautifulSoup(response.text, 'html.parser')

    # Pequeña pausa antes de intentar hacer clic en el botón "Aceptar"
    time.sleep(2)

    # Encuentra y haz clic en el botón "Aceptar" si es necesario
    accept_button = soup.find('a', {'id': 'adultcontentOK'})
    if accept_button:
        try:
            accept_button.click()
            print("Se hizo clic en 'Aceptar'")
        except Exception as e:
            print(f"No se pudo hacer clic en 'Aceptar': {e}")

    # Encuentra todos los elementos de artículo
    articles = soup.select('article.product-miniature')

    # Crear listas para almacenar los resultados
    data = {'Artículo': [], 'Precio': [], 'Enlace': []}

    # Iterar sobre los elementos de artículo
    for article in articles:
        # Obtener el título del artículo
        title_element = article.select_one('div.product-description a')
        article_title = title_element.text.strip()

        # Obtener el precio del artículo
        price_element = article.select_one('span.price')
        article_price = price_element.text.strip()

        # Obtener el enlace del artículo
        link_element = article.select_one('div.product-description a')
        article_link = link_element['href']

        # Almacenar resultados en listas
        data['Artículo'].append(article_title)
        data['Precio'].append(article_price)
        data['Enlace'].append(article_link)

    # Convertir a DataFrame de pandas
    df = pd.DataFrame(data)

    # Obtener la ruta del directorio de usuario
    user_directory = os.path.expanduser("~")

    # Guardar en CSV en el directorio de usuario
    csv_path = os.path.join(user_directory, 'resultados.csv')
    df.to_csv(csv_path, index=False)

    # Imprimir mensaje de éxito
    print(f"Resultados guardados en CSV en: {csv_path}")
else:
    print(f"Error al obtener la página. Código de estado: {response.status_code}")

```

Resultados guardados en CSV en: C:\Users\VORPC\resultados.csv

In []: