```python
import numpy as np
import pandas as pd
from datetime import datetime
from scipy.interpolate import interp1d

# Charger les données
df = pd.read_csv("Nat_Gas.csv")
df['Dates'] = pd.to_datetime(df['Dates'], format="%m/%d/%y")
df = df.sort_values('Dates')
start_date = df['Dates'].min()
df['Days'] = (df['Dates'] - start_date).dt.days

# Création de l'interpolateur
price_function = interp1d(df['Days'], df['Prices'], kind='cubic',
fill_value="extrapolate")

def get_price(date_str):
    """Retourne le prix estimé à une date donnée."""
    date = pd.to_datetime(date_str)
    days = (date - start_date).days
    return float(price_function(days))

def price_storage_contract(injection_dates, withdrawal_dates,
volume_per_day,
                           max_volume, storage_cost_per_month,
                           injection_fee_per_volume,
withdrawal_fee_per_volume):
    """
    Calcule la valeur nette d'un contrat de stockage de gaz naturel.

    Paramètres :
    - injection_dates (list of str) : dates d'injection (ex. : ['2024-
06-01'])
    - withdrawal_dates (list of str) : dates de retrait (ex. : ['2024-
12-01'])
    - volume_per_day (float) : volume injecté ou retiré chaque jour
(en MMBtu)
    - max_volume (float) : capacité maximale de stockage (en MMBtu)
    - storage_cost_per_month (float) : coût mensuel fixe du stockage
($)
    - injection_fee_per_volume (float) : coût unitaire d'injection
($/MMBtu)
    - withdrawal_fee_per_volume (float) : coût unitaire de retrait
($/MMBtu)

    Retour :
    - float : valeur estimée du contrat en dollars
    """
    assert len(injection_dates) == len(withdrawal_dates), "Chaque
injection doit avoir un retrait."
```

```
    total_value = 0

    for inject_date, withdraw_date in zip(injection_dates,
withdrawal_dates):
        date_in = pd.to_datetime(inject_date)
        date_out = pd.to_datetime(withdraw_date)
        duration_days = (date_out - date_in).days

        if duration_days <= 0:
            raise ValueError(f"La date de retrait {withdraw_date} doit
être après l'injection {inject_date}.")

        volume = min(volume_per_day * duration_days, max_volume)

        buy_price = get_price(inject_date)
        sell_price = get_price(withdraw_date)

        gross_profit = (sell_price - buy_price) * volume
        storage_cost = storage_cost_per_month * (duration_days / 30)
        fees = injection_fee_per_volume * volume +
withdrawal_fee_per_volume * volume

        net_value = gross_profit - storage_cost - fees
        total_value += net_value

    return round(total_value, 2)

price_storage_contract(
    injection_dates=["2024-06-01"],
    withdrawal_dates=["2024-12-01"],
    volume_per_day=5000,
    max_volume=500000,
    storage_cost_per_month=100000,
    injection_fee_per_volume=0.01,
    withdrawal_fee_per_volume=0.01
)
```

```
2029153.36
```