

Kode Program

```
[20] def Trapezoid(a,b,f):  
    ...  
    Fungsi untuk mencari Integral Trapezoid dengan mengganti nilai  
    a = batas atas  
    dan  
    b = batas bawah,  
    serta  
    f = yang akan diintegralkan  
    ...  
    n = 100  
    def trapezoid(f,a,b,n=100):  
        h = (b-a)/n  
        sum = 0.0  
        for i in range(1,n):  
            x = a+i*h  
            sum = sum + f(x)  
            integral = (h/2)*(f(a)+2*sum+f(b)) # Rumus Trapezoid  
        return integral  
    integral = trapezoid(f,a,b,n)  
    print(a,"",b,"",round(integral,2))
```

```
[20] # Membuat database dari fungsi-fungsi  
functions = [  
    lambda x: 2*x,  
    lambda x: 2*x + 2,  
    lambda x: 2*x + 4,  
    lambda x: 4*x + 6,  
    lambda x: 6*x + 8,  
    lambda x: 8*x + 10,  
    lambda x: 10*x + 12,  
    lambda x: 12*x + 14,  
    lambda x: 14*x + 12,  
    lambda x: 20*x + 40,  
]  
  
# Melakukan looping untuk membuat database dari beberapa soal integral  
data = []  
for i in range(len(functions)):  
    a, b = i + 2, i + 4 # Perubahan batas sesuai soal  
    integral = Trapezoid(a, b, functions[i])  
    data.append([a, b, integral])
```

```
[20] 2 , 4 , 12.0  
3 , 5 , 20.0  
4 , 6 , 28.0  
5 , 7 , 60.0  
6 , 8 , 100.0  
7 , 9 , 148.0  
8 , 10 , 204.0  
9 , 11 , 268.0  
10 , 12 , 332.0  
11 , 13 , 560.0
```

```
[21] import numpy as np  
import pandas as pd  
from sklearn import svm  
from google.colab import drive  
import matplotlib.pyplot as plt  
  
# Membaca data dari file  
drive.mount('/content/drive')  
file_path = '/content/drive/My Drive/trapezoid.txt' # Ganti path sesuai lokasi file Anda  
Database = pd.read_csv(file_path, sep=",", header=0)  
  
# X = Data, y = Target  
X = Database[['a', 'b']] # Pastikan nama kolom sesuai dengan file Anda  
y = Database['Target']  
  
# Membuat dan melatih model SVM  
clf = svm.SVC()  
clf.fit(X, y)
```

```
# Melakukan prediksi  
y_pred = clf.predict(X.values)  
  
# Menampilkan hasil prediksi  
print("Hasil prediksi:")  
for i, pred in enumerate(y_pred):  
    print(f"[X.iloc[i, 0]], [X.iloc[i, 1]] -> {pred}")  
  
# Membuat plot perbandingan nilai asli dengan nilai prediksi  
plt.figure(figsize=(10, 6))  
plt.plot(range(len(y)), y, 'o-', label='Nilai Asli (Target)', color='blue')  
plt.plot(range(len(y_pred)), y_pred, 'x--', label='Nilai Prediksi (SVM)', color='yellow')  
  
# Menambahkan label dan judul  
plt.xlabel('Indeks Data')  
plt.ylabel('Nilai')  
plt.title('Perbandingan Nilai Asli vs Nilai Prediksi Menggunakan SVM')  
plt.legend()  
plt.grid()  
plt.show()
```

```
[21] Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
Hasil prediksi:
1, 2 -> 2.0
2, 3 -> 4.0
3, 4 -> 6.0
4, 5 -> 8.0
5, 6 -> 10.0
6, 7 -> 12.0
7, 8 -> 14.0
8, 9 -> 16.0
9, 10 -> 18.0
10, 11 -> 20.0
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but SVC was fitted with feature names
warnings.warn(
```

