

# APLIKASI PENGHITUNG SKOR

Randa Andriana Putra<sup>1</sup>, Uliano Wilyam Purba<sup>2</sup>, Tria Yunanni<sup>3</sup>, Daris Samudra<sup>4</sup>, Nasywa Nur Afifah<sup>5</sup>

122450083<sup>1</sup>, 122450098<sup>2</sup>, 122450062<sup>3</sup>, 122450102<sup>4</sup>, 122450125<sup>5</sup>

Program Studi Sains Data Institut Teknologi Sumatera

Jl. Terusan Ryacudu, Way Huwi, Kec. Jatiagung, Kabupaten Lampung Selatan,

Lampung 35365

Email: [randa.122450083@student.itera.ac.id](mailto:randa.122450083@student.itera.ac.id)<sup>1</sup>, [uliano.122450098@student.itera.ac.id](mailto:uliano.122450098@student.itera.ac.id)<sup>2</sup>,  
[tria.122450062@student.itera.ac.id](mailto:tria.122450062@student.itera.ac.id)<sup>3</sup>, [daris.122450102@student.itera.ac.id](mailto:daris.122450102@student.itera.ac.id)<sup>4</sup>,  
[nasywa.122450125@student.itera.ac.id](mailto:nasywa.122450125@student.itera.ac.id)<sup>5</sup>

## PENDAHULUAN

### A. Latar Belakang

Saat ini, perhitungan skor atau poin sering kali merupakan suatu bagian penting dari suatu aktivitas atau permainan. Mulai dari pertandingan olahraga, video game, kompetisi, hingga evaluasi kinerja, penghitungan skor memiliki peran yang sangat penting untuk menentukan hasil akhir atau tingkat keberhasilan.

Akan tetapi, pada beberapa situasi, pencatatan skor masih dilakukan secara manual yang mana akan menghabiskan waktu dan tidak begitu efisien. Selain membutuhkan waktu yang lama, proses manual tersebut bisa meningkatkan resiko terjadinya kesalahan perhitungan sehingga mengakibatkan ketidakakuratan pada hasil akhirnya.

Dengan begitu, mengembangkan aplikasi penghitungan skor lebih efektif dan efisien adalah hal yang sangat penting. Dengan aplikasi seperti itu, kita dapat mengotomatiskan penghitungan skor, memberikan kepastian akurasi yang tinggi, dan mengurangi waktu serta usaha user.

### B. Tujuan

Berdasarkan latar belakang di atas, tujuan dari pembuatan aplikasi penghitung skor ini adalah sebagai berikut:

1. Menerapkan konsep pemrograman fungsional dengan pembuatan aplikasi penghitung skor.
2. Memudahkan user untuk menghitung skor secara otomatis dengan aplikasi penghitung skor.

## METODE

### A. Python

Python adalah bahasa pemrograman tingkat tinggi yang dibuat oleh Guido Van Rossum dan dirilis pada tahun 1991. Python adalah bahasa pemrograman berbasis objek yang interaktif. Bahasa ini mengandung struktur data yang sangat baik. Python adalah

bahasa pemrograman interpretatif dengan banyak fungsi yang terfokus pada kejelasan dan kemudahan pemahaman kode. Dianggap sebagai bahasa yang menggabungkan kemampuan dan sinkronisasi sintaks kode, Python memungkinkan programmer membuat program dengan efisiensi waktu, kemudahan pengembangan, dan kompatibilitas dengan sistem. Python dapat digunakan untuk membuat aplikasi atau pemrograman skrip mandiri

## **B. Fungsi Reduce**

Fungsi `reduce()` dalam Python adalah salah satu metode kelas `functools` bawaan yang mengulangi setiap item dalam daftar atau tipe data yang dapat diubah dan mengembalikan satu nilai. Fungsi `reduce()` juga dapat menggabungkan elemen iterable menjadi satu nilai tunggal dengan bantuan fungsi tertentu. Fungsi `reduce` memungkinkan pengguna menggabungkan elemen iterable menjadi satu nilai akhir. Ini mencakup penggunaan fungsi yang diterapkan pada elemen yang dapat diubah secara kumulatif.

Dengan dua parameter, fungsi "`reduce()`" berfungsi untuk menggabungkan dua elemen dan iterable yang akan dioperasikan. Prosesnya dimulai dengan mengambil dua elemen pertama dari iterable, menerapkan fungsi pada keduanya, dan menggantinya dengan hasil operasi. Prosedur ini berlanjut hingga semua elemen dalam iterable digabungkan menjadi satu.

## **C. Fungsi Lambda**

Fungsi Lambda di Python adalah salah satu alternatif untuk menulis definisi fungsi. Secara sintaksis, ini adalah satu kata dan memiliki definisi dengan kata kunci `lambda`. Banyak bahasa pemrograman mainstream seperti Python, Java, dan C++ saat ini menggunakan ekspresi `lambda`, fitur pemrograman fungsional. Beberapa penelitian tentang C++ dan Java meneliti bagaimana ekspresi `lambda` mempengaruhi programmer.

Fungsi `lambda` juga di sebut fungsi anonim atau tidak memiliki nama, yang dikenal sebagai "`anonym`", adalah fungsi kecil dan terbatas yang terdiri dari hanya satu baris disebut Fungsi Lambda, seperti fungsi normal, dapat memiliki banyak argumen dalam satu ekspresi. Ekspresi `lambda` (atau bentuk `lambda`) digunakan dalam Python untuk membuat fungsi anonim. untuk menggunakannya, yaitu menggunakan kata kunci `lambda`, seperti mendefinisikan fungsi normal dengan `def`. Setiap fungsi anonim yang dibuat dengan Python akan memiliki tiga komponen penting yakni:

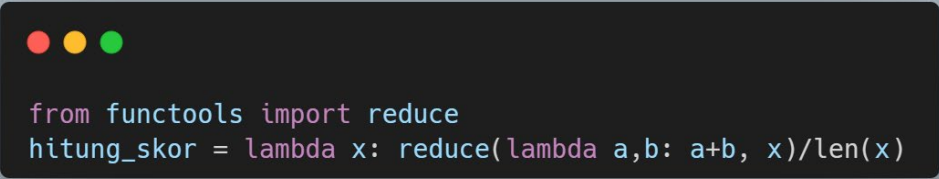
1. Kata kunci `lambda`,
2. Parameter (atau variabel terikat),
3. Badan fungsi

## **PEMBAHASAN**

Pada kasus ini, kami mengambil contoh untuk penerapan penghitungan skor dalam lingkungan akademis kampus. Yaitu untuk menentukan apakah seorang mahasiswa beserta predikat nilainya.

Dalam membuat program aplikasi penghitung skor kali ini, kami menggunakan fungsi `reduce` untuk menghitung total skor dari nilai-nilai yang ada pada list '`nilai`'. Untuk

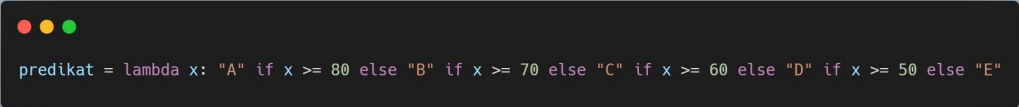
menggunakan fungsi reduce, diperlukan untuk memanggil fungsi reduce dengan cara: `from functools import reduce`. Fungsi reduce tersebut membutuhkan dua argumen, yaitu fungsi dan iterable. Yang mana untuk fungsinya adalah fungsi lambda yang mengambil dua argumen, yaitu `a` dan `b` lalu mengembalikan hasil penjumlahan keduanya dan untuk iterablenya adalah list nilai. Dengan menggunakan fungsi reduce, maka program dapat secara terurut untuk menghitung semua skor nilai dalam list nilai sehingga akhirnya akan mendapatkan total skor. Berikut adalah penggunaan fungsi reduce dalam program yang kami buat:



```
from functools import reduce
hitung_skor = lambda x: reduce(lambda a,b: a+b, x)/len(x)
```

*Gambar 1. Import Library dan Algoritma Penghitungan Skor*

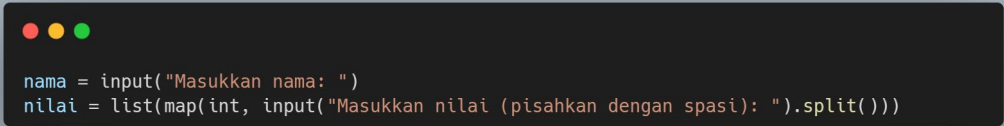
Selanjutnya program akan menentukan predikat mahasiswa berdasarkan nilai-nilai yang telah dihitung. Dalam langkah ini, program menggunakan fungsi lambda yang mengambil satu argumen yaitu `x` yang mana merupakan skor yang telah dihitung sebelumnya. Pada fungsi tersebut menggunakan konsep percabangan untuk menentukan predikat mahasiswanya. Berikut adalah codenya:



```
predikat = lambda x: "A" if x >= 80 else "B" if x >= 70 else "C" if x >= 60 else "D" if x >= 50 else "E"
```

*Gambar 2. Algoritma Penentuan Predikat*

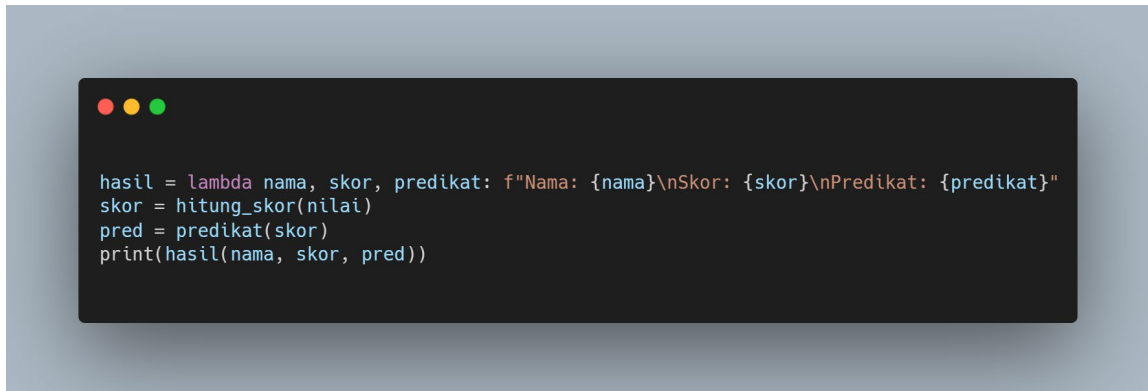
Lalu program akan meminta user yang mana pada kasus kali ini adalah dosen yang menginputkan nama beserta nilai dari mahasiswanya. Berikut adalah codenya:



```
nama = input("Masukkan nama: ")
nilai = list(map(int, input("Masukkan nilai (pisahkan dengan spasi): ").split()))
```

*Gambar 3. Algoritma Penginputan dari User*

Maka pada akhirnya program akan menampilkan outputnya kembali yaitu antara lain: nama mahasiswa, rata-rata nilai mahasiswa, dan predikat mahasiswanya. Berikut adalah codenya:



```
hasil = lambda nama, skor, predikat: f>Nama: {nama}\nSkor: {skor}\nPredikat: {predikat}"
skor = hitung_skor(nilai)
pred = predikat(skor)
print(hasil(nama, skor, pred))
```

*Gambar 4. Algoritma Penampilan Output*

## KESIMPULAN

Aplikasi penghitung skor khususnya untuk menghitung predikat mahasiswa dari nilai-nilainya dapat sangat berguna dalam lingkungan perkuliahan. Aplikasi tersebut sangat mudah digunakan oleh dosen, karena dosen hanya perlu mengetik nama, dan nilai-nilai mahasiswanya saja, tidak perlu melakukan perhitungan manual yang akan memakan waktu dan tenaga. Serta dengan adanya aplikasi tersebut dapat mengurangi kesalahan penghitungan yang jika dilakukan secara manual karena pada aplikasi tersebut menggunakan fungsi lambda dengan fungsi reduce yang akan menghasilkan perhitungan otomatis untuk nilai-nilai dan penentuan predikatnya.