

### 1. Program Hitung Umur

Tujuan : Menghitung tahun kelahiran berdasarkan umur saat ini.  
Input : Nama (string), Umur (integer).  
Logika : Tahun Lahir = Tahun Sekarang (2025) - Umur.  
Output : "Halo [Nama], kamu lahir pada tahun [Tahun Lahir]."

### 2. Konversi Mata Uang Sederhana

Tujuan : Mengkonversi Rupiah ke Dolar AS.  
Input : Jumlah Rupiah (integer).  
Logika : 1 Dolar = Rp 15.500 (asumsi). Hitung hasil konversi.  
Output : "Rp [Jumlah] setara dengan \$ [Hasil]."

### 3. Menghitung Luas Segitiga

Tujuan : Menghitung luas segitiga.  
Input : Alas (float), Tinggi (float).  
Logika :  $\text{Luas} = 0.5 * \text{alas} * \text{tinggi}$ .  
Output : "Luas segitiga adalah [Luas] cm<sup>2</sup>."

### 4. Kasir Toko Buku (Diskon Tetap)

Tujuan : Menghitung total harga pembelian buku.  
Input : Harga Buku (integer), Jumlah Buku (integer).  
Logika : Total = Harga \* Jumlah. Diskon = 5% dari Total. Bayar = Total - Diskon.  
Output : "Total bayar setelah diskon 5%: Rp [Bayar]."

### 5. Konversi Jarak (KM to Meter & CM)

Tujuan : Mengubah jarak kilometer ke meter dan centimeter.  
Input : Jarak dalam KM (float).  
Logika : Meter = KM \* 1000. CM = Meter \* 100.  
Output : Tampilkan hasil dalam meter dan centimeter.

### 6. Biodata Mahasiswa Formatted

Tujuan : Menampilkan biodata dengan format rapi menggunakan f-string. `print(f'nama: {nama}')`  
Input : Nama, NIM, Jurusan, Angkatan.  
Logika : Gabungkan input menjadi kalimat deskriptif.  
Output : "Mahasiswa bernama [Nama] ([NIM]) dari jurusan [Jurusan] angkatan [Angkatan]."

### 7. Penghitung Detik

Tujuan : Mengubah satuan jam menjadi total detik.  
Input : Jumlah Jam (integer).  
Logika : Total Detik = Jam \* 3600.  
Output : "[Jam] jam sama dengan [Total Detik] detik."

#### 8. Rata-rata 3 Nilai

Tujuan : Menghitung nilai rata-rata dari 3 mata pelajaran.  
Input : Nilai A, Nilai B, Nilai C (float).  
Logika :  $\text{Rata-rata} = (A + B + C) / 3$ .  
Output : "Nilai rata-rata kamu adalah [Hasil]."

#### 9. Ganjil atau Genap?

Tujuan : Menentukan apakah sebuah angka ganjil atau genap.  
Input : Angka (integer).  
Logika : Gunakan Modulus.  
Output : "[Angka] adalah bilangan [Genap/Ganjil]."

#### 10. Seleksi Masuk Bioskop

Tujuan : Mengecek apakah pengunjung boleh menonton film rating 17+.  
Input : Umur (integer).  
Logika : Jika Umur  $\geq 17$  tampilkan "Boleh Masuk", jika tidak tampilkan "Dilarang Masuk".  
Output : Pesan sesuai kondisi.

#### 11. Konversi Nilai Huruf

Tujuan : Mengubah nilai angka menjadi huruf mutu.  
Input : Nilai Angka (0-100).  
Logika : 85-100: A; 70-84: B; 60-69: C; < 60: D  
Output : "Nilai Anda: [Huruf]."

#### 12. Cek Tahun Kabisat

Tujuan : Menentukan apakah suatu tahun adalah kabisat.  
Input : Tahun (integer).  
Logika : Jika tahun habis dibagi 4 DAN (tidak habis dibagi 100 ATAU habis dibagi 400), maka Kabisat.  
Output : "[Tahun] adalah/bukan tahun kabisat."

#### 13. Login Sederhana (Hardcoded)

Tujuan : Validasi username dan password.  
Input : Username, Password.  
Logika : Username benar = "admin", Password benar = "12345". Jika keduanya cocok, "Login Sukses", jika tidak "Login Gagal".  
Output : Pesan sukses/gagal.

#### 14. Menentukan Wujud Air

Tujuan : Menentukan wujud air berdasarkan suhu.  
Input : Suhu (Celcius).  
Logika :  $\leq 0$ : Padat (Es);  $0 < \text{Suhu} < 100$ : Cair;  $\geq 100$ : Gas (Uap)  
Output : "Pada suhu ini air berwujud [Wujud]."

### 15. Kategori Usia

Tujuan : Mengelompokkan usia manusia.  
Input : Usia (integer).  
Logika : 0-5 (Balita); 6-12 (Anak); 13-17 (Remaja); 18-59 (Dewasa); 60+ (Lansia).  
Output : "Kategori usia: [Kategori]."

### 16. Validasi Segitiga

Tujuan : Mengecek apakah 3 sisi bisa membentuk segitiga.  
Input : Sisi A, Sisi B, Sisi C.  
Logika : Segitiga valid jika  $(A+B > C)$  DAN  $(A+C > B)$  DAN  $(B+C > A)$ .  
Output : "Bisa membentuk segitiga" atau "Tidak bisa membentuk segitiga".

### 17. Kalkulator Suhu (Fungsi)

Tujuan : Konversi suhu dengan pilihan menu menggunakan fungsi.  
Input : Buat menu 1. Celcius ke Fahrenheit dan 2. Fahrenheit ke Celcius. Kemudian input Suhu.  
Logika : Buat fungsi `celcius_to_fahrenheit()` dan `fahrenheit_to_celcius()`. Panggil sesuai pilihan.  
Output : Hasil konversi.

### 18. Fungsi Biaya Parkir

Tujuan : Menghitung biaya parkir dengan pembulatan waktu ke atas.  
Input : Jam Masuk, Jam Keluar (Format 24 jam, asumsi hari yang sama).  
Logika :  
1. Hitung durasi = Jam Keluar - Jam Masuk.  
2. Jika durasi  $\leq 2$  jam, biaya = Rp 3.000.  
3. Jika durasi  $> 2$  jam, hitung sisa jam. Sisa jam dihitung per jam penuh (contoh: 2 jam 1 menit dianggap masuk jam ke-3).  
4. Biaya =  $3.000 + (\text{Sisa Jam} * 1.000)$ . Output : "Total biaya parkir: Rp [Biaya]."

### 19. Fungsi Luas Lingkaran

Tujuan : Menghitung luas lingkaran menggunakan fungsi.  
Input : Jari-jari (float).  
Logika : Buat fungsi `luas_lingkaran(r)`. Rumus =  $3.14 * r * r$ .  
Output : "Luas lingkaran dengan jari-jari [r] adalah [Hasil]."

### 20. Fungsi Cek Kelulusan

Tujuan : Menentukan status kelulusan siswa berdasarkan nilai KKM.  
Input : Nilai Siswa (integer).  
Logika : Buat fungsi `cek_status(nilai)`. - Jika nilai  $\geq 75$  (75 pas termasuk lulus) return "Lulus". - Jika nilai  $< 75$  return "Remidial".  
Output : "Status Anda: [Hasil]."

### 21. Kalkulator Zakat Mal

Tujuan : Menghitung zakat harta.  
Input : Total Harta (integer).  
Logika : Buat fungsi `hitung_zakat(harta)`. Jika harta  $\geq 85$  juta, zakat = 2.5%. Jika tidak, 0.  
Output : "Zakat yang harus dibayar: Rp [Nominal]."

## 22. Konversi Waktu (Fungsi)

Tujuan : Mengubah detik menjadi format Jam:Menit:Detik.  
Input : Total detik (integer).  
Logika : Buat fungsi format\_waktu(detik).  
Output : "[Jam] jam, [Menit] menit, [Detik] detik."

## 23. Fungsi Diskon Bertingkat

Tujuan : Menghitung harga akhir berdasarkan member.  
Input : Status Member (Ya/Tidak), Total Belanja.  
Logika : Fungsi cek\_diskon(member, total). Member dapat 10%. Jika total > 1 juta, tambah diskon ekstra 5%.  
Output : Total bayar akhir.

## 24. Generator Username

Tujuan : Membuat username otomatis.  
Input : Nama Depan, Nama Belakang, Tahun Lahir.  
Logika : Fungsi create\_username(). Gabungkan NamaDepan + NamaBelakang + TerakhirTahun.  
Output : "Username Anda: [username]" (Contoh: AlfianGading2004).

## 25. Deret Bilangan Genap

Tujuan : Menampilkan bilangan genap dari 1 sampai N.  
Input : Angka Maksimal (N).  
Logika : Gunakan loop for atau while.  
Output : 2, 4, 6, ... N.

## 26. Penjumlahan Angka (Sum of N)

Tujuan : Menjumlahkan angka dari 1 sampai N.  
Input : Angka N.  
Logika : Loop dari 1 sampai N, tambahkan ke variabel total. (Misal input 5 -> 1+2+3+4+5).  
Output : "Total penjumlahan: [Total]."

## 27. Tabel Perkalian

Tujuan : Menampilkan tabel perkalian untuk angka tertentu.  
Input : Angka perkalian (misal 7).  
Logika : Loop 1 sampai 10. Print format i x Angka = Hasil.  
Output : 1 x 7 = 7; 2 x 7 = 14; 3 x 7 = 21 ... dst

## 28. Validasi Input (While Loop)

Tujuan : Meminta user memasukkan angka positif.  
Input : Angka (berulang).  
Logika : Gunakan while True. Jika user input angka < 0, minta lagi. Jika >= 0, berhenti.  
Output : "Terima kasih, Anda memasukkan angka positif."

### 29. Menghitung Faktorial

Tujuan : Menghitung N! (Faktorial).  
Input : Bilangan bulat N.  
Logika : Loop dari N menurun ke 1, kalikan semua angka.  
Output : "Faktorial dari [N] adalah [Hasil]."

### 30. Pola Segitiga Bintang Terbalik

Tujuan : Membuat pola bintang menurun.  
Input : Tinggi segitiga.  
Logika : Nested loop. Baris pertama bintang sebanyak N, baris kedua N-1, dst.  
Output : (Jika input 3)

\*\*\*

\*\*

\*

### 31. Tebak Angka Sederhana

Tujuan : User menebak angka rahasia (misal: 7, di hardcode saja dalam variable).  
Input : Tebakan user.  
Logika : Loop sampai tebak user == 7. Beri hint "Salah, coba lagi". Boleh batasi kesempatan (max 3x).  
Output : "Selamat, tebakkan benar!" atau "Game Over".

### 32. Deret Bilangan Kuadrat

Tujuan : Menampilkan deret angka hasil pemangkatan dua (kuadrat).  
Input : Jumlah bilangan (N).  
Logika : Loop dari 1 sampai N. Di setiap perulangan, hitung  $i * i$  (atau  $i ** 2$ ).  
Output : 1, 4, 9, 16, 25 ... (sesuai jumlah N).

### 33. Daftar Belanja Sederhana (List)

Tujuan : Mengelola daftar belanja.  
Input : Menu (1. Tambah, 2. Lihat, 3. Selesai).  
Logika : Gunakan list belanjaan = []. append untuk tambah. Loop for untuk lihat.  
Output : Daftar item yang sudah diinput.

### 34. Mencari Nilai Maksimal & Minimal (List)

Tujuan : Mencari nilai tertinggi dan terendah dari sekumpulan nilai.  
Input : User input 5 angka (masukkan ke list).  
Logika : Jangan pakai max()/min() bawaan python. Buat sendiri fungsinya. Loop list dan bandingkan.  
Output : "Nilai Tertinggi: [Max], Terendah: [Min]".

### 35. Kamus Sederhana (Dictionary)

Tujuan : Menerjemahkan kata Indonesia ke Inggris.  
Input : Kata bahasa Indonesia.  
Logika : Dict kamus = {"apel": "apple", "kucing": "cat", ...}. Cek apakah input ada di key.  
Output : Terjemahan atau "Kata tidak ditemukan".

### 36. Data Mahasiswa (List of Dictionary)

Tujuan : Menyimpan data nama dan nilai mahasiswa.

Input : Nama, Nilai (bisa input berulang).

Logika : List data\_mhs. Tiap elemen adalah dict {'nama': ..., 'nilai': ...}. Tanyakan kepada pengguna apakah ingin menambahkan input atau tidak. Jika ya, berikan ulang input nama dan nilai. Jika tidak tampilkan semua data

Output : Tampilkan semua data.

### 37. Hitung Vokal (String/List)

Tujuan : Menghitung jumlah huruf vokal dalam kalimat.

Input : Kalimat.

Logika : List vokal = ['a','i','u','e','o']. Loop setiap huruf di kalimat, jika ada di list vokal, counter bertambah.

Output : "Jumlah huruf vokal: [Jumlah]".

### 38. Sistem Absensi (List)

Tujuan : Mencatat nama unik kehadiran (menghindari duplikat).

Input : Nama (berulang).

Logika : Masukkan ke List. Jika nama sudah ada di list, tampilkan "Nama sudah absen". Jika belum, append. Ulang nama input sampai 5 kali. Lalu keluar berikan daftar nama yang hadir.

Output: Daftar nama yang hadir.

### 39. Mengurutkan Angka (Sorting List)

Tujuan : Mengurutkan input angka dari user.

Input : 5 angka acak.

Logika : Masukkan ke list. Gunakan metode lalu sorting dengan sort().

Output : "List terurut: [...]"

### 40. Inventaris Gudang (Dictionary manipulation)

Tujuan : Update stok barang.

Input : Buat menu (1. Tambah barang dan 2. Keluar). Jika memilih 1, Berikan input berikut Nama Barang, Jumlah Barang Masuk/Keluar

Logika : Dict gudang = {"Pensil": 10, "Buku": 5}. Jika barang ada, update valuenya.

Output : Stok akhir barang tersebut.

### 41. Cek Kata Sandi Lemah/Kuat

Tujuan : Validasi panjang password.

Input : Password.

Logika : Jika panjang karakter < 8, "Password Lemah". Jika >= 8, "Password Kuat".

Output : Status kekuatan password.

### 42. Format Tanggal Indonesia

Tujuan : Menampilkan tanggal dengan nama bulan.

Input : Tanggal, Bulan (angka 1-12), Tahun.

Logika : Gunakan List atau Dictionary untuk mapping angka bulan ke nama bulan (1 -> Januari).

Output : "17 Agustus 1945"

#### 43. Kalkulator BMI (Body Mass Index)

Tujuan : Menghitung BMI dan kategorinya.

Input : Berat (kg), Tinggi (cm).

Logika : Ubah tinggi ke meter. Rumus:  $\text{Berat} / (\text{Tinggi}^2)$ . Kategori:  $< 18.5$  Kurus,  $18.5-24.9$  Normal,  $\geq 25$  Gemuk.

Output : "BMI Anda: [Nilai], Kategori: [Kategori]"

#### 44. Konversi Detik ke Jam Menit Detik

Tujuan : Membalikkan konversi waktu.

Input : Total Detik (integer).

Logika :  $\text{Jam} = \text{Detik} // 3600$ . Sisa detik =  $\text{Detik} \% 3600$ . Menit =  $\text{Sisa} // 60$ . Detik Akhir =  $\text{Sisa} \% 60$ .

Output : "[Jam] jam, [Menit] menit, [Detik] detik"

#### 45. Lampu Lalu Lintas

Tujuan : Simulasi aksi berdasarkan warna lampu.

Input : Warna (Merah/Kuning/Hijau).

Logika : Merah -> "Berhenti", Kuning -> "Bersiap", Hijau -> "Jalan". Selain itu -> "Warna tidak valid".

Output : Aksi yang harus dilakukan.

#### 46. Menentukan Kuadran Koordinat

Tujuan : Menentukan posisi titik (x,y) dalam kartesius.

Input : Nilai x dan y.

Logika :  $x > 0, y > 0$  (Kuadran I);  $x < 0, y > 0$  (Kuadran II);  $x < 0, y < 0$  (Kuadran III);  $x > 0, y < 0$  (Kuadran IV).

Output : "Titik berada di [Kuadran]"

#### 47. Bonus Karyawan

Tujuan : Menghitung bonus akhir tahun dengan kondisi bertingkat.

Input : Lama Bekerja (tahun), Performa (Sangat Baik/Baik/Cukup).

Logika :

- Jika Lama  $> 5$  tahun DAN Performa "Sangat Baik" -> Bonus = 5x Gaji.

- Jika Lama  $> 5$  tahun DAN Performa "Baik" -> Bonus = 3x Gaji.

- Jika Lama  $\geq 2$  tahun (kondisi lainnya) -> Bonus = 1x Gaji.

- Selain itu (kurang dari 2 tahun atau performa buruk) -> Bonus = 0.

Output : "Bonus yang diterima: Rp [Nominal]"

#### 48. Tiket Kereta Api

Tujuan : Menghitung harga tiket berdasarkan kelas dan jarak tujuan.

Input : Kelas (Ekonomi/Bisnis/Eksekutif), Tujuan (Bandung/Surabaya), Jumlah Tiket.

Logika :

1. Tentukan Harga Dasar: Ekonomi = 100rb, Bisnis = 250rb, Eksekutif = 500rb.

2. Cek Tujuan: Jika Tujuan "Surabaya", tambah biaya ekstra 50rb per tiket (karena jauh). Jika "Bandung", tidak ada tambahan.

3. Total = (Harga Dasar + Biaya Ekstra) \* Jumlah.

Output : "Total harga tiket ke [Tujuan]: Rp [Total]"

#### 49. Cek Kelipatan

Tujuan : Mengecek apakah angka A adalah kelipatan angka B.

Input : Angka A, Angka B.

Logika : Jika  $A \% B == 0$ , maka kelipatan.

Output : "Benar, [A] adalah kelipatan [B]"

#### 50. Menghitung Jumlah Digit

Tujuan : Menghitung berapa digit dalam sebuah angka (mengabaikan tanda minus).

Input : Angka (integer, bisa negatif).

Logika :

1. Ambil nilai absolut dari angka (hilangkan tanda negatif).

2. Ubah ke string lalu hitung panjangnya dengan len().

3. Contoh: Input -12345 -> Absolut 12345 -> Jumlah digit 5.

Output : "Angka [Input Awal] memiliki [Jumlah] digit."

#### 51. Mencari Angka Terbesar (Tanpa Max)

Tujuan : Mencari angka terbesar dari inputan user.

Input : Jumlah angka N, lalu input angka sebanyak N kali.

Logika : Simpan input pertama sebagai max. Loop sisa input, jika input\_baru > max, update max.

Output : "Angka terbesar adalah: [Nilai]"

#### 52. Menampilkan Huruf per Baris

Tujuan : Mengeja kata secara vertikal.

Input : Kata.

Logika : for huruf in kata: print(huruf).

Output :

A

P

e

l

#### 53. Hitung Mundur Roket

Tujuan : Simulasi hitung mundur peluncuran.

Input : Angka mulai (misal 10).

Logika : Loop range(10, 0, -1). Print angka.

Output : 10, 9, 8 ... 1, "Meluncur!"

#### 54. Teka-Teki Perkalian

Tujuan : User harus menjawab soal perkalian dengan benar agar program berhenti.

Input : Jawaban User.

Logika : Soal statis (misal 7 x 8). While jawaban != 56, minta input terus.

Output : "Salah, coba lagi." -> "Benar!"



#### 55. Filter Angka Genap (List + Loop)

Tujuan : Memisahkan angka genap dari list.

Input : List angka [1, 2, 3, 4, 5, 6].

Logika : Loop list, if angka % 2 == 0, masukkan ke list baru.

Output : List baru: [2, 4, 6]

#### 56. Kamus Sinonim (Update Dictionary)

Tujuan : Menambah kosakata baru ke kamus.

Input : Kata, Sinonim.

Logika : Cek apakah kata sudah ada. Jika belum, dict[kata] = sinonim.

Output : Tampilkan isi kamus terbaru.

#### 57. Total Harga Belanja (List of Dict)

Tujuan : Menghitung total keranjang belanja.

Input : List barang [{'nama': 'Roti', 'harga': 5000}, {'nama': 'Susu', 'harga': 10000}].

Logika : Loop list, sum harga.

Output : "Total Belanja: Rp 15000"

#### 58. Agenda Harian

Tujuan : Menampilkan jadwal berdasarkan hari.

Input : Hari (Senin/Selasa/...).

Logika : Dictionary jadwal = {'Senin': 'Matematika', 'Selasa': 'Fisika'}. Akses by key.

Output : "Jadwal hari [Hari] adalah [Mapel]"

#### 59. Konversi Nilai ke Bobot (List Comprehension - Opsional)

Tujuan : Mengubah list nilai angka menjadi list bobot (A/B/C) sekaligus.

Input : List Nilai [80, 60, 90].

Logika : Loop list, konversi tiap item, simpan di list baru.

Output : ["A", "C", "A"]

#### 60. Cek Ketersediaan Buku (Perpustakaan)

Tujuan : Mengecek status buku.

Input : Judul Buku.

Logika : Dictionary buku = {'Harry Potter': 'Tersedia', 'Laskar Pelangi': 'Dipinjam'}. Cek key.

Output : "Buku [Judul] statusnya: [Status]"

#### 61. Kalkulator Parkir Progresif

Tujuan : Menghitung tarif dengan aturan bertingkat dan batas maksimal (Max Cap).

Input : Lama Parkir (jam - integer).

Logika :

1. 1 jam pertama = Rp 3.000.

2. Jam berikutnya = Rp 1.000/jam.

3. Hitung Total Sementara.

4. Cek Batas Maksimal: Jika Total Sementara > 10.000, maka Total Akhir = 10.000. Jika tidak, gunakan Total Sementara.

Output : "Biaya Parkir: Rp [Total]"

#### 62. Generator Email Sekolah

Tujuan : Membuat email dari Nama dan NIM.

Input : Nama Depan, NIM.

Logika : Gabungkan nama.nim@sekolah.ac.id. Lowercase semua.

Output : alfian.123@sekolah.ac.id

#### 63. Menghitung Umur dalam Hari

Tujuan : Mengestimasi total hari hidup.

Input : Usia (tahun).

Logika : Usia \* 365 (abaikan kabisat untuk penyederhanaan).

Output : "Anda telah hidup selama +/- [Hari] hari."