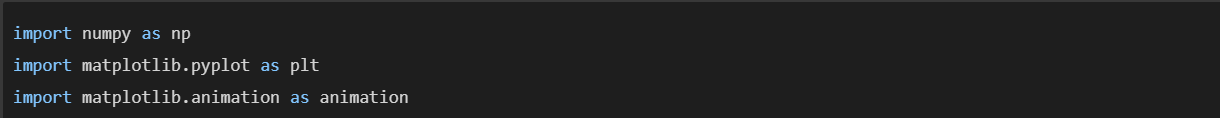
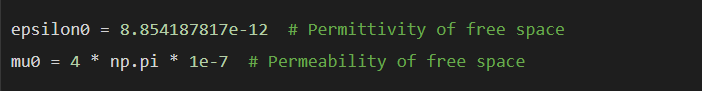
Project ini menggunakan aplikasi Google Collab yang berbasis web untuk menjalankan program yang akan kami buat. Dimana pada langkah awal mengimport pustaka NumPy yang digunakan untuk operasi numerik dan pustaka Matplotlib yang digunakan untuk membuat plot grafik. Selanjutnya, pustaka Matplotlib.animation diimpor untuk membuat animasi. Pada bagian berikutnya, animasi akan diatur menggunakan objek Figure dari Matplotlib dan diinisialisasi menggunakan objek Axis untuk menentukan batas dan skala sumbu grafik. Kemudian, fungsi update akan didefinisikan untuk mengupdate data pada setiap frame animasi. Selanjutnya, objek FuncAnimation akan dibuat dengan menggabungkan animasi dan fungsi update. Terakhir, fungsi plt.show() digunakan untuk menampilkan animasi secara interaktif..



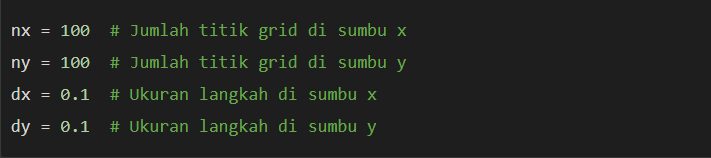
Gambar 1. Pengimportan ………..

Langkah kedua adalah dibuatlah fungsi persamaan Maxwell untuk mengitung pergerakan partikeltersebut. Dalam langkah ini dilakukannya pendefisian Variabel epsilon0 dan mu0 mewakili permittivity ruang hampa dan permeabilitas ruang hampa, masing-masing.GAMBAR 2.



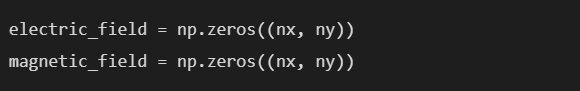
Gambar 2. Pendefinisian Fungsi persamaan Maxwell

Langkah ketiga adalahMenentukan dimensi grid dan ukuran langkah dimana Variabel nx dan ny menentukan jumlah titik grid di sumbu x dan y, sedangkan dx dan dy adalah ukuran langkah di masing-masing sumbu.. GAMBAR 3



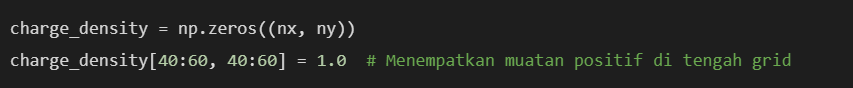
Gambar 3. Perhitungan turunan medan listrik dan medan magnet

Langkah keempat adalah Menginisialisasi grid untuk medan listrik dan medan magnetik Dengan Membuat matriks nol berukuran (nx, ny) untuk merepresentasikan medan listrik dan medan magnetik pada setiap titik grid. GAMBAR 4.



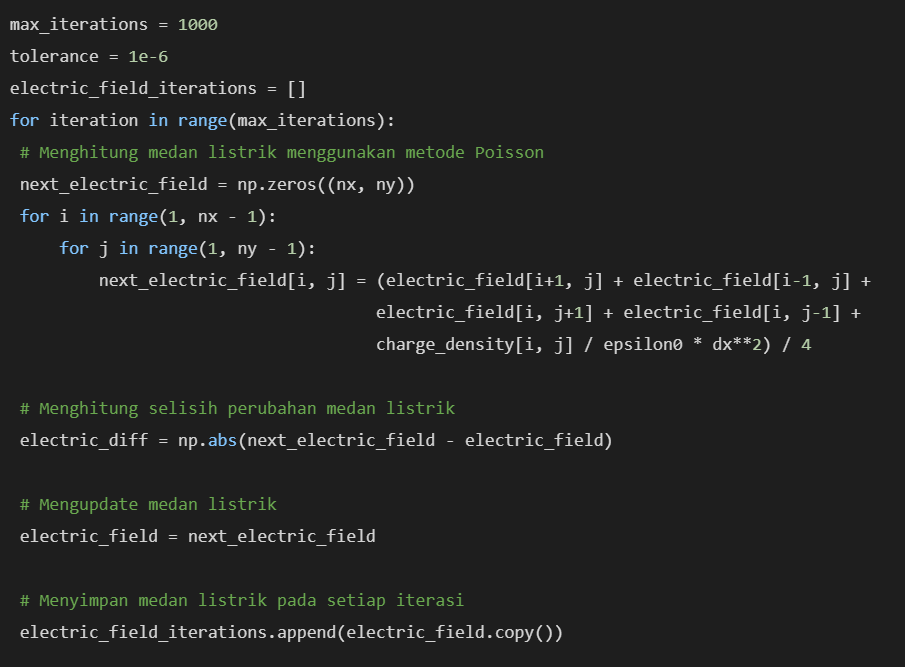
Gambar 4. Menginisialisasi grid medan listrik dan medan magnet

Langkah kelima adalah Menginisialisasi distribusi muatan. Dengan Membuat matriks nol charge\_density untuk merepresentasikan distribusi muatan dalam grid. Pada contoh ini, sebuah kotak dengan muatan positif ditempatkan di tengah grid., GAMBAR 5.



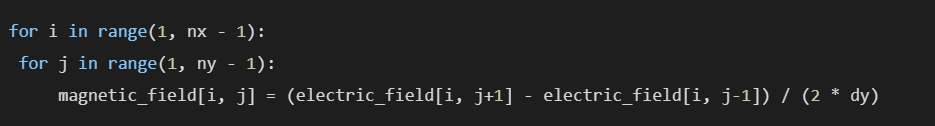
Gambar 5. Menginisialisasi distribusi muatan

Langkah keenam adalah Mengiterasi hingga konvergensi, pada setiap iterasi, medan listrik diperbarui menggunakan metode Poisson. Selisih perubahan medan listrik dihitung untuk memeriksa konvergensi. Jika selisih perubahan sudah lebih kecil dari batas toleransi (tolerance), iterasi dihentikan. GAMBAR 6.



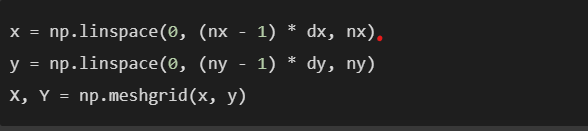
Gambar 6. Mengiterasi hingga konvergensi

Langkah ketujuh adalah Menghitung medan magnetik menggunakan persamaan Maxwell, medan magnetik dihitung menggunakan persamaan Maxwell, dengan mengambil perbedaan medan listrik pada sumbu y. GAMBAR 7.



Gambar 7. Menghitung medan magnet

Langkah kedelapan adalah Membuat grid untuk plotting. Variabel x dan y merepresentasikan sumbu x dan y pada grid. X dan Y adalah matriks 2D yang berisi nilai koordinat dari setiap titik grid. GAMBAR 8.



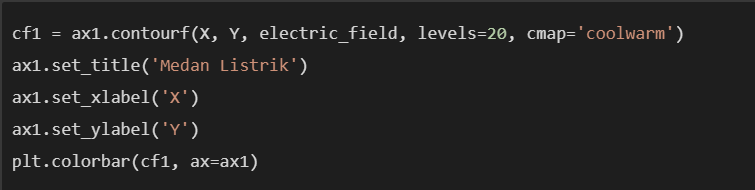
Gambar 8. Membuat grid untuk plotting

Langkah kesembilan adalah inisialisasi plot, Membuat objek figure dan dua subplot dengan ukuran 1 baris dan 2 kolom. fig merepresentasikan keseluruhan gambar, sedangkan ax1 dan ax2 merepresentasikan subplot pertama dan kedua. GAMBAR 9.



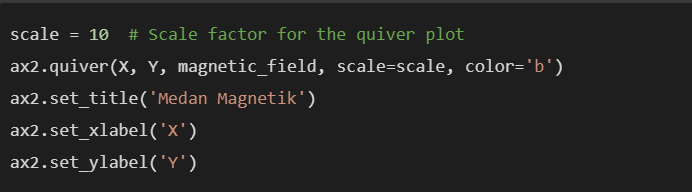
Gambar 9. inisialisasi plot

Langkah kesepuluh adalah Plot medan listrik. Pada langkah ini dilakukan Menggunakan ax1.contourf() untuk membuat plot kontur medan listrik. Parameter levels menentukan jumlah level pada plot, sedangkan cmap='coolwarm' digunakan untuk memilih colormap. Dilakukan juga konfigurasi tambahan seperti judul plot, label sumbu x dan y, serta penambahan colorbar., GAMBAR 9.

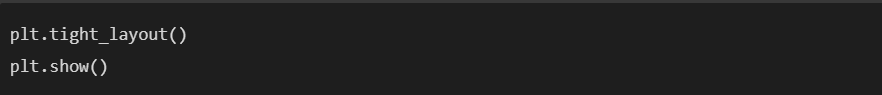


Gambar 9. plot medan listrik

Langkah kesebelas adalah Plot medan magnetik, Menggunakan ax2.quiver() untuk membuat plot medan magnetik menggunakan vektor. scale digunakan untuk mengatur skala vektor pada plot. Dilakukan juga konfigurasi tambahan seperti judul plot, label sumbu x dan y.GAMBAR 10



Gambar 10. Plot medan magnet

Langkah terakhir adalah Menampilkan plot, Dilakukan pengaturan tata letak (tight\_layout()) dan menampilkan gambar menggunakan plt.show(). GAMBAR12

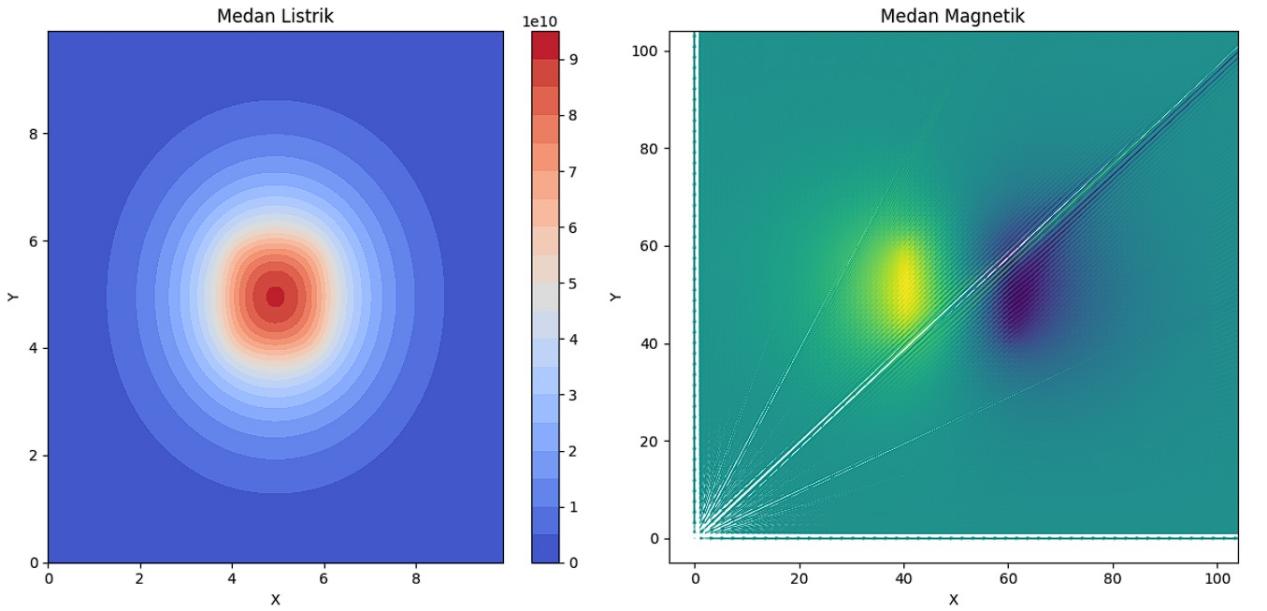
Gambar 8. Menampilkan plot

plt.tight\_layout() digunakan untuk memperbaiki tata letak plot dengan mengoptimalkan jarak antara elemen-elemen plot. Hal ini membantu menghindari tumpang tindih antara elemen-elemen plot dan memastikan tata letak yang lebih menyenangkan secara visual.

plt.show() digunakan untuk menampilkan jendela plot, sehingga hasil plot dapat dilihat untuk eksplorasi interaktif atau analisis lebih lanjut. Dengan menggunakan plt.show(), visualisasi plot yang dihasilkan akan terlihat dengan jelas dan tidak berantakan..

# HASIL

Visual yang dihasilkan dari *running* kode berupa grafik berikut :



Gambar 12. Grafik medan listrik dan medan magnet.

Output dari kode di atas adalah dua plot yang menunjukkan medan listrik dan medan magnetik yang dihasilkan oleh distribusi muatan positif di tengah grid.

Plot pertama menampilkan medan listrik sebagai kontur warna dengan menggunakan fungsi contourf. Warna kontur menggambarkan nilai medan listrik pada setiap titik grid. Semakin tinggi nilai medan listrik, semakin "merah" warna kontur, dan semakin rendah nilai medan listrik, semakin "biru" warna kontur. Nilai medan listrik dihitung menggunakan metode Poisson dan diupdate iteratif hingga konvergensi. Setiap iterasi medan listrik disimpan dalam variabel electric\_field\_iterations untuk tujuan visualisasi.

Plot kedua menampilkan medan magnetik menggunakan panah (quiver plot) dengan menggunakan fungsi quiver. Panah menunjukkan arah dan magnitudo medan magnetik pada setiap titik grid. Arah medan magnetik ditunjukkan oleh arah panah, sedangkan magnitudo medan magnetik ditunjukkan oleh panjang panah. Nilai medan magnetik dihitung menggunakan persamaan Maxwell.

Kedua plot tersebut disajikan dalam satu gambar dengan menggunakan fungsi subplots dari matplotlib. Setiap plot ditempatkan dalam sumbu (ax1 dan ax2) yang sesuai. Judul, label sumbu, dan skala pada plot juga ditentukan.