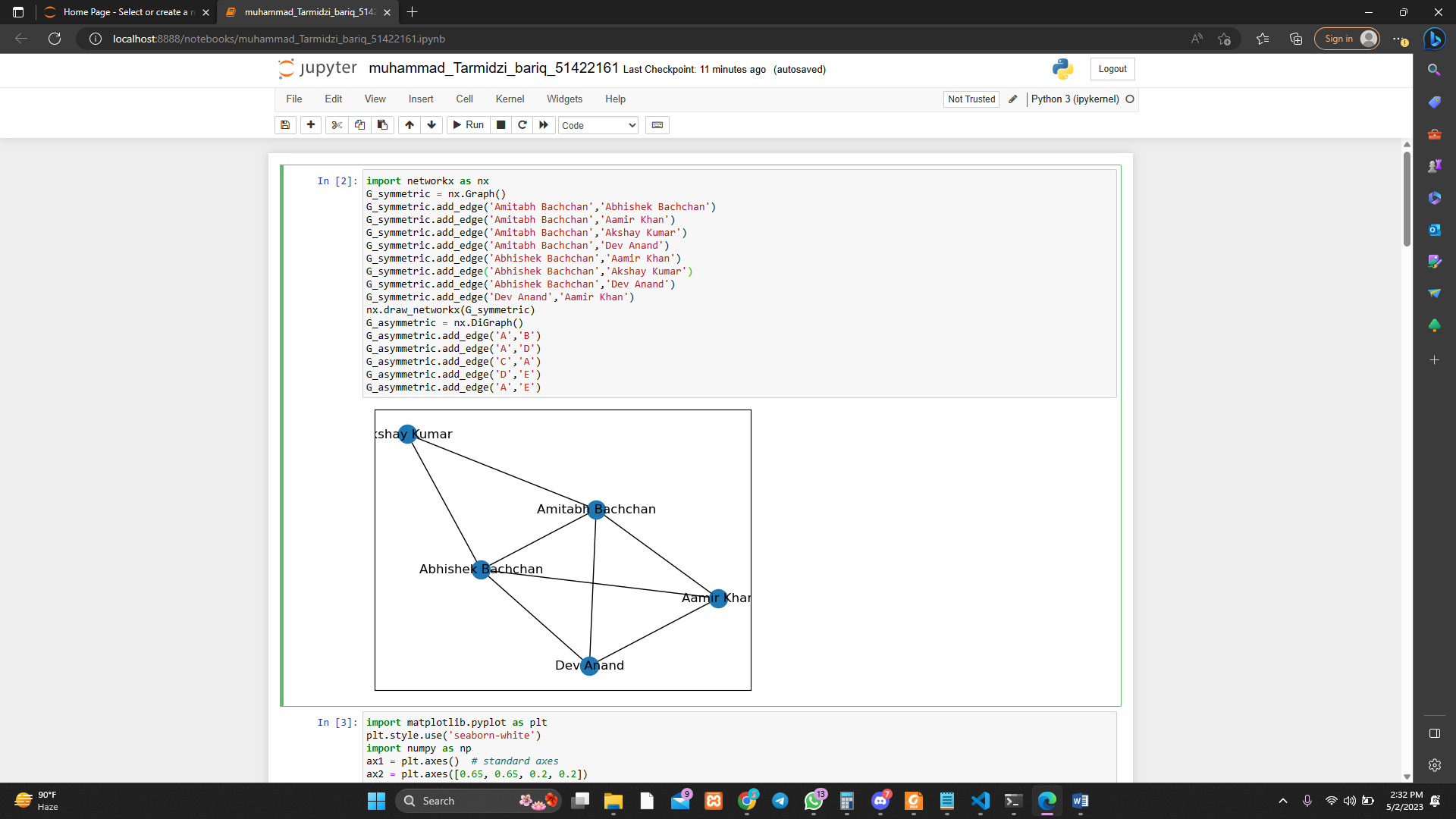
TUGAS PRAKTIKUM PERTEMUAN 6

NAMA : MUHAMMAD TARMIDZI BARIQ  
KELAS : 1IA13  
NPM : 51422161

1



Pada kode pemrograman tersebut, pertama-tama kita mengimport library NetworkX untuk mengakses algoritma dan fungsi yang berkaitan dengan graf dan jaringan.

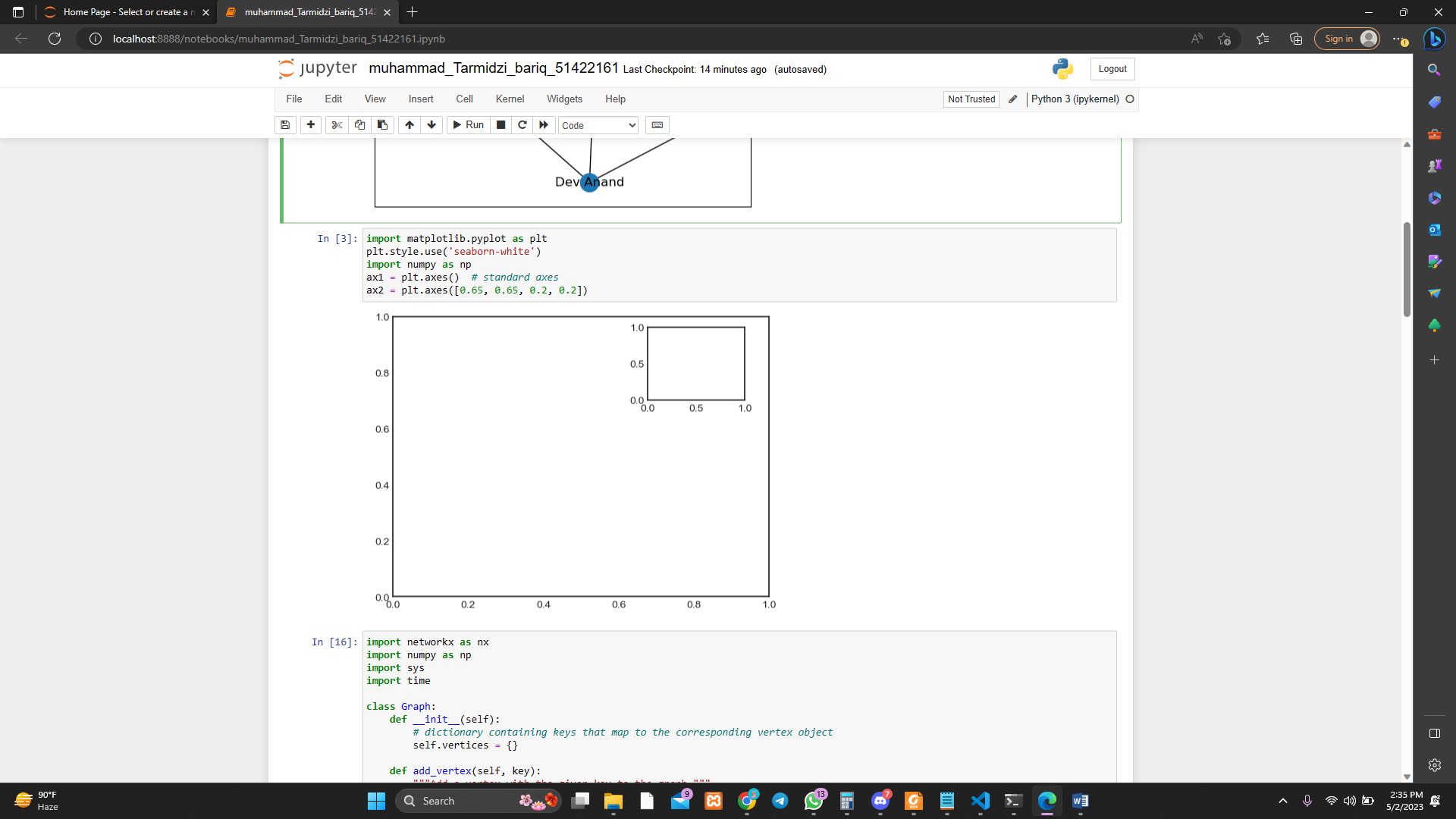
Kemudian kita membuat dua graf yang berbeda: G\_symmetric dan G\_asymmetric. Graf G\_symmetric merupakan graf tak-berarah (undirected) yang berisi beberapa simpul (node) yang merepresentasikan nama-nama aktor Bollywood dan beberapa sisi (edge) yang merepresentasikan hubungan antara aktor-aktor tersebut. Graf ini didefinisikan dengan menggunakan fungsi nx.Graph() dan setiap sisi ditambahkan menggunakan fungsi add\_edge().

Selanjutnya, graf G\_symmetric digambarkan dengan menggunakan fungsi nx.draw\_networkx().

Sedangkan, graf G\_asymmetric merupakan graf berarah (directed) yang juga berisi beberapa simpul dan sisi yang merepresentasikan hubungan antar simpul. Graf ini didefinisikan dengan menggunakan fungsi nx.DiGraph() dan setiap sisi ditambahkan menggunakan fungsi add\_edge().

Dalam contoh ini, simpul (node) direpresentasikan dengan karakter huruf dan sisi (edge) dihubungkan antara dua simpul dengan menggunakan fungsi add\_edge(). Pada graf ini terdapat simpul A, B, C, D, dan E, dan hubungan antar simpul direpresentasikan oleh sisi-sisi yang menghubungkan simpul-simpul tersebut.

2

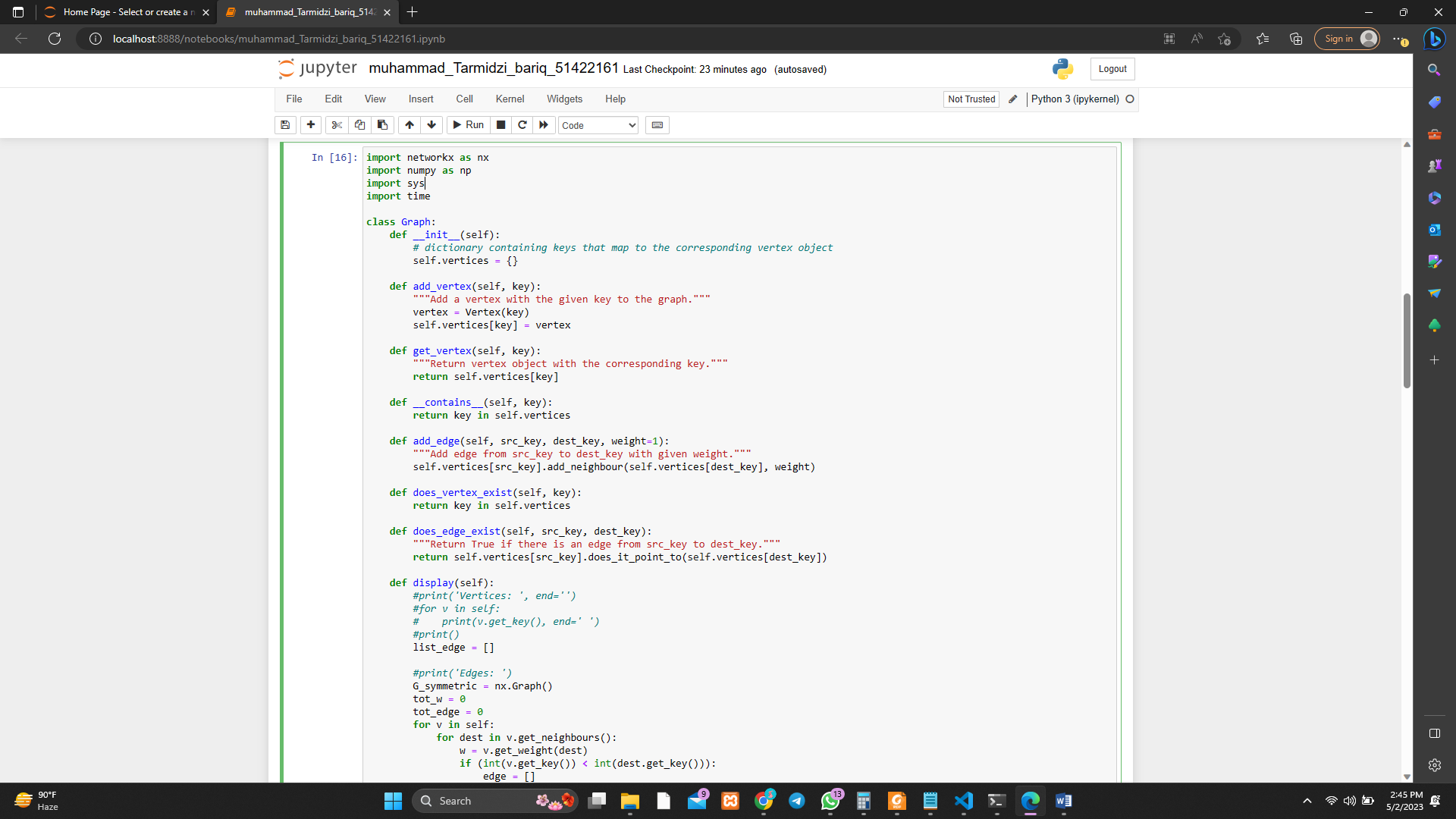


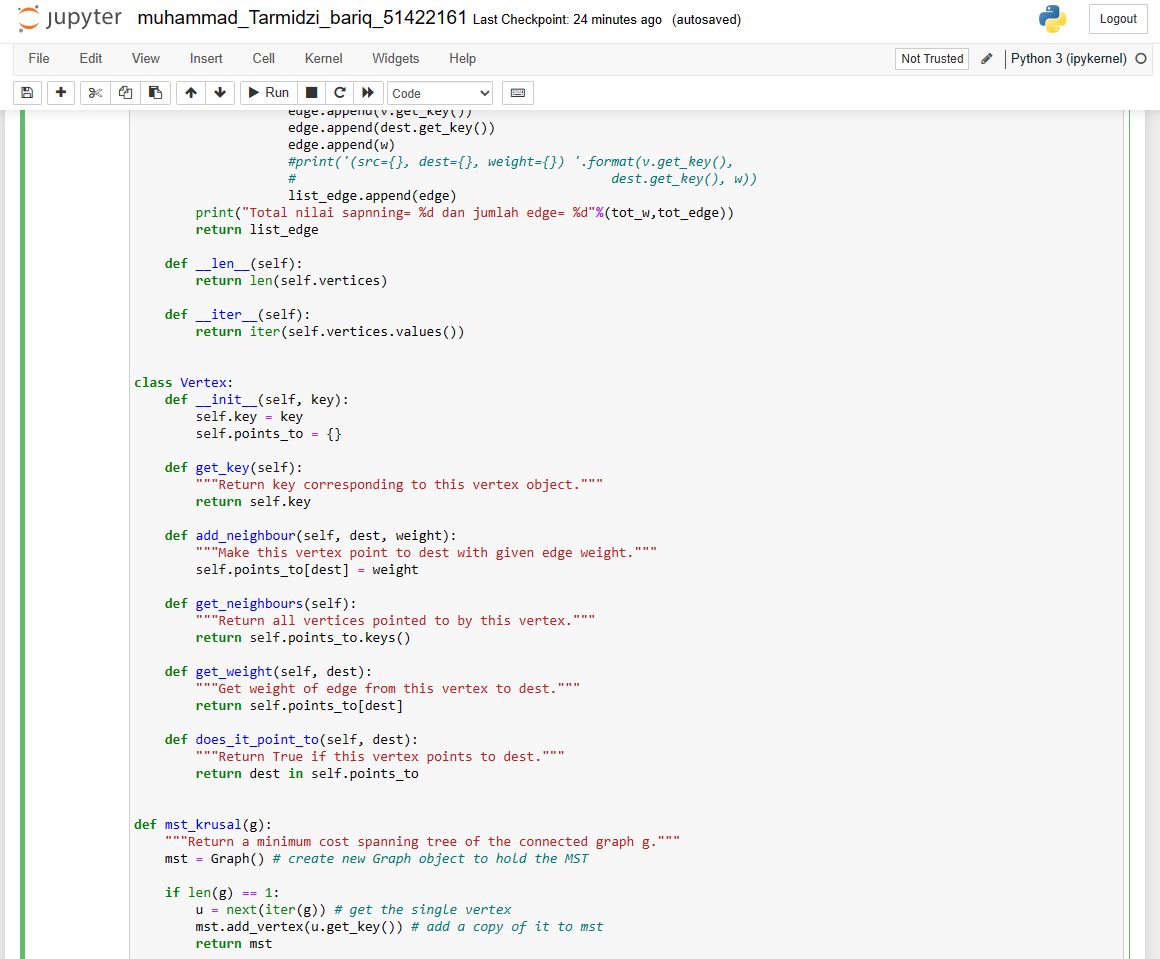
Pada kode pemrograman tersebut, pertama-tama kita mengimport library Matplotlib untuk mengakses algoritma dan fungsi yang berkaitan dengan visualisasi data.

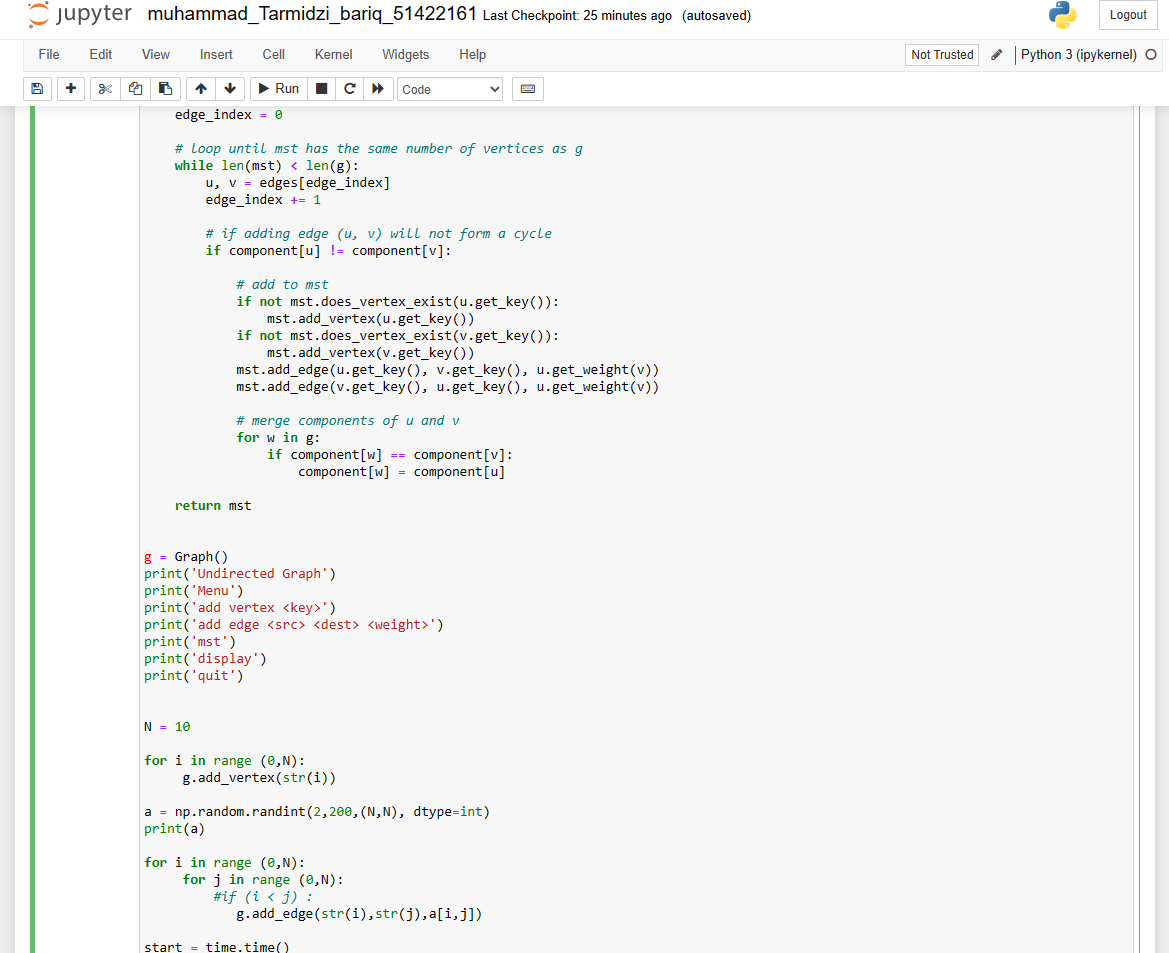
Kemudian, kita mengatur gaya plot yang akan digunakan dengan menggunakan fungsi plt.style.use(). Pada contoh ini, gaya plot yang dipilih adalah 'seaborn-white'.

Selanjutnya, kita membuat dua buah objek axes untuk menampilkan dua plot yang berbeda. Objek pertama dibuat dengan menggunakan fungsi plt.axes() tanpa argumen, sehingga objek tersebut akan berukuran standar. Objek kedua dibuat dengan menggunakan fungsi plt.axes() dengan argumen berupa daftar [0.65, 0.65, 0.2, 0.2], yang menentukan posisi dan ukuran objek tersebut dalam koordinat gambar (0.65, 0.65) dan ukuran (0.2, 0.2).

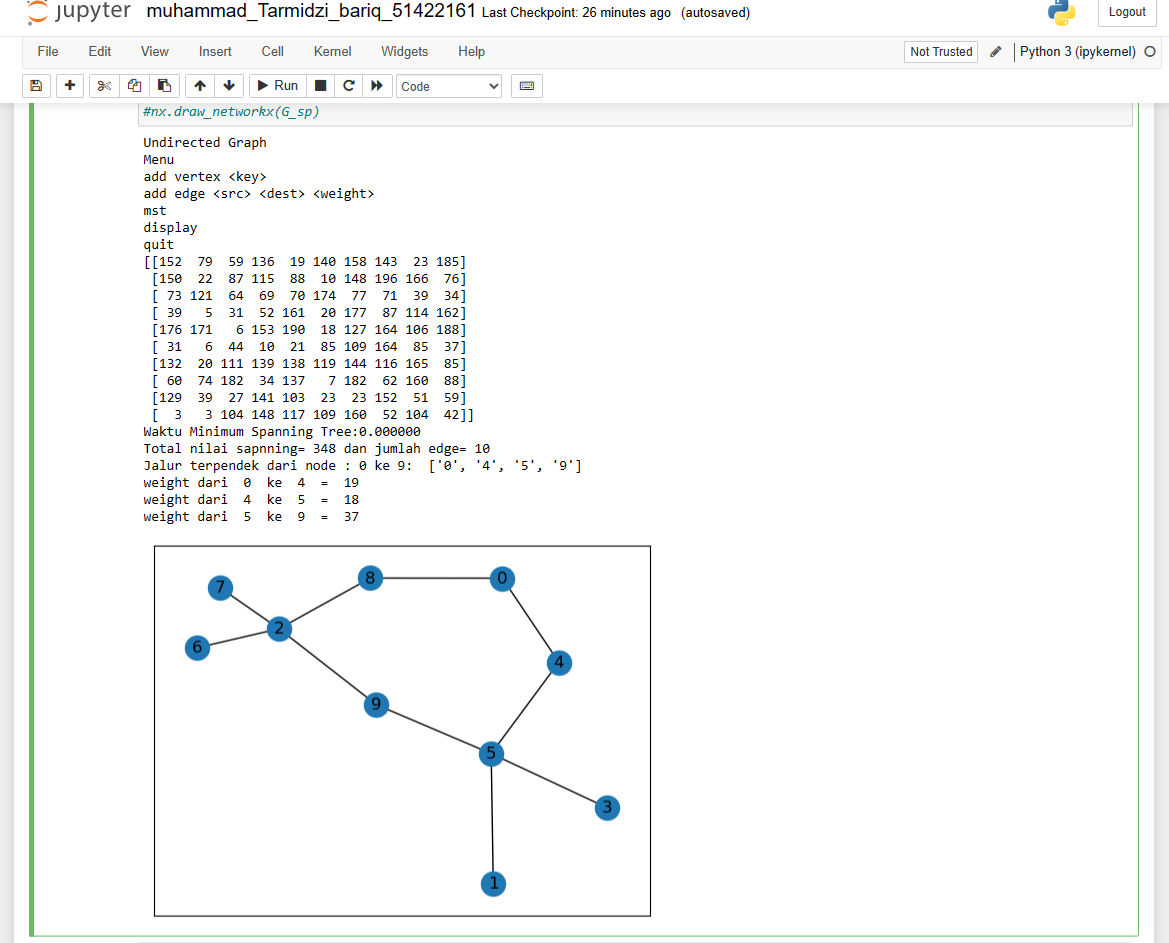
3.











Kode tersebut merupakan implementasi dari algoritma Kruskal untuk menemukan pohon rentang minimum dari graf tak berarah dan terhubung. Grafik direpresentasikan menggunakan daftar adjacency, dan kode menggunakan paket networkx untuk menampilkan grafik dan pohon rentang minimum. Kelas Grafik mewakili grafik, dan kelas Vertex mewakili sebuah simpul dalam grafik. Kelas Grafik berisi kamus yang memetakan kunci ke objek Vertex, dan menyediakan metode untuk menambahkan simpul dan tepi ke grafik, memeriksa apakah ada simpul atau tepi, dan iterasi di atas simpul. Fungsi mst\_krusal mengambil objek Grafik sebagai input dan mengembalikan objek Grafik baru yang mewakili pohon rentang minimum dari grafik input. Fungsi pertama-tama membuat objek grafik kosong untuk menampung pohon rentang minimum. Jika graf masukan hanya berisi satu simpul, fungsi menambahkan salinan simpul ke pohon rentang minimum dan mengembalikannya. Jika tidak, fungsi membuat daftar sisi dalam graf masukan, mengurutkan sisi berdasarkan bobot, dan menginisialisasi setiap simpul untuk berada dalam komponen terhubungnya sendiri. Fungsi tersebut kemudian mengulang tepi, menambahkan setiap tepi ke pohon rentang minimum jika tidak membentuk siklus dan menggabungkan komponen yang terhubung dari titik akhir tepi. Metode tampilan kelas Grafik mencetak simpul dan tepi grafik dan mengembalikan daftar tepi yang dapat digunakan untuk membuat grafik networkx untuk ditampilkan. Fungsi nx.Graph digunakan untuk membuat grafik tak terarah baru, dan tepi dalam daftar ditambahkan ke grafik menggunakan metode add\_weighted\_edges\_from. Terakhir, fungsi draw\_networkx digunakan untuk menampilkan grafik.