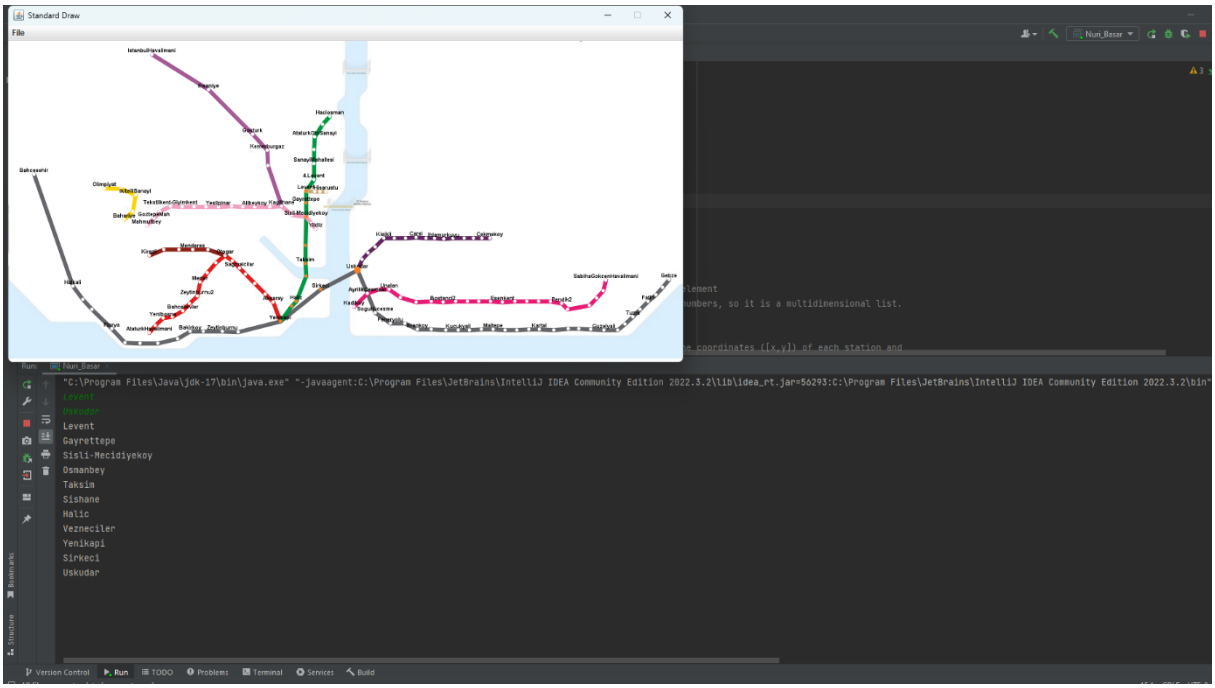
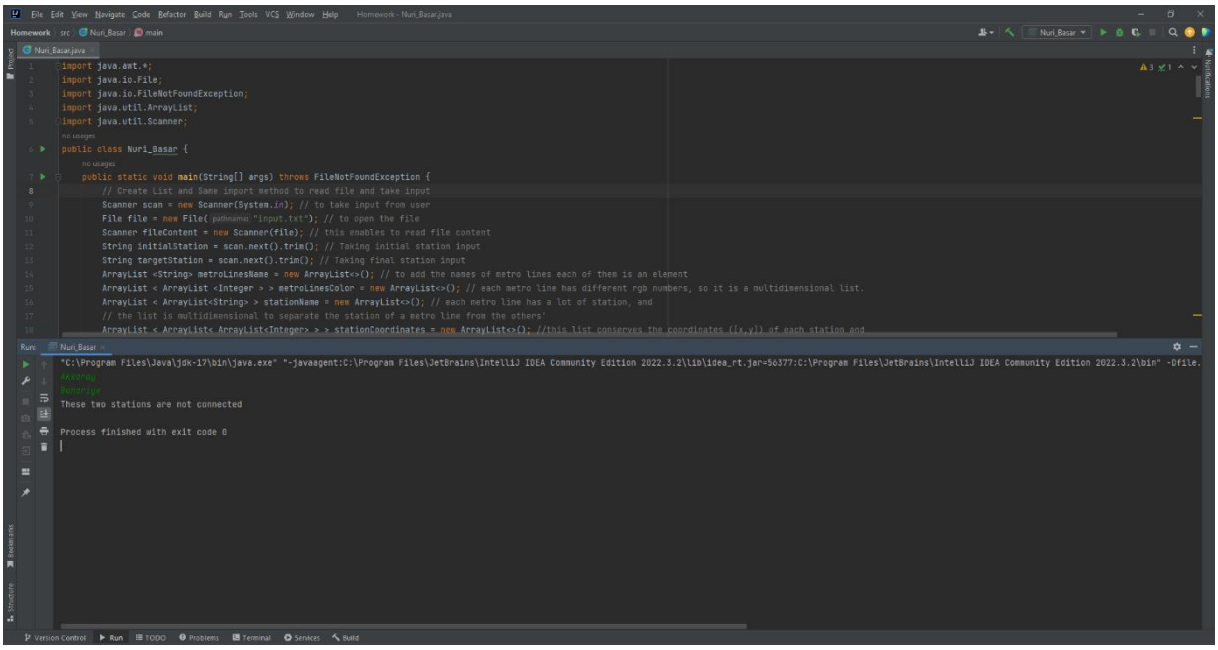
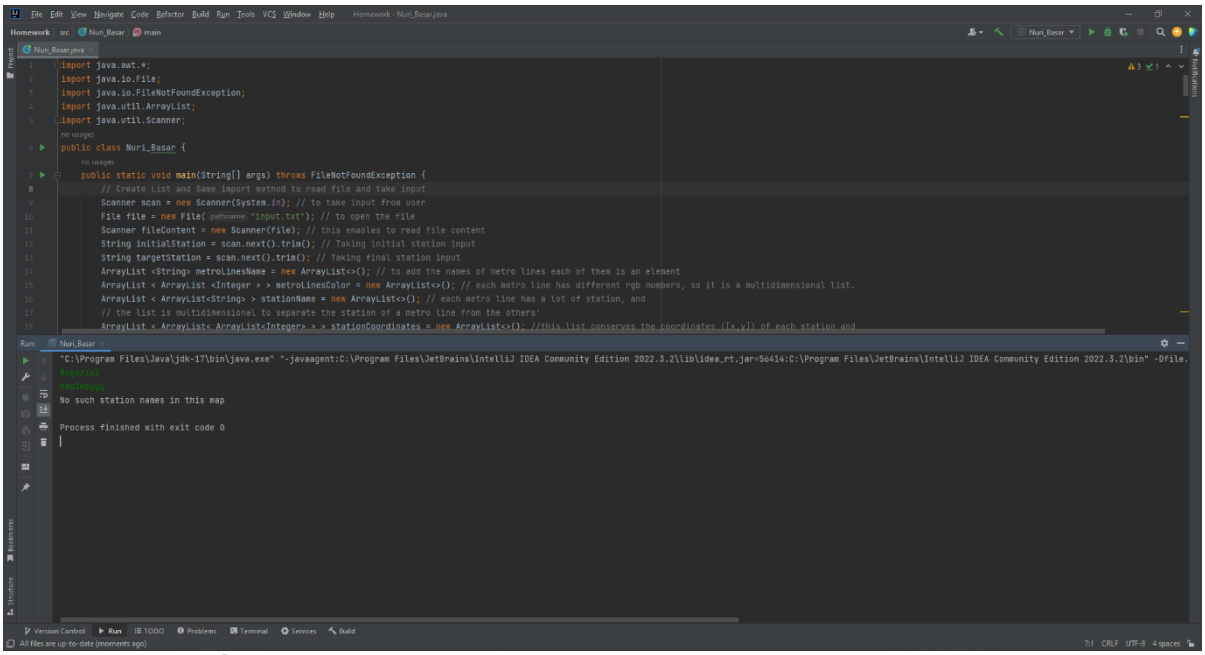
In the beginning, I open the “input” file, and I also take the initial station and destination point as input. After that, I create some arrays to hold information about metro lines and breakpoints. Next, I take all information from the file and separate all information about their related information. I also differentiate the stations I need to write on the background from the others and add them to another array with their name and their coordinates. Next, I try to find the metro lines which include the initial point and destination point as stations. If the code cannot find the stations, this means there are no such stations, and the name of the metro line will be null which means nothing. After this part, I need to control the metro lines from the previous part, and if there are no such stations, I need to print out "No such station names in this map” and finish the code. The Next part is the main algorithm which is in a method named “findMetroLineWayWithRecursion”. The method works recursively. The method always controls the incoming metro lines the current or initial metro line if the metro line is correct or not by returning metro lines in an array respectively. At the beginning of the method, there is a break statement that controls if the current metro line is the destination metro line or not (The way will explain incoming statements). In the second part, I need to find the metro lines which I can change the current metro at a breakpoint in the current metro line. I can do this by calling the “findingAvailableMetroLine” method which finds such metro lines by looking for the current metro line among the metro line array of breakpoints. However, the method is careful about not repeating itself. For example, if I Change M2 to B1, the method doesn’t take M1A as a point even B1 can have M2 and M1A as available metro lines because if I want to pass M1A, I can pass it from M2. Anyways, my main method tries all available metro lines after the current metro line recursively until it finds. For example, my initial metro line is M2, and I want to arrive at M5. First of all, it finds the available metro lines of M2 which are B1 and M1A. It tries both of them separately, and first changes the current metro line(M2) to M1A and finds its available metro lines which are M1B and again changes the current metro line to M1B. look for the same things, but now there is no available metro line. Therefore, it adds “Wrong” to the array containing [M2, M1A, M1B]. After that, it comes back to the available metro lines of M2, and now it changes the current metro line (M2) to B1 and looks at its available metro lines which are M5 and M4. The same as M1A, it tries both. If it’s M5, it is correct and adds it to the outside array, but if it’s M4, it does the same things as M1B and returns. After the method finds the metro line path, I look for which side I need to move to find the station by controlling the index of breakpoints at which I will change the metro line if it has a higher index than the current station (move to right) or it has a lower index than the current station (move to left). Next, I add the coordinates and names of all stations I need to pass to two separate arrays. After I find all stations, I draw the circle and line using the related information such as coordinates and names of the station by using the “drawingLines” and “drawingCircle” methods. Also, write the names of some stations in the background by using “writingStationName”. Finally, I make an animation with the given information, but I use “writingStationName”, “drawingCircle”, and “drawingLines” methods because when I draw the current station with a bigger point, it leaves a trace there and breaks my animation, so I need to create all things again except for changing the color of the stations which have been passed to orange.



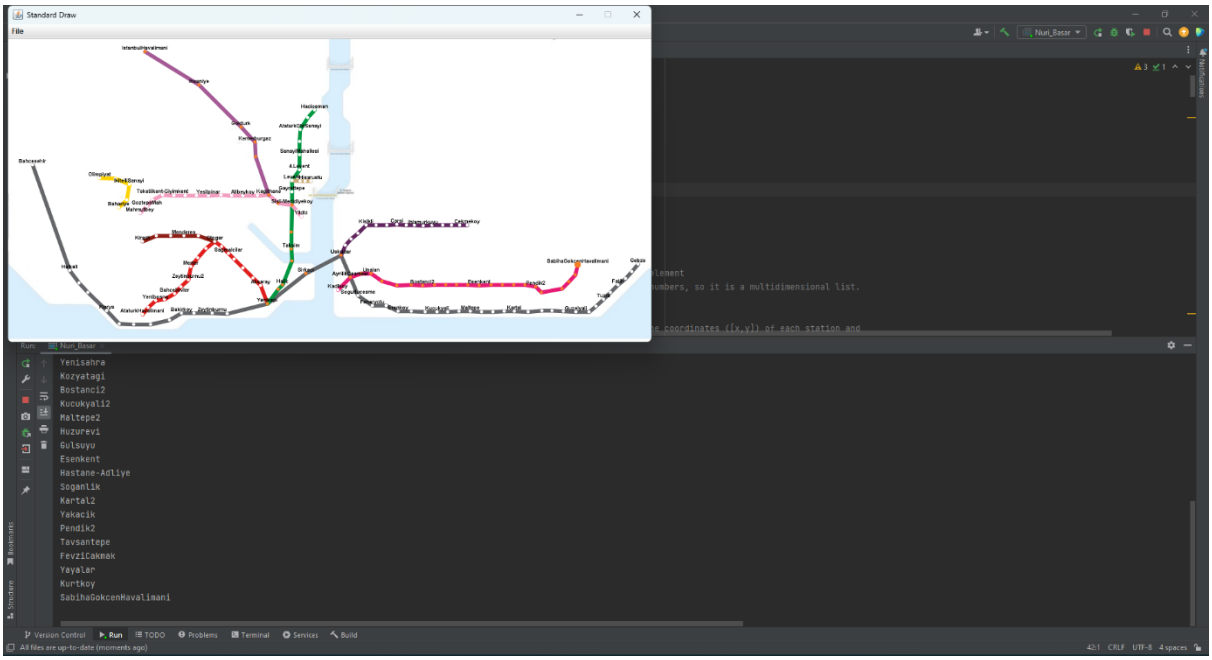
Test Case1: Levent → Uskudar



Test Case2: Aksaray → Bahariye



Test Case3: Bogazici → Kadikoyyy



Test Case4: IstanbulHavalimani →SabihaGokcenHavalimani