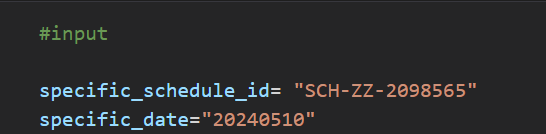
TEAM 52

We have made 3 kinds of solutions

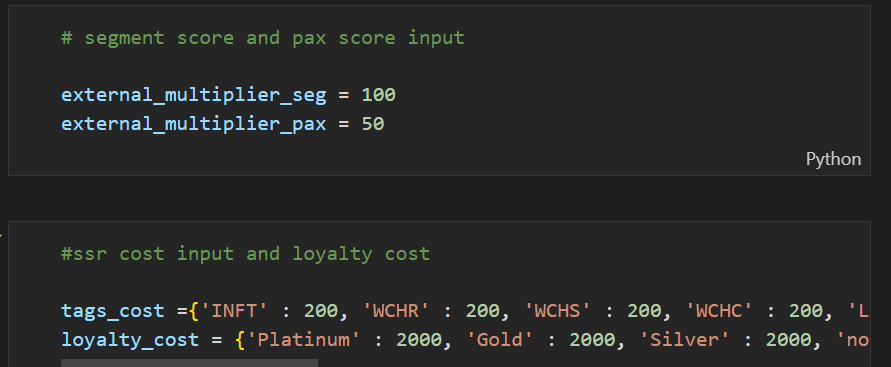
Solution1

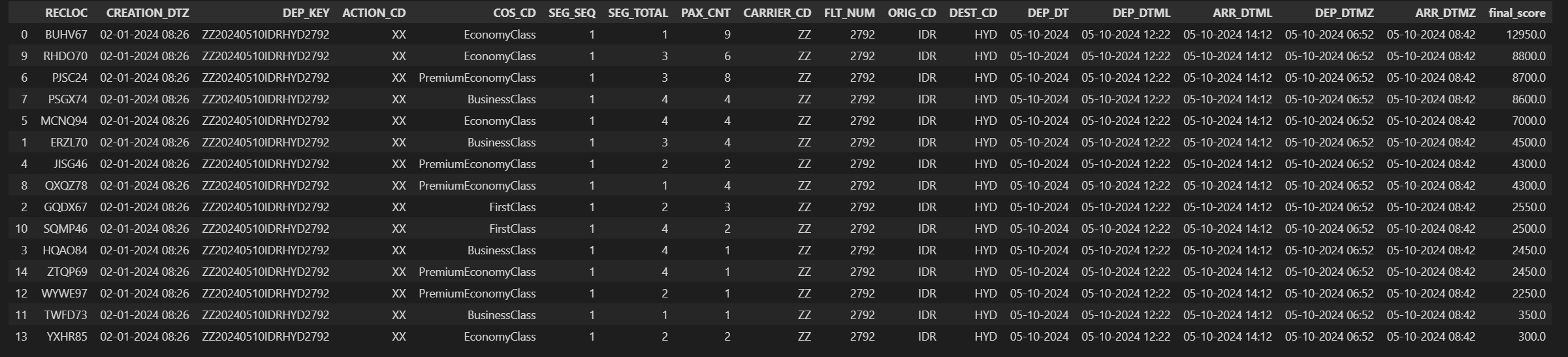
All files related to this are in folder1

When a particular flight is cancelled , the input is given as particular schedule id and date

STEP 1

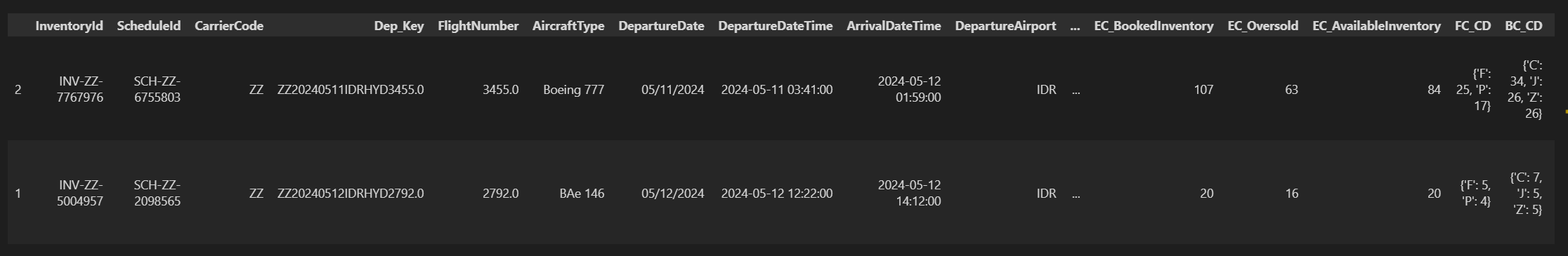
Find the affected passengers and sort the passengers in priority order

Score values can be modified by changing values in corresponding variables

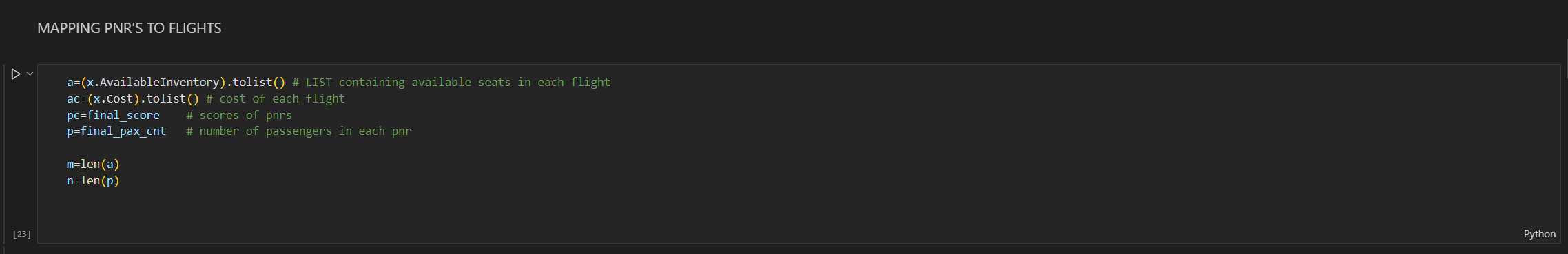
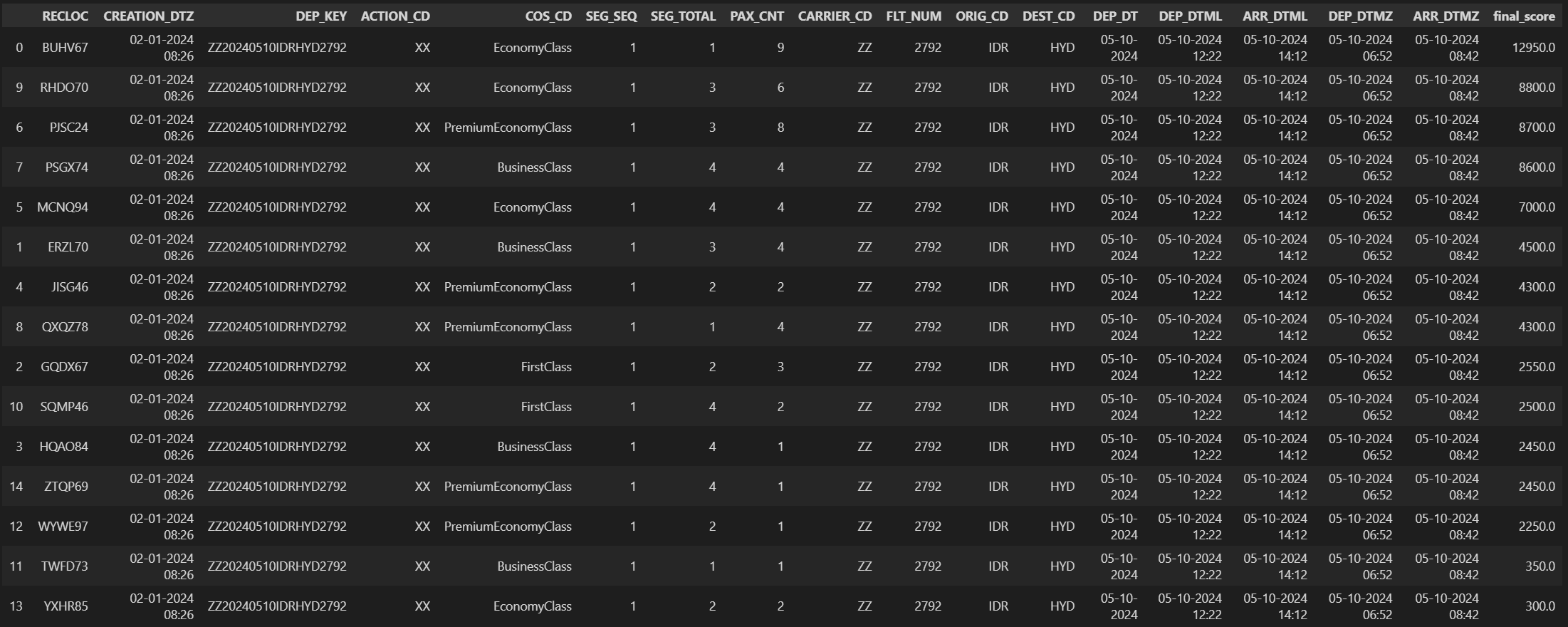
The list of total PNR’S of affected passengers and scores are generated

STEP2

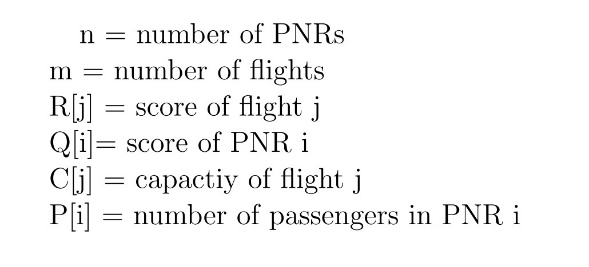
FIND THE BEST ALTERNATE FLIGHTS BASED ON TIME DIFFERENCE

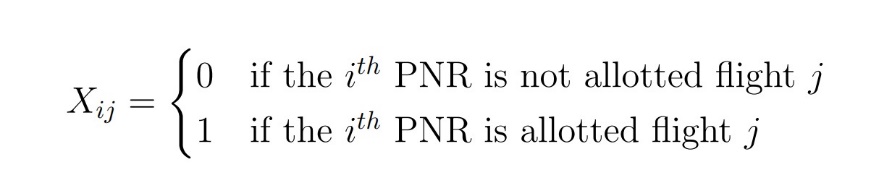
THE LIST OF FLIGHTS AND THEIR SCORES IS GENERATED AS FOLLOWS

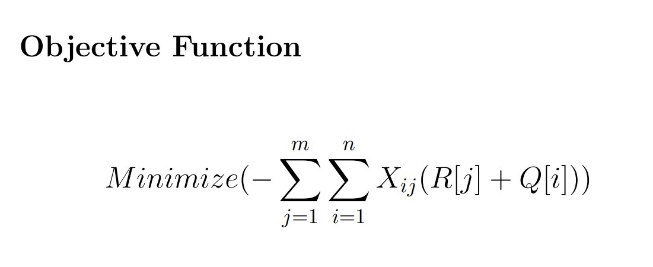
STEP3

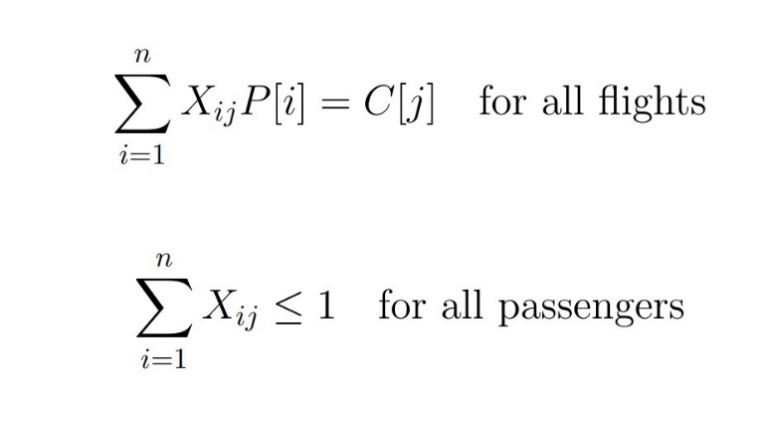


Now we have to allot pnr’s to flights , To do this we designed a linear programming as follows



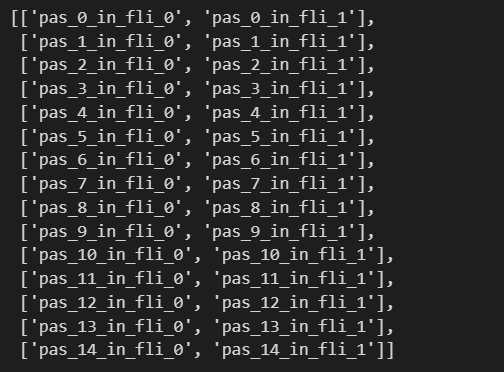


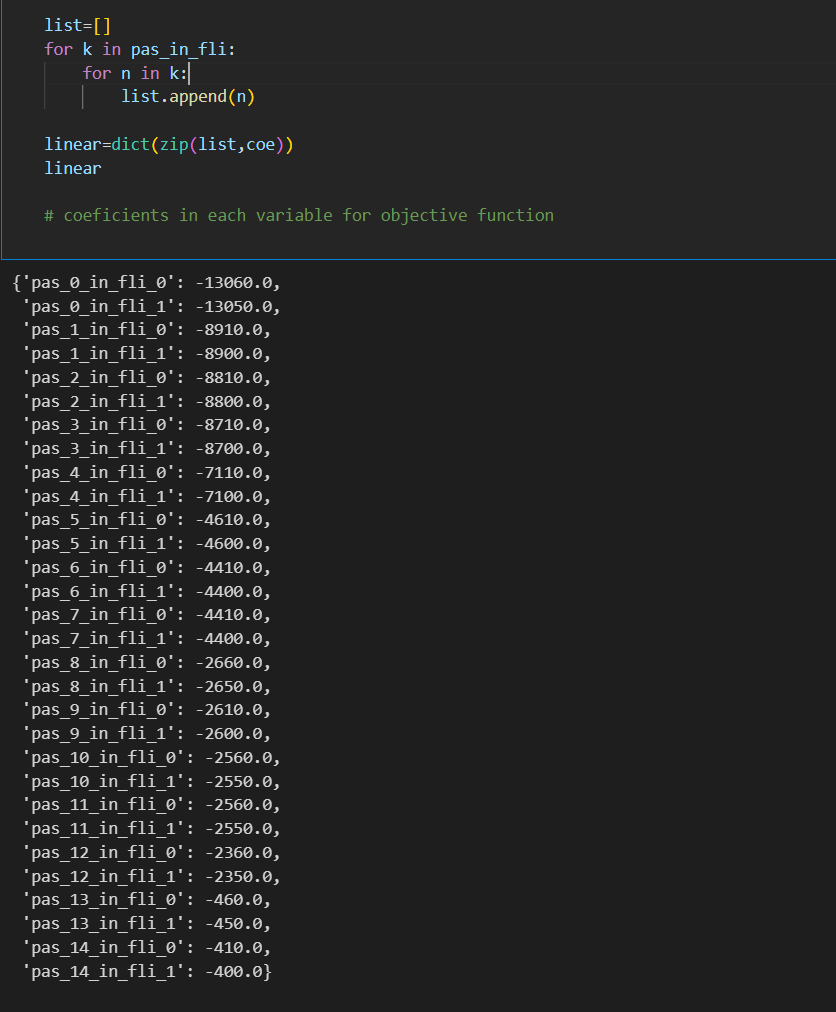


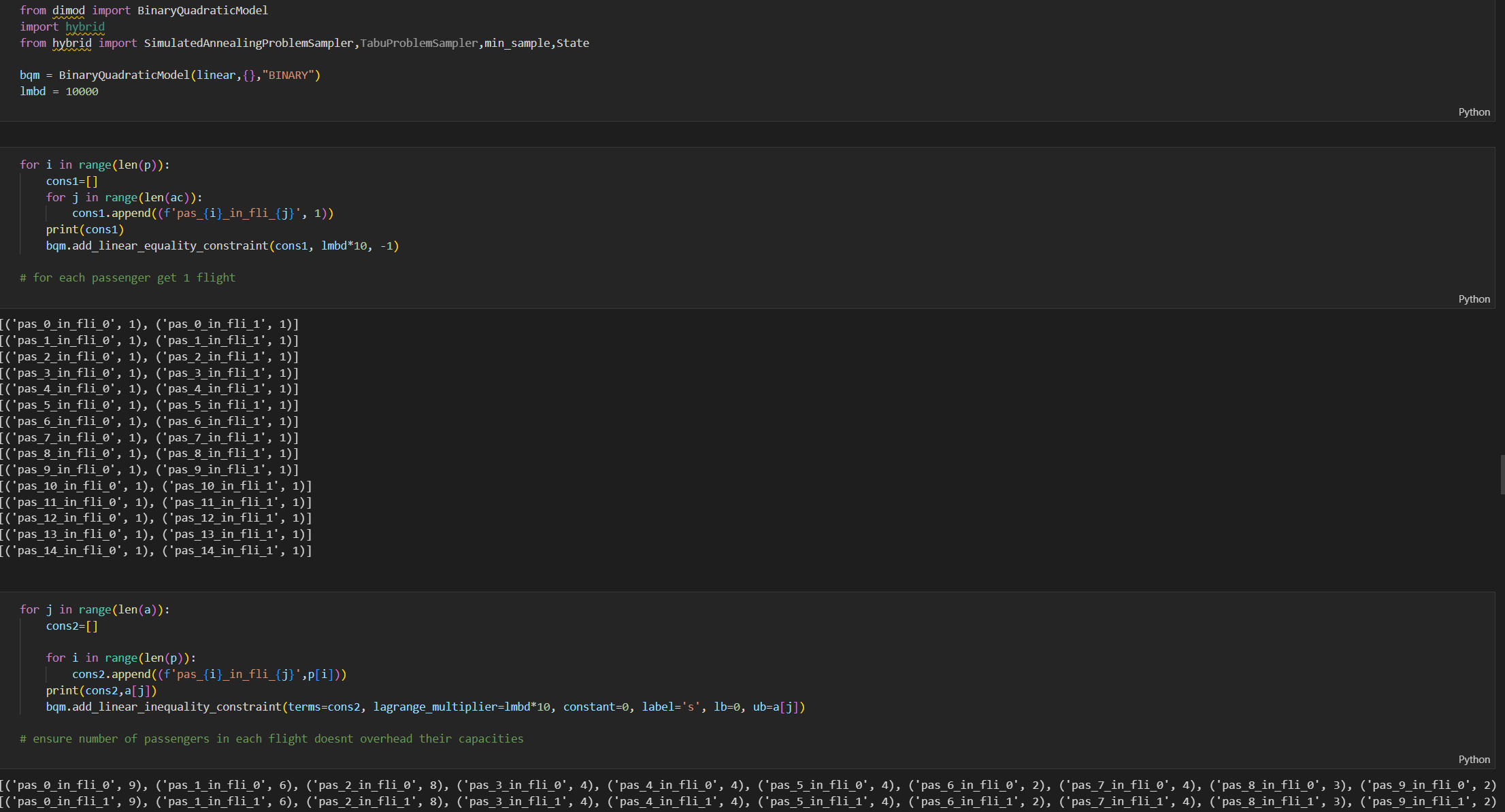
constrains

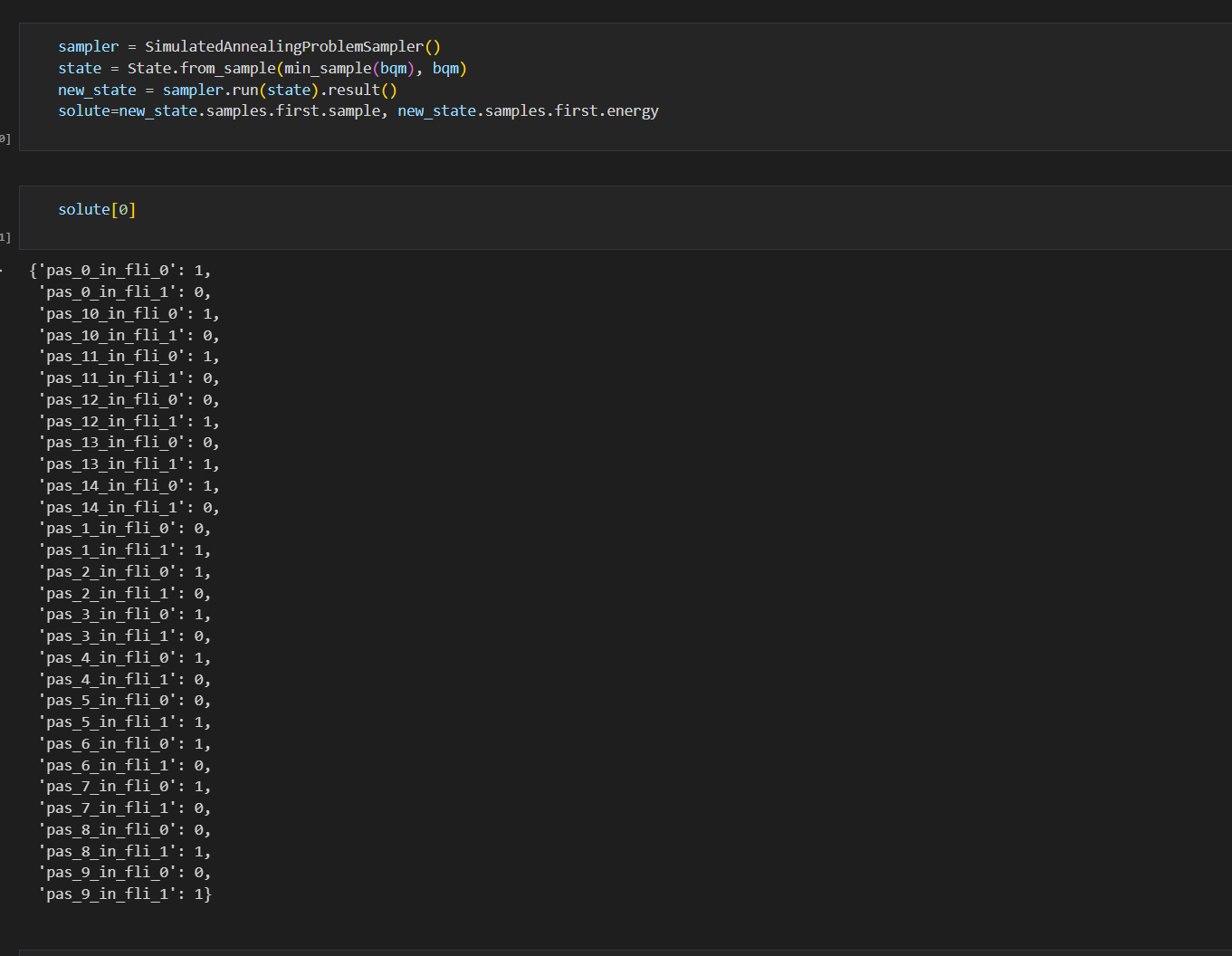
The linear programming is converted to simple Qubo and we used Dwave’s quantum simulator to

Generate solution set

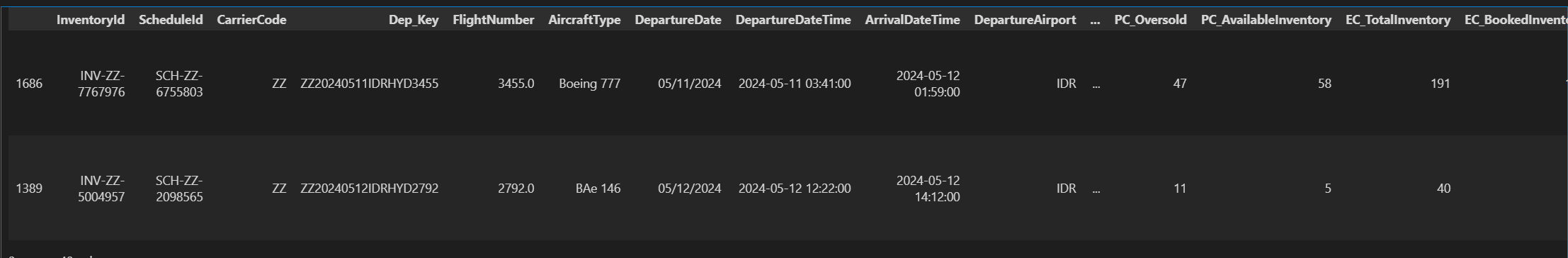
Variable mapping

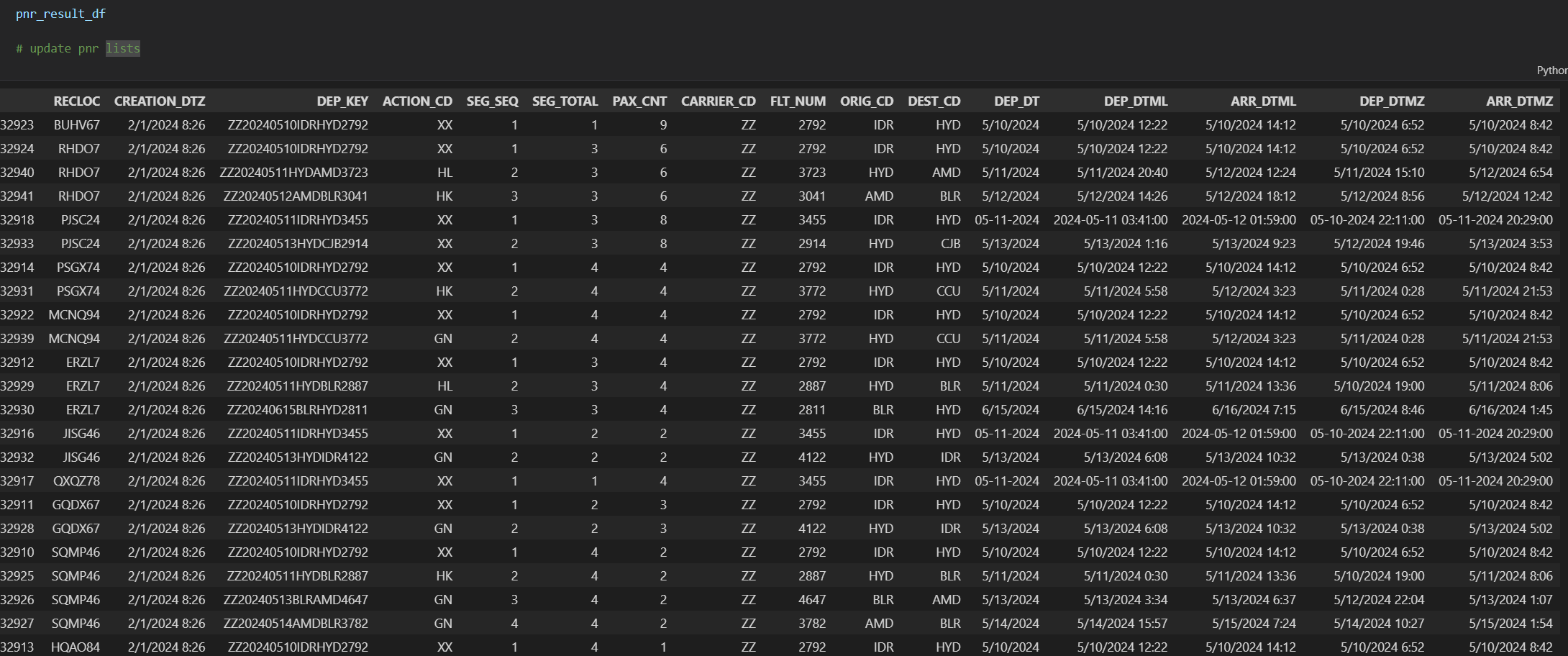
generating objective function

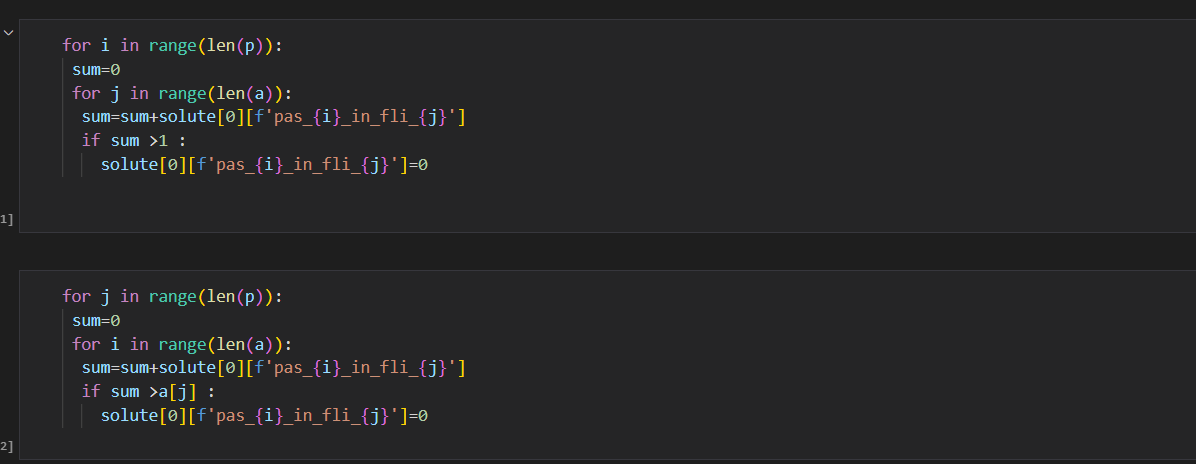
Addition of constrains

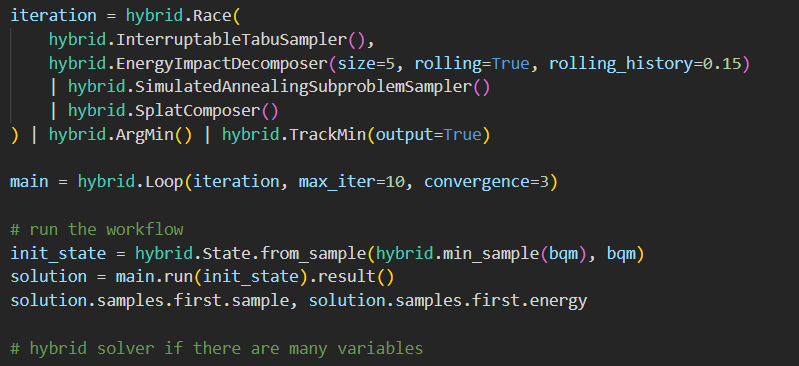
GENERATING SOLUTION

CONVERTING SOLUTION FROM BINARY VARIABLES IN REQUIRED FORM AND SHOWING RESULTS

IN TABLE



Note : the simulator solution may sometimes break inequalities there some checker functions are added to remove anomalies

If number of pnr’s or number of flights are high then we may need to use hybrid solver

3) ideally in this case , the top pnr’s according to scores should get corresponding best classes in

their respective flights, but we were unable to show class mappings as of now

Solution 2

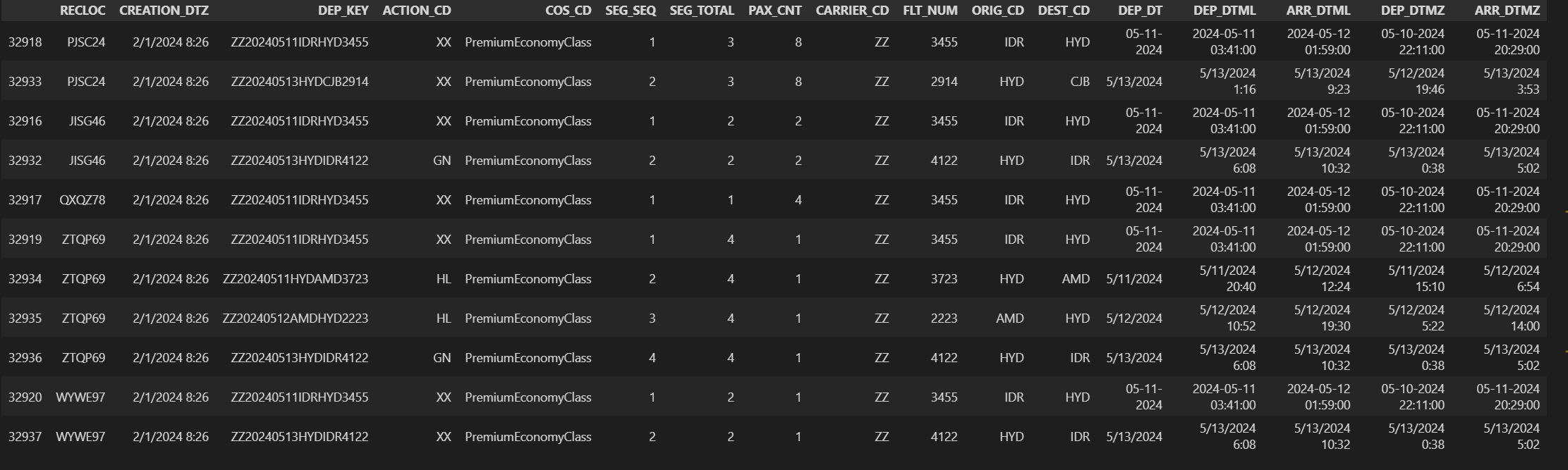
Corresponding files are in folder named 2

The code file is in fintest.ipynb

The strategy is similar as above except

Here, we assumed a pnr of particular class should get same class after reschedule

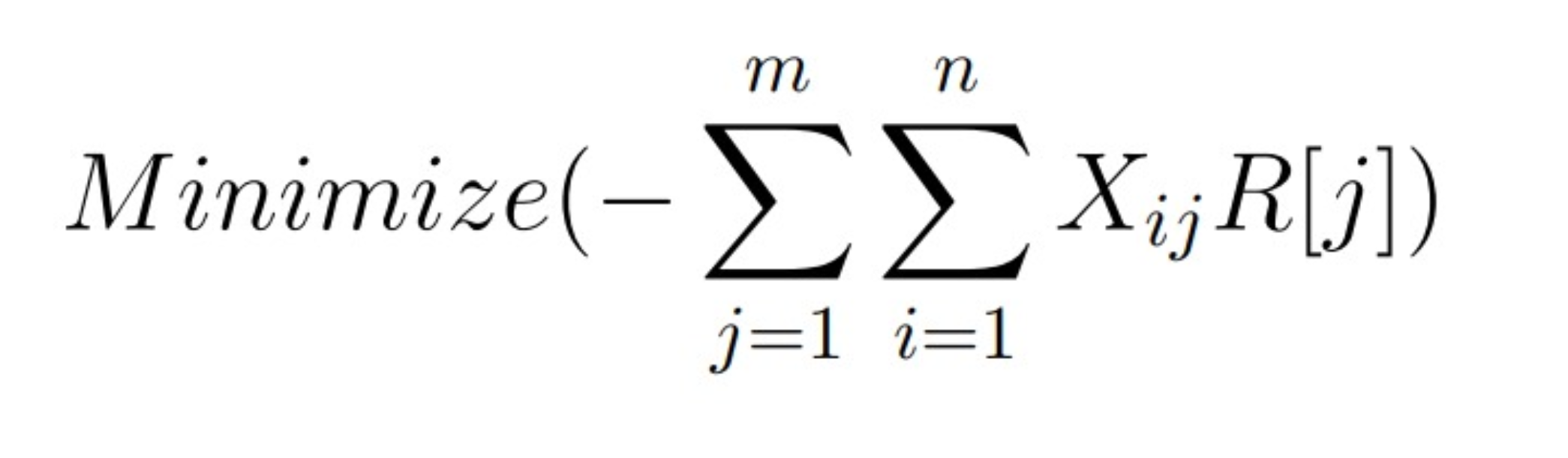
So, the algorithm runs for each class separately and output for each class is shown separately



Solution3

The corresponding files are in folder named “3”

In this case , we wish to accommodate as many passengers as possible

So, the objective function is changed as follows

The preference for passengers is neglected so the term Q[i] is removed

This gives us a simple and most efficient packing of passengers

The output is shown below

