```
import random

'''
0 = empty seat
1 = randomly assigned economy seat
2 = specifically assigned premium seat
'''

economy_seats = 0
premium_seats = 0
num_rows = int(input("How many rows are in the plane? "))
num_columns = int(input("How many columns are in the plane? "))
seats = num_rows*num_columns

# making a 2D array of the plane of rows, cols
def make_a_plane(rows, cols):
  plane = [[0 for x in range(cols)] for y in range(rows)]
  return plane

def show_plane(plane):
  for i in plane:
    print(i)


# this function changes the value of a seat at x,y:
def assign_seat(plane,seat_row,seat_column,value):
  plane[seat_row][seat_column] = value

def choose_seat(plane):
  global seats
  global premium_seats
  show_plane(plane)
  seat_row = int(input("What row would you like? "))
  seat_column = int(input("What column would you like? "))
  if plane[seat_row][seat_column] == 2:
    print("Sorry, that seat is taken")
    choose_seat(plane)
  elif plane[seat_row][seat_column] == 0:
    print("That seat is available!")
    assign_seat(plane,seat_row,seat_column,2)
    seats -= 1
    premium_seats += 1
    print('seats left = ',seats)
    print('premium seats = ',premium_seats)
    print('economy seats = ',economy_seats)
  elif plane[seat_row][seat_column] == 1:
    print("That seat is available!")
    assign_seat(plane,seat_row,seat_column,2)
    economy(plane)
    premium_seats += 1
    seats -= 1
    print('seats left = ',seats)
    print('premium seats = ',premium_seats)
    print('economy seats = ',economy_seats)

def economy(plane):
  seat_row = random.randint(0, num_rows-1)
  seat_column = random.randint(0,num_columns-1)
  if plane[seat_row][seat_column] !=0:
    economy(plane)
  else:
    assign_seat(plane,seat_row,seat_column,1)


#~~~~~Main~~~~~#
plane = make_a_plane(num_rows,num_columns)

while seats !=0:
  customer = int(input("What kind of customer are you? (1 for economy, 2 for premium) "))
  num_tickets = int(input("How many tickets are you purchasing? ")) #added user input for
number of tickets to determine whether customer is buying for group or self
  if customer == 1:
    if num_tickets == 1: #added a nested conditional to execute original code where one ticket
was assumed
      economy(plane)
      economy_seats += 1
      seats -= 1
      print('seats left = ',seats)
      print('premium seats = ',premium_seats)
      print('economy seats = ',economy_seats)
    else: #tries to deal with placing group tickets together unless displaced by premimum
      zero_counter = 0 #counts how many 0s in a row as array is traversed
      for x in range(num_rows):
        if zero_counter == num_tickets:
          break #if consecutive empty seats matches num_tickets, loop breaks
        for y in range(num_columns):
          if zero_counter == num_tickets:
            break #ditto
          if x == num_rows - 1 and y == num_columns - 1: # if the entire array is traversed without
finding the correct number of seats, customers are randomly assigned to remaining seats; the
position of this code is currently preventing it from assigning a group of 3 in the last row
            while num_tickets > 0:
              economy(plane)
              economy_seats += 1
              seats -= 1
              num_tickets -= 1
            break
          if y == 0:
```

```
      zero_counter = 0 #resets to 0 if starting in a new row to prevent seats from being assigned non-adjacently when switching from one row to the next
     if plane[x][y] == 0:
      zero_counter += 1
      #print(zero_counter) #added this just to troubleshoot
      if zero_counter == num_tickets: #this condition means we've found the correct number of adjacent seats and can assign now
       for z in range(num_tickets):
        assign_seat(plane, x, y - z, 1) #will assign economy seat to current position in array and the positions directly before it
        economy_seats +=1 #accounting
        seats -= 1 #accounting
       #Need to add code to deal with scenario where seats left don't match this condition
     else: #resets counter to 0 if a non-zero value is detected while traversing array or if row boundary is reached
      zero_counter = 0
   print('seats left = ',seats)
   print('premium seats = ',premium_seats)
   print('economy seats = ',economy_seats)



  ''' Alex's code
  counter = 0

  while counter < num_tickets:
    plane[seat_row][seat_column] == 0:
    assign_seat(plane,seat_row,seat_column,1)
    seats -= 1
  '''

 if customer == 2:
  while num_tickets > 0: #Changed this to while loop to keep letting user choose seats for number tickets desired
   choose_seat(plane)
   num_tickets -= 1
 print("Here is your seat:")
 show_plane(plane)
 print('\n')

print("Plane is full!")


'''
Additional Ideas:
-Finalize dimensions of plane: 5 rows, 3 columns, no dividing aisle

-How to deal with group tickets
```

-If 2/3 tickets (regardless of customer type?), seat them together in the same row on the same side of the dividing column if we have one.
-If >3 tickets (regardless of customer type?), seat them first in the same row, then in adjacent row.

-One algorithm for how premium bumps economy: seat economy together when possible, but if premium chooses their seat, they are randomly assigned and will never be purposely placed together again except through chance
-Need code for number of tickets input
-Need code to assign adjacent seats
-Need code for accounting updates
-Algorithm 2 for bumping groups: Do the same as above but code economy group tickets as 3 (a new category) and try to do them a courtesy if possible

Lower Priority Ideas?
-Add an empty column that visually divides the plane?

-Make UI more realistic for rows/columns when customer chooses seat(s); might require 2 2d arrays, one for customer type, one for seat assignation
'''