# i'm creating restaurant booking system.  
# you can check all the tables for availability.  
# you can clear the reservation2, or you can overwrite existing booking.  
# further extension can be added by adding an external list to facilitate  
# more days/weeks(eventually - a calendar, but you will need a database for that)  
  
class Restaurant:  
  
 def \_\_init\_\_(self, amount\_of\_tables):  
 self.amount\_of\_tables = amount\_of\_tables  
 self.blank\_table = [0, '-empty-', 123456789]  
 self.tables = [self.blank\_table for i in range(self.amount\_of\_tables)]  
  
 def table\_printout(self, input\_table\_number):  
 n = input\_table\_number  
 print("Table n." + str(n + 1))  
 if self.tables[n][0] == 0:  
 print(" - is available")  
 else:  
 print(" - time: " + str(self.tables[n][0]))  
 print(" - name: " + self.tables[n][1])  
 print(" - phone: " + str(self.tables[n][2]))  
  
 # showing all tables  
 def show\_tables(self):  
 for i in range(self.amount\_of\_tables):  
 self.table\_printout(i)  
  
 # making a booking  
 def book\_table(self, number, time, name, phone):  
 self.tables[number - 1] = [time, name, phone]  
  
 # clearing a booking(setting a blank\_table)  
 def clear\_table(self, number):  
 self.tables[number - 1] = self.blank\_table  
  
 def int\_input(user\_int\_input):  
 while True:  
 try:  
 some\_integer = int(input(user\_int\_input))  
 return some\_integer  
 except ValueError as e:  
 print("Not a proper number! Try again")  
  
  
# creating a restaurant with 6 tables  
rest1 = Restaurant(6)  
  
# creating bookings for tables 1,3,4  
rest1.book\_table(1, 15, 'Smith', 863967861)  
rest1.book\_table(3, 18, 'Smith2', 864447861)  
rest1.book\_table(4, 17, 'Jones', 86000861)  
  
exit1 = False  
  
while not exit1:  
 print("Please select an option:")  
 print("1 - view tables, 2 - add booking, 3 - clear booking, 0 - EXIT")  
  
#checking for proper input type  
 while True:  
 try:  
 userinput = int(input())  
 break  
 except ValueError:  
 print("Not a valid number. Try again...")  
  
 if userinput == 1:  
 print(rest1.show\_tables())  
  
 if userinput == 2:  
 print("Select table number:")  
 user\_input\_table\_number = int(input())  
 if rest1.tables[user\_input\_table\_number - 1][0] != 0:  
 print("Table n. " + str(user\_input\_table\_number) + " is booked")  
 else:  
 print("Choose time (between 2pm and 9pm):")  
 while True:  
 try:  
 table\_time = int(input())  
 break  
 except ValueError:  
 print("Not a valid number. Try again...")  
  
 if table\_time < 2 or table\_time > 9:  
 raise Exception('Time must be between 2 and 9. Selected time was: {}'.format(table\_time))  
  
 table\_name = input("Enter name:")  
 table\_phone = input("Enter phone number:")  
 rest1.book\_table(user\_input\_table\_number, table\_time, table\_name, table\_phone)  
  
 if userinput == 3:  
 print("Select table number:")  
 user\_input\_table\_number = int(input())  
 rest1.clear\_table(user\_input\_table\_number)  
  
 if userinput == 0:  
 exit1 = True