```
#AIRLINES TICKET BOOKING SYSTEM

#PASSENGER CLASS

class Passenger:

  #CONSTRUCTOR

    def __init__(self, pid, name, flightNum, seatNum):
        self.pid = pid    #ATTRIBUTES
        self.name = name
        self.flightNum = flightNum
        self.seatNum = seatNum

    #METHOD

    def display(self):
        print("ID : ", self.pid)
        print("Name : ", self.name)
        print("Flight Number : ", self.flightNum)
        print("Seat Number : ", self.seatNum)
        print()

#ECONOMY & BUSINESS CLASS
#INHERITANCE

class EconomyPassenger(Passenger):
    def __init__(self, pid, name, flightNum, seatNum):
        super().__init__(pid, name, flightNum, seatNum)

#SUPER HELPS IN ACCESSING ATTRIBLUTES FROM CONSTRUCTOR

class BusinessClassPassenger(Passenger):
    def __init__(self, pid, name, flightNum, seatNum):
        super().__init__(pid, name, flightNum, seatNum)

#LIST

passengers = []
cancelled = []

#FUNCTIONS
#TO ADD PASSENGERS

def AddPassenger():
    pid = input("Enter passenger ID: ")
    name = input("Enter passenger Name: ")
    flightNum = input("Enter passenger Flight Number: ")
    seatNum = input("Enter passenger Seat Number: ")
    pClass = input("Enter passenger class (economy/business): ").lower()

    if pClass == "business":
        passenger = BusinessClassPassenger(pid, name, flightNum, seatNum)
    else:
        passenger = EconomyPassenger(pid, name, flightNum, seatNum)

    passengers.append(passenger)
    print("Booking Successful!!\n")

#TO CANCEL BOOKINGS

def CancelPassenger():
    pid = input("Enter Passenger ID to cancel: ")
    found = False
    for p in passengers:
        if p.pid == pid:
            cancelled.append(p)
            passengers.remove(p)
            print("Booking cancelled..\n")
            found = True
            break

    if not found:
        print("Passenger not found..\n")

#TO SEARCH BY NAME
```

```
def SearchName():
    search = input("Enter name to search: ").lower()
    found = False
    for p in passengers:
        if search in p.name.lower():
            p.display()
            found = True

    if not found:
        print("No match found..\n")

#TO SORT BY NAME

def sort():
    sorted_list = sorted(passengers, key=lambda x: x.name.lower())
    print("Sorted list is:")
    for p in sorted_list:
        p.display()

#TO SEARCH BY ID

def BinarySearch(pid):
    sorted_list = sorted(passengers, key=lambda x: x.pid)
    low = 0
    high = len(sorted_list) - 1
    while low <= high:
        mid = (low + high) // 2
        if sorted_list[mid].pid == pid:
            return sorted_list[mid]
        elif pid < sorted_list[mid].pid:
            high = mid - 1
        else:
            low = mid + 1
    return None

#TO PERFORM STRING OPERATIONS

def stringOperations():
    for p in passengers:
        print("Passenger:", p.name)
        print("Uppercase:", p.name.upper())
        print("Lower case:", p.name.lower())
        print("First letter:", p.name[0])
        print("Last letter:", p.name[-1])
        print("Reversed name:", p.name[::-1])
        print()

#TO RESCHEDULE

def reSchedule():
    for p in cancelled:
        if "cancelled" in p.name.lower():
            p.name = p.name.replace("cancelled", "reScheduled")
    print("Replaced cancelled bookings with rescheduled bookings..\n")


#MAIN MENU

def Menu():
    while True:
        print("AIRLINE TICKET BOOKING SYSTEM\n")
        print("1. Add Booking\n")
        print("2. Cancel Booking\n")
        print("3. Search By Name\n")
        print("4. Sort By Name \n")
        print("5. Search By Passenger ID \n")
        print("6. String Operations\n")
        print("7. Replacing Cancelled With Rescheduled\n")
        print("8. Show All Bookings\n")
        print("9. Show Cancelled Bookings\n")
        print("0. EXIT!!\n")

        choice = input("Enter your choice: ")

        if choice == "1":
            AddPassenger()
```

```
            elif choice == "2":
                CancelPassenger()

            elif choice == "3":
                SearchName()

            elif choice == "4":
                sort()

            elif choice == "5":
                pid = input("Enter passenger ID to search: ")
                result = BinarySearch(pid)
                if result:
                    result.display()
                else:
                    print("Passenger not found..\n")

            elif choice == "6":
                stringOperations()

            elif choice == "7":
                reSchedule()

            elif choice == "8":
                if not passengers:
                    print("No bookings available..\n")
                else:
                    print("All Bookings\n")
                    for p in passengers:
                        p.display()

            elif choice == "9":
                if not cancelled:
                    print("No cancelled bookings available..\n")
                else:
                    print("All Cancelled Bookings\n")
                    for p in cancelled:
                        p.display()

            elif choice == "0":
                print("Exiting System!!\n")
                break
            else:
                print("Invalid Choice!!\n")

    #CALLING MAIN MENU

    Menu()
```

Show hidden output

Start coding or generate with AI.