Hw2 Pseudo code

Date Created	@October 2, 2022 10:55 PM
Assign	
Status	
Priority	
Due Date	@10/02/2022
Property	
URL	

1. Seat Definition
    a.

```
Seat_Number //
Seat_Name // init the seat name as N/A, if the name has changed, then it was taken
```

2. Display the seating Map
    a. Function **Show_main_menu()** // Return to the main menu when there is no valid input.

```
Show_main_menu()
{
  1.Clear the screen
  2.Print all Menu the content

}


printMenu()
{

}

option_for_menu()
{
  printMenu();
```

```
   input ->
   Switch {
       case1:
           task1;
           exit_current_menu()
       case2:
   }
 }
```

3. Print a Passenger manifest

```
print_passager_manifest()
{
  Seatnumber_assending
  Seatnumber_dessending
  Alphabetical_assending
  Alphabetical_dssending


}
print_seatMap()
{

}
```

4. Load Previous “Seat Assignment and Passenger Manifest”
    a. to save (and later load) the current seat assignments and passenger manifest from file(s)

```
Load_passage_manifest()
{

}
```

5. Save current “Seat Assignment and Passenger Manifest”
    a. to save (and later load) the current seat assignments and passenger manifest from file(s)

```
Save_passage_manifest()
{
```

```
}
```

b. AutoSave

```
auto_save__manifest()
{

}
```

6. Choose a Seat
   a. **Function Initial_seat_selection();**
      i. Type: Fist class, Business Class, Economy Class.
         1. Define FirstClass, businessClass, Economy Class
         2.
      ii. Generate random seat numbers for seat allocation.

```
RandomNumGenerator()
{
  int = num_generator()
  if(int not seen before)
}
```

      iii. Check whether the current seat has been allocated.

```
Check_Whether_this_seat_is_available()
```

      iv. Check with user which class to go? if not available First Class → Business Class → Economy→ next flight in 3 hours.
         1. **Which Class the user choose?**

```
Which_class_the_user_input()
{
  FirstClass()
  BusinessClass()
```

```
  Economy()
}
```

b. **Function Final seat Assignment()**

Upgrading to higher class.

1. Class full, no inquiry
    a.

```
bool Check_the_higherClass_is_FULL()
{

}
```

2. if Avaiable, start to choose for Final class.

```
Bool check_theLowerClass-is_full()
{

}
```

3.

Downgrade to Lower class.

1. Class full, no inquiry
2. if Avaiable, start to choose for Final class.

c. Initial Selection became the final selection

d. if the upgrade/downgrade option is available. the user should be notified about for both higher or lower class.

e. AutoSave

```
Trigger_autoSave_every_1_secs();
{
  vector.save()
}
```

7. Cancel seat assignment
    a. Cancel the seat(with a seat number and name match)

```
 Cancle_the_Seat(str name, int seat_num)
{
  Delete the seat from the reserved(vector)
}
```

    b. AutoSave- Trigger the auto_save() every 1 s
8. Print a boarding pass for a chosen passenger or seat number
    a. AutoSave- Trigger the auto_save() every 1 s
9. Quit the program

```
// Reture to the main menu when the user doen not gave valid input.

Function Show_main_menu()


End
```