```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "WeatherDetails.h"


struct weather * readWeatherInfo(int day)
{

    struct weather * wthr = (struct weather*)malloc(sizeof(struct weather)*day);
    printf("Enter the Weather Details: \n");
    for(int i=0;i<day;i++)
    {
            printf("Enter City Name: \n");
        scanf("%s", wthr[i].city_name);
        printf("Enter Day: \n");
        scanf("%s", wthr[i].weekdays);

            printf("Enter Max Temperature: \n");
            scanf("%f",&wthr[i].max_temp);
            printf("Enter Min Temperature: \n");
            scanf("%f",&wthr[i].min_temp);
            printf("Enter Humidity: \n");
            scanf("%f",&wthr[i].humidity);

        wthr[i].avg_temp=(wthr[i].max_temp+wthr[i].min_temp)/2.0;

    }
    return wthr;
}

void printWeatherInfo(struct weather * wthr, int day)
{
        printf("###############\n");
    for(int i=0;i<day;i++)
    {
        printf("City Name: %20s \t", wthr[i].city_name);
            printf("Weekdays:  %10s \t",  wthr[i].weekdays);
        printf("Max Temp:  %0.2f\t", wthr[i].max_temp);
            printf("Min Temp:  %0.2f\t", wthr[i].min_temp);
        printf("Avg Temp:  %0.2f\t", wthr[i].avg_temp);
            printf("Humidity:  %0.2f\n",wthr[i].humidity);
            //printf("\n");

     }
    printf("###############\n");
}


struct WeatherLinkedList * createWeatherSLL(struct weather * wthr, int day)
{
    struct WeatherLinkedList * newList = (struct WeatherLinkedList*) malloc(sizeof(struct Weath
    newList->first = NULL;
    newList->count = 0;

    for(int i = 0; i<day; i++)
    {
        if (i==0)
        {
            struct WeatherNode * newNode = (struct WeatherNode*) malloc(sizeof(struct WeatherNo
```

```
 61:             newNode->weather_info = wthr[i];
 62:             newNode->next = NULL;
 63:             newList->first = newNode;
 64:             newList->count++;
 65:         }
 66:         else
 67:         {
 68:             struct WeatherNode * newNode = (struct WeatherNode*) malloc(sizeof(struct WeatherNo
 69:             newNode->weather_info = wthr[i];
 70:             newNode->next = newList->first;
 71:             newList->first = newNode;
 72:             newList->count++;
 73:         }
 74:     }
 75:     return newList;
 76: }
 77:
 78: void printWeatherSLL(struct WeatherLinkedList * newList)
 79: {
 80:     struct WeatherNode * newNode = newList->first;
 81:     while(newNode!=NULL)
 82:     {
 83:         printf("[City Name: %20s \t ", newNode->weather_info.city_name);
 84:     printf("Weekdays: %10s  \t", newNode->weather_info.weekdays);
 85:         printf("Max temp: %0.2f \t ", newNode->weather_info.max_temp);
 86:     printf("Min Temp: %0.2f \t", newNode->weather_info.min_temp);
 87:     printf("Avg Temp: %0.2f \t ", newNode->weather_info.avg_temp);
 88:         printf("Humidity: %0.2f] ==>\n", newNode->weather_info.humidity);
 89:
 90:         newNode = newNode->next;
 91:     }
 92:     return;
 93: }
 94:
 95:
 96: void printStats(struct  WeatherLinkedList * newList)
 97: {
 98:     int i=0, j=0,p=0, min;
 99:     struct weather temp;
100:     struct weather * twthr = (struct weather*)malloc(sizeof(struct weather)*5);
101:
102:     struct WeatherNode * newNode = newList->first;
103:     while(newNode!=NULL)
104:     {
105:         if ((newNode->weather_info.max_temp>= 26) && (newNode->weather_info.max_temp <= 40))
106:             twthr[i++]=newNode->weather_info;
107:
108:         newNode = newNode->next;
109:     }
110:     printf("\n weather records  sorted by Humidity based on the max temp within [26-40]:\n");
111:
112:       for (p = 0; p < i; p++)
113:       {
114:         min = p;
115:
116:         for (j = p+1; j < i; j++)
117:          {
118:             if (twthr[j].humidity < twthr[min].humidity)
119:             min = j;
120:             }
```

```
121:
122:          temp = twthr[min];
123:          twthr[min] = twthr[p];
124:          twthr[p] = temp;
125:    }
126:
127:     for (int y=0;y<i;y++)
128:     {
129:         printf("[City Name: %20s \t ", twthr[y].city_name);
130:         printf("Weekdays: %10s  \t", twthr[y].weekdays);
131:             printf("Max temp: %0.2f \t ", twthr[y].max_temp);
132:         printf("Min Temp: %0.2f \t", twthr[y].min_temp);
133:         printf("Avg Temp: %0.2f \t ", twthr[y].avg_temp);
134:             printf("Humidity: %0.2f] =>\n", twthr[y].humidity);
135:
136:     }
137:
138: }
139:
140:
141: // end of all function
142:
143:
```