SIMPLE CALENDAR APPLICATION USING C

By ASTHA AND DHAWAL

Project Abstract

The calendar application presented here is a very simple console application developed using C programming language.

It is built without using graphics properties; instead, it utilizes many windows properties to give the application a colorful look and feel.

It is compiles in vs code using GCC compiler.

Working

At first the user need to provide which task the application has to perform. For Ex. finding the day.

The user need to provide the date month and year.

The application displays day corresponding to a given date

It also allow user to add notes for a particular date.

The whole calendar of a month can pe printed by providing the month and year by the user.

Screenshots

```
1. Find the day
2. Print calendar of a month
3. Add Note
4. Exit
Enter your choice: 1
Enter the day, month and year: 20 05 2003
The day is : Tuesday
1. Find the day
2. Print calendar of a month
3. Add Note
4. Exit
```

```
1. Find the day
2. Print calendar of a month
3. Add Note
4. Exit
Enter your choice: 2
Enter the month and year: 05 2022
Please enter 's' to see the notes
 Press any other key to continue

        May 2022
Su   Mo   Tu   We   Th   Fr   Sa
 1    2    3    4    5    6    7
 8    9   10   11   12   13   14
15   16   17   18   19   20   21
22   23   24   25   26   27   28
29   30   31
```

Screenshots

```
1. Find the day
2. Print calendar of a month
3. Add Note
4. Exit
Enter your choice: 3
Enter the day, month and year: 01 01 2022
Enter the note: HAPPY NEW YEAR
Note added sucessfully
```

```
1. Find the day
2. Print calendar of a month
3. Add Note
4. Exit
Enter your choice: 2
Enter the month and year: 01 2022
Please enter 's' to see the notes
 Press any other key to continue

        January 2022
Su   Mo   Tu   We   Th   Fr   Sa
                      | 1|
 2    3    4    5    6    7    8
 9   10   11   12   13   14   15
16   17   18   19   20   21   22
23   24   25   26   27   28   29
30   31
s
Here are list of notes for 1 2022
1: HAPPY NEW YEAR
```

Source code

```c
#include <stdio.h>
#include <stdlib.h>

int isLeapYear( int year );          /* True if leap year */
int leapYears( int year );           /* The number of leap year */
int todayOf( int y, int m, int d);   /* The number of days since the beginning of the year */
long days( int y, int m, int d);     /* Total number of days */
void calendar(int y, int m);         /* display calendar at m y */
int getDayNumber(int d,int m,int y);
char *getName(int day);


void flush()
{
    int c;
    while ((c = getchar()) != '\n' && c != EOF);
}

typedef struct {
    int day;
    int month;
    int year;
    char note[255];
} Note;

int main(int argc, char* argv[]){
    int year,month, day;
    char choice;
    Note note;
    FILE *fp;

    fp = fopen("note.bin", "r");
    if (fp == NULL) {
        fp = fopen("note.bin", "w");
    }
```

```c
        fclose(fp);

        while(1) {
            printf("1. Find the day\n");
            printf("2. Print calendar of a month\n");
            printf("3. Add Note\n");
            printf("4. Exit\n");
            printf("Enter your choice: ");
            scanf("\n%c", &choice);
            switch(choice) {
                case '1':
                printf("Enter the day, month and year: ");
                scanf("%d %d %d", &day, &month, &year);
                printf("The day is : %s\n", getName(getDayNumber(day, month, year)));
                break;
                case '2':
                printf("Enter the month and year: ");
                scanf("%d %d", &month, &year);
                printf("Please enter 's' to see the notes\n Press any other key to continue\n");
                calendar(year, month);
                break;
                case '3':
                printf("Enter the day, month and year: ");
                scanf("%d %d %d", &note.day, &note.month, &note.year);
                flush();
                printf("Enter the note: ");
                fgets(note.note, 255, stdin);
                fp = fopen("note.bin", "a+");
                if (fp == NULL) {
                    printf("File note.bin can not be opened\n");
                    exit(1);
                }
                fwrite(&note, sizeof(Note), 1, fp);
                printf("Note added sucessfully\n");
                fclose(fp);
                break;
```

```c
                case '4':
                    printf("Bye!!");
                    exit(0);
                    break;
                default:
                    printf("Not a valid option\n");
                    break;
            }
        }
        return 0;
}

int isLeapYear( int y ){
    return(y % 400 == 0) || ((y % 4 == 0) && (y % 100 != 0));
}

int leapYears( int y ){
    return y/4 - y/100 + y/400;
}

int todayOf( int y, int m, int d) {
    static int DayOfMonth[] =
        { -1,0,31,59,90,120,151,181,212,243,273,304,334};
    return DayOfMonth[m] + d + ((m>2 && isLeapYear(y))? 1 : 0);
}

long days( int y, int m, int d){
    int lastYear;
    lastYear = y - 1;
    return 365L * lastYear + leapYears(lastYear) + todayOf(y,m,d);
}

void calendar(int y, int m){
    FILE *fp;
    Note* notes, note;
```

```c
    int len, j, hasNote = 0;
    char choice;
    const char *NameOfMonth[] = { NULL/*dummp*/,
        "January", "February", "March", "April", "May", "June",
        "July", "August", "September", "October", "November", "December"
    };
    char Week[] = "Su  Mo  Tu  We  Th  Fr  Sa";
    int DayOfMonth[] =
        { -1,31,28,31,30,31,30,31,31,30,31,30,31 };
    int weekOfTopDay;
    int i,day;

    weekOfTopDay = days(y, m, 1) % 7;

    fp = fopen("note.bin", "rb");
    if (fp == NULL) {
      printf("Couldn't read notes\n");
    }
    len = 0;
    while(fread(&note, sizeof(Note), 1, fp)) {
      if (note.year == y && note.month == m) {
        len++;
      }
    }
    rewind(fp);
    j = 0;
    notes = (Note*) malloc (sizeof(Note) * len);
    while(fread(&note, sizeof(Note), 1, fp)) {
      if (note.year == y && note.month == m) {
        notes[j] = note;
        j++;
      }
    }

    fclose(fp);
```

```c
143          if(isLeapYear(y))
144              DayOfMonth[2] = 29;
145      printf("\n       %s %d\n%s\n", NameOfMonth[m], y, Week);
146
147      for(i=0;i<weekOfTopDay;i++)
148          printf("    ");
149      for(i=weekOfTopDay,day=1;day <= DayOfMonth[m];i++,day++){
150          hasNote = 0;
151          for (j = 0; j < len; j++) {
152              if (notes[j].day == day) {
153                  printf("|%2d| ",day);
154                  hasNote = 1;
155                  break;
156              }
157          }
158          if (hasNote == 0) {
159              printf("%2d   ",day);
160          }
161          if(i % 7 == 6)
162              printf("\n");
163      }
164      printf("\n");
165      scanf("\n%c", &choice);
166      if (choice == 's') {
167          printf("Here are list of notes for %d %d\n", m, y);
168          for (j = 0; j < len; j++) {
169              printf("%d: %s\n", notes[j].day, notes[j].note);
170          }
171      } else {
172          return;
173      }
174  }
```

```c
int getDayNumber(int d, int m, int y){ //retuns the day number
    static int t[] = {0, 3, 2, 5, 0, 3, 5, 1, 4, 6, 2, 4};
    y -= m < 3;
    return (y + y/4 - y/100 + y/400 + t[m-1] + d) % 7;
}

char *getName(int day){ //returns the name of the day
    switch(day){
        case 0 :return("Sunday");
        case 1 :return("Monday");
        case 2 :return("Tuesday");
        case 3 :return("Wednesday");
        case 4 :return("Thursday");
        case 5 :return("Friday");
        case 6 :return("Saturday");
        default:return("Error: Invalid Argument Passed");
    }
}
```