Code

```
#include "library.h"



// Finish  code C++ method because UNIX Putty is not working



// Make a function for the year/month,day to be used within the calendar output
// Y/ 4= total days within a quarter of a regular year

int Year(int year)
{
	if (year/4==91.25)
		return 365;
	else return 366;

}



// Separate months from the amount of days each one has
// Do same for Leap Years on February

//On years with 365 days this would not be concluded as a leap year, therefore there for their is
28 days rather than 29 days in the month of February

//& == and statement

int Month(int month,int year)
```

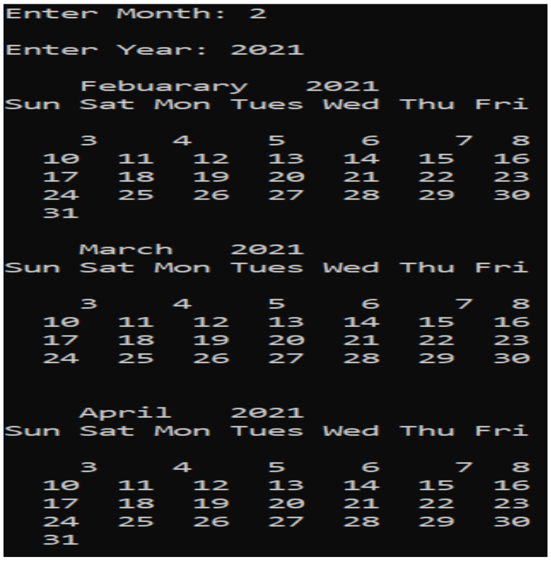
```
{
        if(month==1)
                return 31;
        else  if(month==2)
                return 28;
        else if (Year(year)==366 & month==2)
                return 29;
        else if (month ==3)
                return 31;
        else if (month ==4)
                return 30;
        else if (month ==5)
                return 31;
        else if (month ==6)
                return 30;
}




int Day(int day,int month, int year )
{
        int e=day;

        if(month>0)
        {
                e=Month(month,year);
        }

        return e;
}



int century_forever (int day,int month,int year,int century)
{
        int e=day;
        if (month>1) e=(month-1,year==year,century+1);
```

```
    month=month-1;

    int e=day;
    if (month>0) e=(month-1,year==year,century+1);

    month=month;

    if (year>=0) e=e+Year(year-1);


    return e;

}




//To create an array of for the weekdays

string week_week (const int day_2,const int month_2, const int year_2)
{ string week_week[]= { "Friday", "Saturday",
"Sunday" , "Monday" ,"Tuesday",
"Wednesday" ,"Thursday" ,"Friday"};
new_line();


//Separate string by 7 days
//Follow up with an if statement

if (week_week)
    return ;
    string month_month[(7)];


string week_week_2=week_week[(day_2,month_2,year_2)%7];
new_line();
return(week_week_2);
```

```
new_line();
}




//To create an array for the months of the year
//Make months countdown from its int value

string months_month( const int month)
{
    string months_month[]={"January", "February", "March" ,
        "April", "May", "June", "July",
        "August", "September", "October",
        "November", "December"};

    if(month<12)
        return;

    string month_month[(12)];
    new_line();
    return ((month_month)[month-1]);
}



void calender_1(int const year, int const month)
{

    int const day =1;
    //Add a function for the first day of the month that will repeat

    int tot_1= (day,month,year);
    int tab,tot_1,order_1,x,y,order_2;
    order_2=(day+1,month+1,year);
    order_1= Month(month+1,year);
    tab=(tot_1);


    //proper tabbing
    //Fix syntax error to get your code to function properly

    cout<< "    "<< months_month(month)<< "   "<<year<< "   "<<endl;
    cout<< "Sun Sat Mon Tues Wed Thu Fri" << endl;
    for (x=0; x, x<tab%7;)
    { cout<< " " << endl;
    }



    for(x, y, x<=order_1;y=tab;)
    {

        if(y%1!=0)

        if(x<8)
        {cout<<"  "<<x;}
        if(x<12)
        {cout<<"  "<<x;}


        else if(x>=8)
        {cout<<"  "<<x;}



            cout<<endl;
    }
    cout<<endl;

}

//String to add input of calender
//Add 3 calendars

        void draw_calender_2()
        {



            cout<<"Please Enter Month: ";

            const int month = read_int();
            new_line();

            cout<< "Please Enter Year: ";
```

```
                const int year = read_int();
                new_line();

                calender_1(year,month);
                new_line();

                calender_1(year,month+1);
                new_line();
                calender_1(year,month+2);
}




//Make x > 2000
        //Make Month>1

int main()
{
        //cout<<"Please Enter Month: ";
        //read_input();
        calender_1(2000,1);
        draw_calender_2();

}
```

Output



```
Enter Month: 2

Enter Year: 2021

    Febuarary   2021
Sun Sat Mon Tues Wed Thu Fri

    3    4    5    6    7  8
  10  11  12  13  14  15  16
  17  18  19  20  21  22  23
  24  25  26  27  28  29  30
  31

    March   2021
Sun Sat Mon Tues Wed Thu Fri

    3    4    5    6    7  8
  10  11  12  13  14  15  16
  17  18  19  20  21  22  23
  24  25  26  27  28  29  30


    April   2021
Sun Sat Mon Tues Wed Thu Fri

    3    4    5    6    7  8
  10  11  12  13  14  15  16
  17  18  19  20  21  22  23
  24  25  26  27  28  29  30
  31
```