

Name: Jason Gunn

Number: 06039966

Unit: Further Programming

Unit Code: CMP1024

Assignment Number: 1

Unit Lecturer: Stephen Rank

Index

Page 1 – Cover

Page 2 – Index

Page 3 – UML Diagram

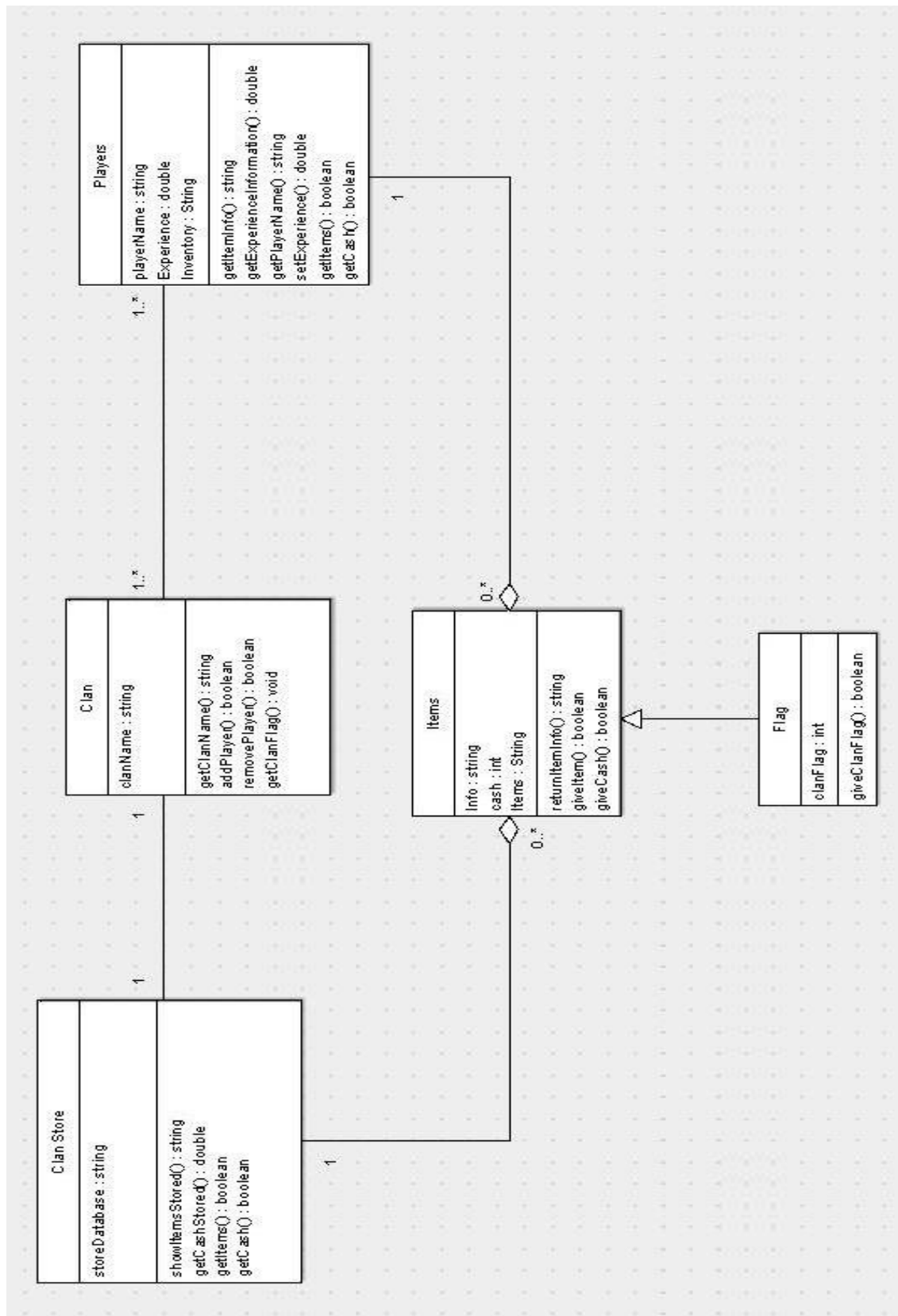
Page 4 – 7 - Program Testing

Page 7 – 22 - Program – Date

Page 22 – 24 – Program – Main

Page 25 – Conclusion And Bibliography

UML Diagram



Program Testing

I faced many problems over the course of this assignment, from tiny bugs to having to start from scratch at points. However I have managed to develop a working piece of code, which I hope fulfils the requirements.

I created each section of code separately, and ensured that it was working, before moving onto the next, I would test code by using `System.out.println()` constantly, as well as compile every few lines to ensure that it was working.

Once I was satisfied that one section of code was working I would move onto the next, and begin the process again. Thanks to formatting correctly I found it very simple to find problems, e.g. if I forgot to put a `}` in I could clearly see where it was meant to be.

Some sections of code required me to rethink my plan, as they either did not work at all, or did not return the desired effect. For example I created a for loop, however had missed out one line, I wanted to add one year each loop, however all that was being printed was the same year over and over. I managed to fix this simply thanks to testing with `System.out.println()`.

Some of the code I completely rewrote, I tried to condense it as much as possible, and keep it as tidy as I could. I did attempt to change large sections of if statements into for loops, however after 2 hours of testing, altering etc I was unable to get them to give the correct output so due to time constraints decided it would be better to have a working piece of code, even though I wanted to make it as tidy as possible.

As I have said before my main methods for testing were with output to screen, and constant compiling. However this was not the only way I tested if things were working:

For Input Month:

During original testing I found that this would accept any value (Within reason) so I could have the month as 1, or 100, due to this I had to create validation rules to prevent the user from entering invalid dates.

I tested this by continually inputting different months, possible, and negative until I was satisfied that no number other than 1 – 12 could be accepted.

For Input Day:

As with the above any number was able to be input, to tackle this I created 3 if statements, to deal with the 3 different month types, 31 days, 30 days and 28 days!

To test this I would input a month of 31 days, and test that I could not put out of range dates in, but that it accepted in range dates, and I then repeated this process for 28 day, and 30 day months.

Originally I had separate if statements for each mother, however this was untidy, and unnecessary so I condensed my code into a much smaller fragment.

For Input Year:

Year again would accept any date, and this is fine as the program will still calculate no matter what year is input. However I felt it sensible to add at least some limitations, so I decided that the valid range would be between 1 and 9999.

Testing of this was as above, entering a combination of both valid and invalid dates, until I was satisfied it was working correctly.

For Difference In Days

For this section of code, I had to rewrite it several times in order to get it working correctly, I found the best method was to calculate the total number of dates from year 0, until the set date for both dates, and then subtract start date from end date.

I tested this by entering many different dates, and adjusting my code until it displayed the correct number of days. To work out the number of days I used a calculator.

For tomorrow()

This initially had many problems, for example if you added 1 to 1/31/2007 it would become 1/32/2007, in order to solve this I again had to add validation rules, to ensure that if you add 1 to the end of a month then the month will increase, and the day will become 1.

I also had to ensure that if 1 was added to 12/31 that the year would increase.

Testing of this involved adding 1 to many different dates, until I found it to be working as I wanted.

For yesterday()

Again I had the same problem with this as the above, except it was with subtracting 1 from the beginning of the month, which would return -1 instead of the end of the previous month. Testing was the same.

I also had to make sure that if 1 was subtracted from 1/1 then the year would decrease.

For monthAdd()

For this section of code I found that originally when adding a month to the end of a year, it would become the 13th month, rather than the 1st of the next year. So I created new validation rules for this, to ensure that this would be handled correctly.

I also had to ensure that if you added a month to the end of a month that it would handle correctly, For example, 1/31/2007, needed to become 2/28/2007. I eventually managed to get this working perfectly also.

Again testing involved putting in many different combinations of dates, and making sure the correct month, year and day was displayed after.

For monthSubtract()

The above is again true for this however in reverse, I had to make sure that taking a month from January would minus a year.

I again tested this by inputting many different dates.

For yearAdd()

This was a very simple element to the program, as I only ever need to add one year, rather than account for changes to months, and days.

However to test it I did enter several different dates until I was satisfied that it was working correctly, and that nothing but the actually year would change.

For yearSubtract()

Again this was very simple to implement, only having to subtract one year from a date.

I did however give it testing until I was satisfied that it was working.

While Loops

I used while loops many times in my program to ensure that invalid data would not be entered by the user, for example when the option was 1, or 0, anything other than that would not be accepted making the program much more secure for the user.

Gap Year Calculation

Gap year calculation took some time to implement, and I learnt some new things about programming. Originally my program gave the total number of days, minus gap years, but I was not happy with this so I decided to research operators, and how gap years work and implement a code to do this.

For some time I was unable to get any output from my code, however after examination I found the error, and corrected it. I had it looping infinitely instead of adding 1 each time.

Once finished I had to test this again by inputting many different dates, and comparing them to correct calculations to ensure that they were correct.

For replay

I implemented a reuse option, which allows the user to start the program again, or to quit, this was simple to do as I had used the same methods in my while loops, Setting

a Boolean to false, and then either re-looping by keeping it false, or setting it to true which breaks the loop.

I did have several problems with this, initially as it would instantly loop back to the beginning and not allow any other options, however this was due first to bad placement of a bracket, and then due to the fact that I changed some of my code. However I managed to get it working eventually.

I tested this by running my program, and looping through it several times, exiting, restarting and repeating.

Overview

I have tried to cover as many testing aspects as possible, to ensure my code is as bug free as possible, and to provide the user with an enjoyable experience, that will not cause the computer to crash, or the program to fail.

I used many different types of data, for example I would try entering words, characters, whole numbers, fractions, negative numbers etc into the different sections to ensure that the program would work correctly, and only allow valid data to be accepted.

I have used many loops, and if statements, and hope that I have created a program that is relatively fool proof.

Because of all my testing I found it necessary to use good validation methods, to make the program as fool proof as possible, I created a plan for each section, outlining what acceptable data would be, and what it would not be, and then created methods that would prevent the invalid data from being input, without crashing the program, or causing other undesired effects.

I feel that overall my program works quite well, and after many hours of testing, and fixing I feel that the program is relatively safe for the user, and hope that others will find this to.

Program

A note – Some of the programming in word does not look how it should due to lines being to long etc.

Section 1 - Date

```
public class Date{

    int[] daysInMonth = {-1,31,28,31,30,31,30,31,31,30,31,30,31}; //This is an array,
    used to represent the number of days in each month, 1 - 12, it starts at -1, so that the
    user can input 1 - 12, rather than 0 - 11.

    protected int monthBirth; //This is used to represent the Birth/Start month the user
    will input.
    protected int dayBirth; //This is used to represent the Birth/Start day the user will
    input.
    protected int yearBirth; //This is used to represent the Birth/Start year the user will
    input.

    protected int month; //This is used to represent today's/the end month that the user
    will input.
    protected int day; //This is used to represent today's/the end day that the user will
    input.
    protected int year; //This is used to represent today's/the end year that the user will
    input.

    public Date(int m, int d, int y, int n, int b, int c){ //public Date is used to get input
    from the user, and to set both dates to the correct values they enter.
        monthBirth = m; //This defines monthBirth as = to m.
        dayBirth = d; //This defines dayBirth as = to d.
        yearBirth = y; //This defines yearBirth as = to y.

        month = n; //This defines month as = to n.
        day = b; //This defines day as = to b.
        year = c; //This defines year as = to c.
    }
    public void setMonthBirth(){ //This section of code is used to get input into
    monthBirth, and to validate it. Also to return error messages where needed.

        boolean monthBirthNull = false; //For each validation method, I have created a
        while loop, which will loop continually while ever my boolean is set to false, however
        as soon as it is set to true, the loop will break, and the next part of code will activate.
        while(!monthBirthNull){
            monthBirth = TextIO.getInt();//This section of code is used to get input from
            the uer and store it into monthBirth.
            if(monthBirth >12){//This section of code states that if the month is greater
            than 12, then the loop should restart as it is invalid.
                System.out.println("Im Sorry This Month Is Invalid (To High), Please Try
                Again!");
                monthBirthNull = false;
            }
            else if(monthBirth <1){//This section of code states that if the month is less
            than 1, the loop should restart as the code is invalid.
                System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
                Again!");
                monthBirthNull = false;
            }
        }
    }
}
```

```
    }
    else{//This section of code states that if the above are not the case, then the
date is valid, and the loop should break.
        monthBirthNull = true;
    }
}
}
}
public void setDayBirth(){//This section of code is used to get input into dayBirth,
and to validate it. Also to return error messages where needed.

    boolean dayBirthNull = false;

    while(!dayBirthNull){
        dayBirth = TextIO.getInt();//This section of code gets input from the user and
stores it in dayBirth
        if(monthBirth == 1 || monthBirth == 3 || monthBirth == 5 || monthBirth == 7 ||
monthBirth == 8 || monthBirth == 10 || monthBirth == 12){//This section of code
takes all the months with 31 days in, and gives them validation.
            if(dayBirth >31){//This states that if a month has 31 days, and a value
higher than this is entered then it will not be accepted.
                System.out.println("Im Sorry This Month Is Invalid (To High), Please
Try Again!");
                dayBirthNull = false;
            }
            else if(dayBirth <1){//This states that if the value is less than 1, it is invalid.
                System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
                dayBirthNull = false;
            }
            else{//This states that if the data is between 1 and 31 then it is valid and the
loop shall be broken.
                dayBirthNull = true;
            }
        }
        else if(monthBirth == 2){//This states that if the month is not one of the above,
and if it is 2 that it will go into validation for the 2nd month.
            if(dayBirth >28){//This states that if the user enters a value greater than 28
then it will not be accepted.
                System.out.println("Im Sorry This Month Is Invalid (To High), Please
Try Again!");
                dayBirthNull = false;
            }
            else if(dayBirth <1){//This states that if the user enters a value less than 1, it
will not be accepted.
                System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
                dayBirthNull = false;
            }
            else{//If the data entered is valid then the loop will be broken.
                dayBirthNull = true;
            }
        }
    }
}
```

```

    }
    }
    else if(monthBirth == 4 || monthBirth == 6 || monthBirth == 9 || monthBirth
== 11){//This is for the months with 30 days, if none of the above is true this will
activate.
        if(dayBirth >30){//This states that if the user enters a value more than 30,
then it is not valid.
            System.out.println("Im Sorry This Month Is Invalid (To High), Please
Try Again!");
            dayBirthNull = false;
        }
        else if(dayBirth <1){//This states that if the user enters a value less than 1
then it is not valid.
            System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
            dayBirthNull = false;
        }
        else{//If the user enters data that is valid then the loop will be broken.
            dayBirthNull = true;
        }
    }
}
}
}
}
}
public void setYearBirth(){//This section of code is used to get input from the user
and store it in yearBirth. Also to return error messages where needed.
    boolean yearBirthNull = false;
    while(!yearBirthNull){
        yearBirth = TextIO.getInt();
        if(yearBirth >9999){//This states that if the user enters a value over 9999 that
it is not valid.
            System.out.println("Im Sorry This Year Is Invalid (To High), Please Try
Again!");
            yearBirthNull = false;
        }
        else if(yearBirth <1){//This states that if the user enters a value of less than 1,
that it is not valid.
            System.out.println("Im Sorry This Year Is Invalid (To Low), Please Try
Again!");
            yearBirthNull = false;
        }
        else{//If the user enters valid data then the loop will be broken.
            yearBirthNull = true;
        }
    }
}
}
}
}
}
public void setMonth(){//This section of code is used to get input from the user and
store it in month. Also to return error messages where needed.
    boolean monthNull = false;
    while(!monthNull){
        month = TextIO.getInt();

```

```

        if(month >12){//This states that if the month entered is over 12, it is invalid.
            System.out.println("Im Sorry This Month Is Invalid (To High), Please Try
Again!");
            monthNull = false;
        }
        else if(month <1){//This states that if the month entered is less than 1, it is
invalid.
            System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
            monthNull = false;
        }
        else{//If the data is valid then the loop will be broken.
            monthNull = true;
        }
    }
}

public void setDay(){//This section of code is used to get input from the user and
store it in day. Also to return error messages where needed.
    boolean dayNull = false;
    while(!dayNull){
        day = TextIO.getInt();
        if(month == 1 || month == 3 || month == 5 || month == 7 || month == 8 || month
== 10 || month == 12){//This section of code looks at all months with 31 days.
            if(month >31){//If the user inputs over 31 then it is an invalid entry.
                System.out.println("Im Sorry This Month Is Invalid (To High), Please
Try Again!");
                dayNull = false;
            }
            else if(day <1){//If the user inputs less than 1 then it is an invalid entry.
                System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
                dayNull = false;
            }
            else{//If the user inptus between 1 and 31 then the entry is valid and the
loop will break.
                dayNull = true;
            }
        }
        else if(month == 2){//If the month is not one of the above, and is the 2'nd then
this will validate month 2.
            if(day >28){//If the user enters a day over 28, then it is invalid.
                System.out.println("Im Sorry This Month Is Invalid (To High), Please
Try Again!");
                dayNull = false;
            }
            else if(day <1){//If the user enters a day less than 1, then it is invalid.
                System.out.println("Im Sorry This Month Is Invalid (To Low), Please Try
Again!");
                dayNull = false;
            }
        }
    }
}

```



```

    int dayTotal = 0; //This is used to represent the total number of days for the
end/today's date.
    int yearTotal = daysInMonth[1] + daysInMonth[2] + daysInMonth[3] +
daysInMonth[4] + daysInMonth[5] + daysInMonth[6] + daysInMonth[7] +
daysInMonth[8] + daysInMonth[9] + daysInMonth[10] + daysInMonth[11] +
daysInMonth[12]; //This int is used to calculate the total number of days in a year.
    int yearBirthCalc = 0; //This int will be used to store the total number of days
over a set number of years for the Birth/Start Date.
    int yearCalc = 0; //This is used to store the total number of days over a set
number of years for the End/Today's Date.
    int birthTotal = 0; //This int is used to hold the total number of days for
birth/start date.

    int leapYearTarget = yearBirth; //This is used in a for loop to set leapYearTarget
as = to yearBirth.
    int leapYearTarget2 = year; //This is used in a for loop to set leapYearTarget2
as = to year.

    for(int counter=leapYearTarget;counter<leapYearTarget2;counter++){//This for
loop will loop while ever counter is less than leapYearTarget2.
        leapYearTarget++; //This is used to add one year to the date each time, so that
eventually leapYearTarget and leapYearTarget2 will be of the same value.
        if(leapYearTarget % 4 == 0 && leapYearTarget % 100 != 0 ){ //This section
of code is used to calculate if the year is a leap year, to do this is first checks that the
year is divisible by 4, and then checks that it is not divisible by 100.
            dayTotal++; //If the above is true one day is added to dayTotal.
        }
        else if(leapYearTarget % 400 == 0){//This finds out if the year is divisible by
400, and if yes adds one to the dayTotal.
            dayTotal++;
        }
    }

    /*The following if statement works out how many days there are between the
beginning of the year, and the month the user enters.
    *or example if the user enters 12, all the days in months 1 - 11 will be added.
But not the days from month 12, as this has not yet passed.
    */

    if(monthBirth == 1){
        monthTotal = monthTotal;
    }
    else if(monthBirth == 2){
        monthTotal = monthTotal+daysInMonth[1];
    }
    else if(monthBirth == 3){
        monthTotal = monthTotal+daysInMonth[1]+daysInMonth[2];
    }
    else if(monthBirth == 4){

```

```

    monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3];
    }
    else if(monthBirth == 5){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4];
    }
    else if(monthBirth == 6){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5];
    }
    else if(monthBirth == 7){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6];
    }
    else if(monthBirth == 8){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6]+daysInMonth[7];
    }
    else if(monthBirth == 9){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8];
    }
    else if(monthBirth == 10){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMonth[9];
    }
    else if(monthBirth == 11){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMonth[9]+
daysInMonth[10];
    }
    else if(monthBirth == 12){
        monthTotal =
monthTotal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[4]+da
ysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMonth[9]+
daysInMonth[10]+daysInMonth[11]];
    }

```

//The following if statement works out how many days there are between the beginning of the year, and the month the user enters.

```

if(month == 1){
    monthTotalFinal = monthTotalFinal;

```

```
    }
    else if(month == 2){
        monthTotalFinal = monthTotalFinal+daysInMonth[1];
    }
    else if(month == 3){
        monthTotalFinal = monthTotalFinal+daysInMonth[1]+daysInMonth[2];
    }
    else if(month == 4){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3];
    }
    else if(month == 5){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4];
    }
    else if(month == 6){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5];
    }
    else if(month == 7){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6];
    }
    else if(month == 8){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6]+daysInMonth[7];
    }
    else if(month == 9){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8];
    }
    else if(month == 10){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMont
h[9];
    }
    else if(month == 11){
        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMont
h[9]+daysInMonth[10];
    }
    else if(month == 12){
```

```

        monthTotalFinal =
monthTotalFinal+daysInMonth[1]+daysInMonth[2]+daysInMonth[3]+daysInMonth[
4]+daysInMonth[5]+daysInMonth[6]+daysInMonth[7]+daysInMonth[8]+daysInMont
h[9]+daysInMonth[10]+daysInMonth[11];
    }

```

//The following code calculates how many days total thee are for yearBirth. It calculates the years by yearBirth*yearTotal so that it can work out the number of days total, and these can then be taken away from the following section to return the correct value it does the same with year/month and day.

```

    yearBirthCalc = yearBirth*yearTotal;//This returns the value of days between year
0 and the year entered.
    birthTotal = yearBirthCalc+monthTotal+dayBirth;//This returns the total number of
days including the above.

```

```

    yearCalc = year*yearTotal; //This returns the value of days between year 0 and the
year entered.

```

```

    dayTotal = dayTotal+day+monthTotalFinal+yearCalc-birthTotal;

```

```

    System.out.print("There Are "); //This section of code shows the user how many
days there are between the 2 dates.

```

```

    System.out.print(dayTotal);
    System.out.println(" Day's Between Those Dates");
}

```

```

public void output(){//This section of code prints out the dates the user has entered.

```

```

    System.out.print("Your Date Of Birth Is: ");
    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
    System.out.print("Today's Date Is: ");
    System.out.println(day+ "/" + month+ "/" + year);
}

```

```

    public void tomorrow(){//This section of code allows the user to add 1 day to the
date they wish.

```

```

        int select = 2; //This will store the choice the user picks.

```

```

        System.out.println("To Add 1 Day To Your Start Date Press 0");

```

```

        System.out.println("Else To Add 1 Day To Your End Date");

```

```

        while(select >1 || select <0){//This is a while loop, that will continue to loop
untill valid date is input.

```

```

            select = TextIO.getInt();//This lets ther user input their choice.

```

```

            if(select == 0){//If the user inputs 0, then this code will execute.

```

```

                if(dayBirth == daysInMonth[monthBirth] && monthBirth != 12){//this
states that if dayBirth is = to the number of days in monthBirth, and is not month 12,
that the next date will be the 1st of the next month.

```

```

                    monthBirth++;

```

```

                    dayBirth = 1;

```

```

                    System.out.print("Your New Birth/Start Date Is: ");

```

```

                    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);

```

```

                }

```

```

        else if(dayBirth == daysInMonth[monthBirth] && monthBirth == 12)
    {
        //This code will set the date to the 1st of the following year.
        monthBirth = 1;
        yearBirth++;
        dayBirth = 1;
        System.out.print("Your New Birth/Start Date Is: ");
        System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
    }
    else{
        dayBirth++; //Else the day will go up by 1.
        System.out.print("Your New Birth/Start Date Is: ");
        System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
    }
}
else if(select == 1){ //This code will execute if the user selects 1.
    if(day == daysInMonth[month] && month != 12){
        month++;
        dayBirth = 1;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
    else if(day == daysInMonth[month] && month == 12){
        month = 1;
        year++;
        dayBirth = 1;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
    else{
        day++;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
}
else{ //If the user inputs incorrect data then the loop will restart.
    System.out.println("Im Sorry, Please Enter Either 0 For Add, Or 1 For
Subtracto!");
}
}
}
}
public void yesterday(){ //This section of code allows the user to subtract 1 day
from their chosen date.

    int select = 2; //This is used to store the value the user inputs.

    System.out.println("To Subtract 1 Day From Your Start Date Press 0");
    System.out.println("Else To Subtract 1 Day From Your End Date Press 1");
    while(select > 1 || select < 0){ //This is a while loop, that will not break until one of
the 2 valid options have been input.

```

select = TextIO.getInt();//This allows the user to input which date they wish to add a day to.

if(select == 0){ //This will execute if the user inputs 0 for the beginning/birth date.

if(dayBirth == 1 && monthBirth != 1){//If the 1st of a month is input this section of code will set the date to the last of the previous month, unless it is the 1st of Jan.

```

    monthBirth--;
    dayBirth = daysInMonth[monthBirth];
    System.out.print("Your New Date Is: ");
    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
}

```

else if(dayBirth == 1 && monthBirth == 1){//If it is the 1st of jan, this code will execute setting the date correctly to the previous year.

```

    monthBirth = 12;
    yearBirth--;
    dayBirth = daysInMonth[monthBirth];
    System.out.print("Your New Date Is: ");
    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
}

```

else{//Else a day will be removed from the current date.

```

    dayBirth--;
    System.out.print("Your New Date Is: ");
    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
}

```

else if(select == 1){//If the user selects option 2 then this code will execute for the end/today's date.

```

    if(day == 1 && month !=1){
        month--;
        day = daysInMonth[month];
        System.out.print("Your New Date Is: ");
        System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
    }

```

```

    else if(day ==1 && month ==1){
        month = 12;
        year--;
        day = daysInMonth[month];
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }

```

```

    else{
        day--;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
}

```

else{//If the user inputs incorrect data then an error message will be returned and the loop will restart.

```

        System.out.println("Im Sorry, Please Enter Either 0 For Add, Or 1 For
        Subtracto!");
    }
}
}

```

public void monthAdd(){//This section of code allows the user to select which date they would like to add one month to.

```

    int select = 2; //This is used to store the data input by the user.

```

```

        System.out.println("To Add 1 Month To Your Start Date Press 0");
        System.out.println("Else To Add 1 Month To Your End Date");
        while(select >1 || select <0){//This is a while loop that will not break until valid
        data has been input.
            select = TextIO.getInt();//This allows the user to input their choice and the
            choice will be stored in select.
            if(select == 0){ //If the user chooses option 1 then this code will execute for
            the beginning/birth date.
                if(monthBirth == 12){//If the month is = to 12 then the year will go up by 1
                and the month will be set to 1.
                    yearBirth++;
                    monthBirth = 1;
                    System.out.print("Your New Birth/Start Date Is: ");
                    System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
                }
                else{ //Else if the days entered are = to the daysInMonth then the month will
                go up by one, and the days will be set to the end of the next month, E.G 1/31/2007
                becomes 2/28/2007
                    if(dayBirth == daysInMonth[monthBirth]){
                        monthBirth++;
                        dayBirth = daysInMonth[monthBirth];
                        System.out.print("Your New Birth/Start Date Is: ");
                        System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
                    }
                    else{//If none of the above are true then one month will be added to the
                    date.
                        monthBirth++;
                        System.out.print("Your New Birth/Start Date Is: ");
                        System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
                    }
                }
            }
        }
        else if(select == 1){//If the user selects this option, then it will execute for end/
        today's date
            if(month == 12){
                year++;
                month = 1;
                System.out.print("Your New Date Is: ");
                System.out.println(day+ "/" + month+ "/" + year);
            }
        }
    }
}

```

```

        else {
            if (day == daysInMonth[month]) {
                month++;
                day = daysInMonth[month];
                System.out.print("Your New Date Is: ");
                System.out.println(day + "/" + month + "/" + year);
            }
            else {
                month++;
                System.out.print("Your New Date Is: ");
                System.out.println(day + "/" + month + "/" + year);
            }
        }
    }
}
else {
    System.out.println("Im Sorry, Please Enter Either 0 For Add, Or 1 For
Subtract!");
}
}
}

public void monthSubtract(){

    int select = 2;

    System.out.println("To Subtract 1 Month From Your Start Date Press 0");
    System.out.println("Else To Subtract 1 Month From Your End Date");
    while(select > 1 || select < 0){
        select = TextIO.getInt();
        if(select == 0){
            if(monthBirth == 1){
                yearBirth--;
                monthBirth = 12;
                System.out.print("Your New Date Is: ");
                System.out.println(dayBirth + "/" + monthBirth + "/" + yearBirth);
            }
            else if(dayBirth == daysInMonth[monthBirth]){
                monthBirth--;
                dayBirth = daysInMonth[monthBirth];
                System.out.print("Your New Date Is: ");
                System.out.println(dayBirth + "/" + monthBirth + "/" + yearBirth);
            }
            else {
                monthBirth--;
                System.out.print("Your New Date Is: ");
                System.out.println(dayBirth + "/" + monthBirth + "/" + yearBirth);
            }
        }
    }
    else if(select == 1){
        if(month == 1){

```

```

        year--;
        month = 12;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
    else if(day == daysInMonth[month]){
        month--;
        day = daysInMonth[month];
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
    else{
        month--;
        System.out.print("Your New Date Is: ");
        System.out.println(day+ "/" + month+ "/" + year);
    }
}
else{
    System.out.println("Im Sorry, Please Enter Either 0 For Add, Or 1 For
Subtract!");
}
}
}
}
public void yearAdd(){//This section of code will add one year to the chosen date.

    int select = 2;//This deals with user input.

    System.out.println("To Add 1 Year To Your Start Date Press 0");
    System.out.println("Else To Add 1 Year To Your End Date");
    while(select >1 || select <0){//This is a while loop to ensure that invalid data is
not input.
        select = TextIO.getInt();//This allows the user to input their choice and will be
stored in select.
        if(select == 0){ //If the user inputs 0, then the following code for birth/start
date will execute. 1 year will be added to the date.
            yearBirth++;
            System.out.print("Your New Birth/Start Date Is: ");
            System.out.println(dayBirth+ "/" + monthBirth+ "/" + yearBirth);
        }
        else if(select == 1){//Else if they select 1, 1 year will be added to the
end/today's date.
            year++;
            System.out.print("Your New Date Is: ");
            System.out.println(day+ "/" + month+ "/" + year);
        }
        else{//Else and error message will be returned, and the loop will go round
again.
            System.out.println("Im Sorry, Please Enter Either 0 For Add, Or 1 For
Subtract!");
        }
    }
}

```



```
boolean useAgain = false;
while(!useAgain){//While ever useAgain = false, the loop will continue, once it
= true the programing will end.
```

```
Date date = new Date(1,1,1900,1,1,1900);//This creates an object new date,
and sets the ints initial values.
```

```
System.out.println("Please Input Your Month Of Birth (Starting Month) In
Format 1-12");//This Code asks the user to input their start/birth month, and will
allow them to do so.
```

```
date.setMonthBirth();//This calls set.MonthBirth().
```

```
System.out.println("Please Input Your Day Of Birth (Starting Day) In Format
1-31 (Dependant On Month)");//This Code asks the user to input their start/birth day,
and will allow them to do so.
```

```
date.setDayBirth();//This calls setDayBirth().
```

```
System.out.println("Please Input Your Year Of Birth (Starting Day), Between
1 And 9999");//This code asks the user to input their start/birth year, and will allow
them to do so.
```

```
date.setYearBirth();//This calls setYearBirth();
```

```
System.out.println("Please Input Today's Month (End Month) In Format
1-12");//This code asks the user to input their end/today's month, and will allow them
to do so.
```

```
date.setMonth();//This calls setMonth().
```

```
System.out.println("Please Input Today's Day (End Day) In Format 1-31
(Dependant On Month) ");//This code asks the user to input their end/today's day, and
allows them to do so.
```

```
date.setDay();//This calls setDay().
```

```
System.out.println("Please Input Today's Year(End Year), Between 1 And
9999");//This code asks the user to input their end/today's year, and allows them to do
so.
```

```
date.setYear();//this calls setYear().
```

```
date.math();//This calls math(), and will display the total number of days
between the two dates the user input.
```

```
date.output();//This shows the two dates the user input.
```

```
System.out.println("We Will Now Add 1 Day, And Subtract One Day, From
The Date You Choose!"); //This lets the user know what will happen next.
```

```
date.tomorrow(); //This allows the user to select which date they wish to add
1 day to, and adds 1 day.
```

```
date.yesterday(); //This allows the user to select which date they wish to
subtract 1 day from, and subtracts 1 day.
```

```
System.out.println("We Will Now Add 1 Month, And Subtract One Month,
From The Date You Choose!"); //This shows the user what the program will do next.
```

```
date.monthAdd(); //This allows the user to select which date they wish to add
one month to, and adds one month.
```

```
date.monthSubtract();//This allows the user to select which date they wish to
subtract one month from, and subtracts one month.
```

```
System.out.println("We Will Now Add 1 Year, And Subtract One Year, From
The Date You Choose!"); //This informs the user what will happen next.
```

```
date.yearAdd(); //This allows the user to select which date they wish to add
one year to, and adds one year.
```

```
date.yearSubtract(); //This allows the user to select which date they wish to
subtract one year from, and subtracts one year.
```

```
int reUse = 2; //This sets reUse, which will be used to store the users input.
```

```
while(reUse >1 ^ reUse <0){ //This is a while loop to ensure that only valid
data is input.
```

```
System.out.println("Would You Like To Use This Program Again? Press 1
For Yes Or 0 For No!");
```

```
reUse = TextIO.getInt(); //This allows the user to input their choice, and
stores it in reUse.
```

```
if(reUse == 1){ //If the user inputs 1, then the program will loop to the start.
useAgain = false;
```

```
}
else if(reUse == 0){ //If the user inputs 0, then the program will end.
useAgain = true;
```

```
}
else{ //Else and error message will be returned, and the loop will not break.
System.out.println("Im Sorry, Please Enter Either 1 For Yes, Or 0 For
```

```
No!");
```

```
}
```

```
}
```

```
}
```

```
System.out.println("Thank You For Using My Program! Copyright Jason Gunn
2007 - Contact Details: Jasongunn99@hotmail.com And
06039966@students.lincoln.ac.uk"); //This is displayed at the end of the program.
```

```
}
```

```
}
```

Conclusion

I found this assignment quite challenging, however I feel that my previous programming, as well as the workshops has helped greatly, and while I am far from good I feel my programming has improved thanks to the challenge this assignment posed.

I have attempted to create a complete program, that meets all the needs of the assignment brief, that is secure, and as tidy as possible. I admit there are still several areas of which could be improved, however I feel that this is something I will learn with time.

I have learnt several new programming aspects from this unit so far, and from researching for assignment 1.

Bibliography

Sources Used:

Lincoln Virtual Campus

Resources Used:

TextIO.Java - <http://www.faqs.org/docs/javap/source/TextIO.java>