

AP Computer Science A Summer Assignment

You have made an awesome decision for your future to take this class during the 2019-2020 school year. This assignment is meant to keep your Java knowledge fresh over the summer in preparation. In it, you will create four methods. They are listed below in order of increasing difficulty. I would suggest starting them by **August 1**. This will give you more than enough time to complete them even working roughly **1 hour per week**.

The directions below assume you will complete this assignment in Java. If you are unfamiliar with Java, be aware that the entirety of this course is Java-based. If you are familiar with a different language, that is an acceptable starting point, and you may adapt the directions below accordingly to work for that language.

You will submit an assignment called SummerAssignment*.java to me when school begins, where the * is replaced by your last name. Above all else, I am looking for effort here. On the following page, I have included some method calls you can make from a main method to check your work.

Method 1: Write a `static` method that returns an `int` and is called `countAbsences`. This method should accept an `int` array as a parameter that represents the number of class periods a student has attended on each day of a week. It should count how many of the days the student was considered absent, and return that number. For this method, an absence is a day when a student has attended less than 4 class periods.

Method 2: Write a `static` method that returns a `boolean` variable and is called `plagiarism`. This method should accept two `String` parameters, the first representing a student text submission, and the second a phrase to search for. The method should return `true` if the submission contains the phrase, and `false` otherwise. For a small extra challenge and a better-overall method, write the method so that capitalization does not matter.

Method 3: Write a `static` method that returns a `boolean` and is called `isLeapYear`. This method should accept an `int` as a parameter that represents a year. The method should return `true` if the year is a leap year, and `false` otherwise. A leap year is any year divisible by 4, except years divisible by 100 but not 400.

Method 4: Write a `static` method that returns a `String` and is called `getDate`. This method should accept two `int` parameters, the first representing how many days into a year we are, and the second representing a year. The method should return the date in the format `month/day/year`. To complete this, you will have to figure out how to translate the “days into a year” into a month and date—for example, the 35th day of the year would be 2/4 (February 4th). Please note that this can change slightly depending on whether or not the year parameter is a leap year. For example, day 365 of 2004 is 12/30/2004, but day 365 of 2005 is 12/31/2005. Additionally, I want the program to return error messages if invalid values are passed. If there are less days in the calendar year than the first parameter, the program should return:

ERROR: Invalid number of days.

Days Beyond 1 Year: (insert how many days over you went here)

If the user enters a number less than 1, the program should return:

ERROR: Invalid number of days.

Day Number: (whatever they entered for the first parameter)

Year: (whatever they entered for the second parameter)

You can add the following lines of code into a main method, and should get the output that follows:

```
System.out.println("1999 is a leap year: " + isLeapYear(1999));
System.out.println("2000 is a leap year: " + isLeapYear(2000));
System.out.println("2004 is a leap year: " + isLeapYear(2004));
System.out.println("2100 is a leap year: " + isLeapYear(2100));
```

```
System.out.println("\n");
```

```
System.out.println("Day 247 of 2018: " + getDate(247, 2018));
System.out.println("Day 365 of 2018: " + getDate(365, 2018));
System.out.println("Day 365 of 2020: " + getDate(365, 2020));
System.out.println("Day 366 of 2020: " + getDate(366, 2020));
```

```
System.out.println();
```

```
System.out.println(getDate(0, 2018));
System.out.println(getDate(366, 2018));
```

```
System.out.println("\n");
```

```
System.out.println("Joke 1 is Stolen: " + plagiarism("There are 10
types of people; those who understand binary, and those who don't.",
"those who understand binary"));
```

```
System.out.println("Joke 2 is Stolen: " + plagiarism("Binary is as
easy as 01-10-11.", "it's as easy"));
```

```
System.out.println("Joke 3 is Stolen: " + plagiarism("The semicolon:
Hide and seek champion since 1958.", "champion"));
```

```
System.out.println("\n");
```

```
int[] student1 = {3, 7, 7, 4, 7};
int[] student2 = {7, 7, 7, 7, 7};
int[] student3 = {0, 0, 3, 0, 0};
```

```
System.out.println("Student 1 Absences: " + countAbsences(student1));
System.out.println("Student 2 Absences: " + countAbsences(student2));
System.out.println("Student 3 Absences: " + countAbsences(student3));
```

Your output should be the following:

1999 is a leap year: false
2000 is a leap year: true
2004 is a leap year: true
2100 is a leap year: false

Day 247 of 2018: 9/4/2018
Day 365 of 2018: 12/31/2018
Day 365 of 2020: 12/30/2020
Day 366 of 2020: 12/31/2020

ERROR: Invalid Number of days.
Day Number: 0
Year: 2018
ERROR: Invalid number of days.
Days Beyond 1 Year: 1

Joke 1 is Stolen: true
Joke 2 is Stolen: false
Joke 3 is Stolen: true

Student 1 Absences: 1
Student 2 Absences: 0
Student 3 Absences: 5