

Lab 3.3 - Let Me Check My Calendar

In this lab, you will write some custom blocks that take arguments and are useful for calculations involving dates and calendars.

1 - Basics

1. Write a custom SNAP block called "month name" that takes a number between 1 and 12 as an argument and says the name of the corresponding month.
2. Write a custom SNAP block called "day name" that takes a number between 1 and 7 as an argument and says the name of the corresponding day. For our purposes, the week begins on Sunday.
3. Write a custom SNAP block called "days in " that takes a month name as an argument and says how many days are in that month. Assume a non-leap year.

2 - Going Farther

1. Write a custom SNAP block called "is a leap year" that takes a year number as an argument and says whether or not that year is a leap year.
 - A year is a leap year if the year is a multiple of 4 that is not a multiple of 100 (e.g. 1984), or if it is a multiple of 400 (e.g. 2000). Years that are multiples of 100 but not multiples of 400 are NOT leap years (e.g. 1800). See [Wikipedia](https://en.wikipedia.org/wiki/Leap_year) for more detail.
2. Write a custom SNAP block called "is a valid date" that takes a month name and a date as arguments and says whether or not that date exists in that month. For example, the 31st is a valid date in January, but not in June. The 5th is a valid date in every month, and the 40th is not a valid date in any month.
3. Write a custom SNAP block called "day in year" that takes a year number and a number between 1 and 366 and says the date that corresponds to that numbered day of the specified year. For example, in non-leap years day #1 is January 1, day #32 is February 1, day #365 is December 31, and day #185 is July 4. Give an error message if the number is 366 and a non-leap year is specified.
4. BONUS: Write a custom SNAP block called "day of week" that takes a month name, date, and year as arguments and says the day of week on which that date falls in that year.
See http://en.wikipedia.org/wiki/Determination_of_the_day_of_the_week for information on finding the day of the week from a date.



Lab 3.3 Criteria	
1.1 month name	0.5 points
1.2 day name	0.5 points
1.3 days in	0.5 points
2.1 is leap year	0.5 points
2.2 is a valid date	0.5 points
2.3 day in year	0.5 points
2.4 Bonus: day in week	0.5 points
PROJECT TOTAL	3.5 points

Big Ideas**Products can be designed for life cycle**

Think carefully about how these custom blocks could be used in other programs, or how they could be used if you had to expand this current program. There are quite a lot of uses for custom blocks that format the days, months and years. The following software applications might all require these types of custom blocks:

- calendar software
- billing and payments software
- alarms and notification software
- recording of events and data software

In addition, if any updates had to be made to existing programs then it would be much easier to make these updates if code is contained in custom blocks.

