CalendarInfo Class

Attributes:

- **DAYS_OF_WEEK**
  (list) Holds the name for the days of the weeks
  e.g. 'Su', 'Mo'
- **calendar_dict**
  (dictionary) Stores and updates the information for every month as well as the holidays for each month.

UserCalendar Class

Attributes:

- **Int_year**
  (int) holds the desired year as an integer(must be four digits)
  e.g. 2022, 1990, 2500
- **str_year**
  (str) holds the desired year as an integer from the int_year
  e.g. 2022, 1990, 2500
- **first_day_of_month**
  (int) Holds the first day of the month for every year and with this the name for day of the week is determined.
- **disp_months_2d**
  (2d-list) Horizontal list: holds 12 lists for each month
  Vertical side: holds everyday for each month

Methods:

- **opens_file()**
  opens calendar file in local folder
- **rand_year(upper_bound_year, lower_bound_year)**
  Parameter upper_bound_year, lower_bound_year(int)
  Generates a random year depending on what bounds the user entered
- **user_year()**
  returns the user's choice on the year in both int and string in that order
- **days_in_feb(int_year)**
  Determines how many days there will be on Febuary depending on the year
- **start_day_formula()**
  returns what day of the week the year began with a formula
- **print_leap_year(first_day_of_month)**
  Parameter first_day_of_month(int) name of weekday
  Prints out if the year is a leap year and if it is the variable first_day_of_month is changed
- **create_2d_list()**
  Creates a list and appends more lists on to creata a 2d list
- **append_holiday()**
  adds more holidays for the desired month in the calendar_dict
- **insert_zeros(month_count)**
  Parameter month_count(int)

Determines the first day of the week for the new month and inserts zeroes as place holder

- **close_file()**
  Closes the curent file to reduces the risk of being unwarrantedly modified or read

WriteCalendar Class

Methods:

- **write_year()**
  Writes the year on the calendar.txt file
- **write_months(month_name)**
  Parameter month_count(int)
  Writes the name of the month on the txt file.
- **write_week()**
  writes and Organizes the days of the week for every month
- **write_days(month_name, month_count)**
  Parameter month_name(str), month_count(int)
  Writes the days for the current month by casting the days to string. The zeros will then be converted to blank strings, “”.
- **write_dash()**
  writes the dashes for under the days of the week
- **write_holidays(month_name)**
  Parameter month_name(str)
  reads from the calendar.py to write the holiday on the calendar.txt file.
- **write_calendar()**
  Writes and organizes the year, month, weeks, and days as an organized calendar by using the previous methods