Michael Richards

2/25/19

CS 140

Assignment 2

Program

//Student: Michael Richards  
//Course: CS 140C  
//Date: 2/15/19  
//Project: Assignment Part 2  
//  
//This program will produce a calendar of a specific month.  
  
import java.time.\*;  
import java.util.\*;  
  
public class MyCalendar2 {  
  
 public static int *NUM\_MONTH* = 0;  
 public static int *NUM\_DAY* = 0;  
 public static int *NUM\_YEAR* = 0;  
  
 public static void main(String[] args) { //Main method to call everything  
 *menu*();  
 }  
  
 /\* Methods for the menu \*/  
  
 public static void menu() {  
 System.*out*.println("Please type a command.");  
 System.*out*.println(" \"e\" to enter a date and display the corresponding calendar");  
 System.*out*.println(" \"t\" to get today's date and display today's calendar");  
 System.*out*.println(" \"n\" to display the next month");  
 System.*out*.println(" \"p\" to display the previous month");  
 System.*out*.println(" \"q\" to quit the program");  
 Scanner console = new Scanner(System.*in*);  
 String userIn = console.next();  
 *whatToDo*(userIn);  
 }  
  
 public static void whatToDo(String userIn) {  
 if (userIn.toLowerCase().equals("e")) {  
 // Enter date and display  
 *enterDate*();  
 *menu*();  
 } else if (userIn.toLowerCase().equals("t")) {  
 // Today's date and calendar  
 *todaysDate*();  
 *menu*();  
 } else if (userIn.toLowerCase().equals("n")) {  
 // Next month  
 *nextMonth*();  
 *menu*();  
 } else if (userIn.toLowerCase().equals("p")) {  
 // Previous month  
 *previousMonth*();  
 *menu*();  
 } else if (userIn.toLowerCase().equals("q")) {  
 // Quit  
 System.*out*.println("Sorry to see you go. Come back soon!");  
 System.*exit*(0);  
 } else {  
 System.*out*.println();  
 System.*out*.println("Invalid input. Please try again.");  
 *menu*();  
 }  
 }  
  
 // Method for "e"  
 public static void enterDate() {  
 Scanner console = new Scanner(System.*in*);  
  
 System.*out*.println("What date would you like to look at? MM/DD/YYYY");  
 String userInput = console.next();  
 *drawCalendar*(userInput);  
 }  
  
 // Method for "t"  
 public static void todaysDate() {  
 Date today = new Date();  
 Calendar c = Calendar.*getInstance*();  
 c.setTime(today);  
  
 int month = c.get(Calendar.*MONTH*) + 1;  
 int day = c.get(Calendar.*DAY\_OF\_MONTH*);  
 int year = c.get(Calendar.*YEAR*);  
 String out = month + "/" + day + "/" + year;  
  
 System.*out*.println();  
 System.*out*.println("Today's Date:");  
 *drawCalendar*(out);  
 }  
  
 // Method for "n"  
 public static void nextMonth() {  
 if (*NUM\_DAY* == 0) {  
 System.*out*.println("You need to select a month with \"e\" or \"t\"");  
 *menu*();  
 } else {  
 int month = *NUM\_MONTH* + 1;  
 int day = *NUM\_DAY*;  
 int year = *NUM\_YEAR*;  
 if (month == 13) {  
 month = 1;  
 year = *NUM\_YEAR* + 1;  
 }  
 String out = month + "/" + day + "/" + year;  
  
 System.*out*.println();  
 System.*out*.println("Next Month's Date:");  
 *drawCalendar*(out);  
 }  
 }  
  
 // Method for "p"  
 public static void previousMonth() {  
 if (*NUM\_DAY* == 0) {  
 System.*out*.println("You need to select a month with \"e\" or \"t\"");  
 *menu*();  
 } else {  
 int month = *NUM\_MONTH* - 1;  
 int day = *NUM\_DAY*;  
 int year = *NUM\_YEAR*;  
 if (month == 0) {  
 month = 12;  
 year = *NUM\_YEAR* - 1;  
 }  
 String out = month + "/" + day + "/" + year;  
  
 System.*out*.println();  
 System.*out*.println("Last Month's Date:");  
 *drawCalendar*(out);  
 }  
 }  
  
 /\* Methods to draw the calendar \*/  
  
 // Method to draw the calendar  
 public static void drawCalendar(String userInput) { //Draws the calendar  
 *drawArt*();  
 int date = 1;  
 int end = userInput.length();  
 int mid = userInput.indexOf("/");  
 int mid2 = userInput.lastIndexOf("/");  
  
 *NUM\_MONTH* = Integer.*valueOf*(userInput.substring(0, mid));  
 String month = userInput.substring(0, mid);  
 *NUM\_DAY* = Integer.*valueOf*(userInput.substring(mid + 1, mid2));  
 String day = userInput.substring(mid + 1, mid2);  
 *NUM\_YEAR* = Integer.*valueOf*(userInput.substring(mid2 + 1, end));  
 String year = userInput.substring(mid2 + 1, end);  
  
 int month2 = Integer.*parseInt*(month);  
 int monthNum = *findMonth*(month2);  
 for (int i = 0; i < 35; i++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.println(monthNum);  
 *drawLine*();  
 System.*out*.println("| Sun | Mon | Tues | Wed | Thurs | Fri | Sat |");  
  
 int weekDay = *findWeek*(month2);  
 date -= weekDay;  
  
 for (int i = 0; i < 6; i++) { //Number of rows  
 *drawLine*();  
 date = *drawRow*(day, monthNum + 1, date);  
 }  
 *drawLine*();  
  
 System.*out*.println("Year: " + year);  
 System.*out*.println("Month: " + month);  
 System.*out*.println("Day: " + day);  
 }  
  
 // Method to draw ASCII art  
 public static void drawArt() {  
 for (int lead = 0; lead < (10 \* 7 / 2) - 20; lead++) {  
 System.*out*.print(" ");  
 }  
 for (int i = 0; i < 5; i++) {  
 for (int j = 0; j < 5; j++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("/\\");  
 }  
 System.*out*.println();  
 for (int lead = 0; lead < (10 \* 7 / 2) - 20; lead++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print(" ");  
 for (int i = 0; i < 5; i++) {  
 for (int j = 0; j < 3; j++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("/");  
 for (int j = 0; j < 2; j++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("\\");  
 }  
 System.*out*.println();  
 for (int lead = 0; lead < (10 \* 7 / 2) - 20; lead++) {  
 System.*out*.print(" ");  
 }  
 for (int j = 0; j < 2; j++) {  
 System.*out*.print(" ");  
 }  
 for (int i = 0; i < 5; i++) {  
 System.*out*.print(" ");  
 for (int j = 0; j < 3; j++) {  
 System.*out*.print("/\\");  
 }  
 }  
 System.*out*.println();  
 for (int lead = 0; lead < (10 \* 7 / 2) - 20; lead++) {  
 System.*out*.print(" ");  
 }  
 for (int i = 0; i < 2; i++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("/");  
 for (int i = 0; i < 5; i++) {  
 for (int j = 0; j < 6; j++) {  
 System.*out*.print(" ");  
 }  
 System.*out*.print("\\");  
 }  
 System.*out*.println();  
 }  
  
 // Method to draw the line in the calendar  
 public static void drawLine() { //Draws the line  
 int length = 10 \* 7;  
 for (int i = 0; i < length; i++) {  
 System.*out*.print("=");  
 }  
 System.*out*.println();  
 }  
  
 // Finds the month  
 public static int findMonth(int monthNum) { //Puts the month number with the month  
 int month = Calendar.*JANUARY*;  
 if (monthNum == 2) {  
 month = Calendar.*FEBRUARY*;  
 } else if (monthNum == 3) {  
 month = Calendar.*MARCH*;  
 } else if (monthNum == 4) {  
 month = Calendar.*APRIL*;  
 } else if (monthNum == 5) {  
 month = Calendar.*MAY*;  
 } else if (monthNum == 6) {  
 month = Calendar.*JUNE*;  
 } else if (monthNum == 7) {  
 month = Calendar.*JULY*;  
 } else if (monthNum == 8) {  
 month = Calendar.*AUGUST*;  
 } else if (monthNum == 9) {  
 month = Calendar.*SEPTEMBER*;  
 } else if (monthNum == 10) {  
 month = Calendar.*OCTOBER*;  
 } else if (monthNum == 11) {  
 month = Calendar.*NOVEMBER*;  
 } else if (monthNum == 12) {  
 month = Calendar.*DECEMBER*;  
 }  
 return month;  
 }  
  
 // Finds the day of the week  
 public static int findWeek(int monthNum) {  
 int weekDay = 7;  
  
 LocalDate localDate = LocalDate.*of*(2019, monthNum, 1);  
 java.time.DayOfWeek dayOfWeek = localDate.getDayOfWeek();  
  
 if (dayOfWeek == DayOfWeek.*MONDAY*) {  
 weekDay = 1;  
 } else if (dayOfWeek == DayOfWeek.*TUESDAY*) {  
 weekDay = 2;  
 } else if (dayOfWeek == DayOfWeek.*WEDNESDAY*) {  
 weekDay = 3;  
 } else if (dayOfWeek == DayOfWeek.*THURSDAY*) {  
 weekDay = 4;  
 } else if (dayOfWeek == DayOfWeek.*FRIDAY*) {  
 weekDay = 5;  
 } else if (dayOfWeek == DayOfWeek.*SATURDAY*) {  
 weekDay = 0;  
 } else if (dayOfWeek == DayOfWeek.*SUNDAY*) {  
 weekDay = 6;  
 }  
  
 return weekDay;  
 }  
  
 public static int drawRow(String day, int monthNum, int date) { //Draws the rows  
 int height = 5;  
 int day2 = Integer.*parseInt*(day);  
 int month = *findMonth*(monthNum);  
  
 Calendar c = new GregorianCalendar(*NUM\_YEAR*, month, day2);  
 int totalDays = (c.getActualMaximum(Calendar.*DAY\_OF\_MONTH*)) + 1;  
  
 for (int i = 1; i <= height; i++) { //Height of day box  
 for (int j = 0; j < 8; j++) { //Number of lines in a row  
 int width;  
 if (date > 0) {  
 width = 10 - (int) (Math.*log10*(date) + 2); //Finds the new width with day number  
 } else {  
 width = 9;  
 }  
 System.*out*.print("|");  
 if (i == 1 && j != 7 && date == day2) { //Bolds the date selected or today's date  
 System.*out*.print("\033[1m" + date + "\033[0m");  
 date++;  
 } else if (i == 1 && j != 7 && date <= totalDays && date > 0) {  
 System.*out*.print(date);  
 date++;  
 } else if (i == 1 && j != 7 && date <= totalDays) {  
 date++;  
 }  
 if (i == 1 && date <= totalDays) {  
 for (int k = 0; k < width; k++) {  
 System.*out*.print(" ");  
 }  
 } else {  
 for (int k = 0; k < 9; k++) {  
 System.*out*.print(" ");  
 }  
 }  
 if (date == totalDays) {  
 date = date + 100;  
 }  
 }  
 System.*out*.println();  
 }  
 return date;  
 }  
}