**DATE**

**PROGRAMS**

PROGRAM 1

Write a program in java to accept year, month, and the weekday’s name of the 1st day of that month. Generate its calendar in calendar format week starting with Monday.

Sample input:

Enter a year: 2017

Enter a month: June

Enter 1st day of the month: Thursday

Sample output:

June 2017

M T W TH F S S

1 2 3 4

5 6 7 8 9 10 11

12 13 14 15 16 17 18

19 20 21 22 23 24 25

26 27 28 29 30

46

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: The names of the weekdays are kept in array weekdays[]

Step 4: The total days of the month of a leap year are kept in array

lmonth[]

Step 5: The total days of the month of a normal year are kept in

array month[]

Step 6: The names of the months are kept in array m[]

Step 7: The year, month, and the weekday’s name of the 1st day of

that month is accepted from the user in variable y, m1, dm

respectively

Step 8: an array a[][] is initialized to build the number of rows and

Columns of the calendar

Step 9: initialize variable c=1

Step 10: execute the following steps from 1 to 7 using variable i

Step 10.1: if the weekday’s name matches with the accepted weekday then initialize k=i

Step 11: Run a loop i from 1 to 12 executing the following steps

Step 11.1: Check if the accepted month is equal to m[i] or not. Then check if the year is leap year or not. when it matches initialize lmonth[i] to nod variable otherwise initialize month[i] to nod variable

Step 11.2: initialize value of k as k-1

Step 12: Run a loop i from 0 to 6 and an inner loop j from value of k to 7 executing the following steps

Step 12.1: initialize the value of c in a[i][j] based on ith value and jth Value and increment the value of c by 1

Step 12.2: if the cth value is greater than nod then break statement is executed. The inner loop stops and initialize k=0

Step 13: Run the loop i from 0 to 6 and inner loop j from 0 to 7 executing the following steps

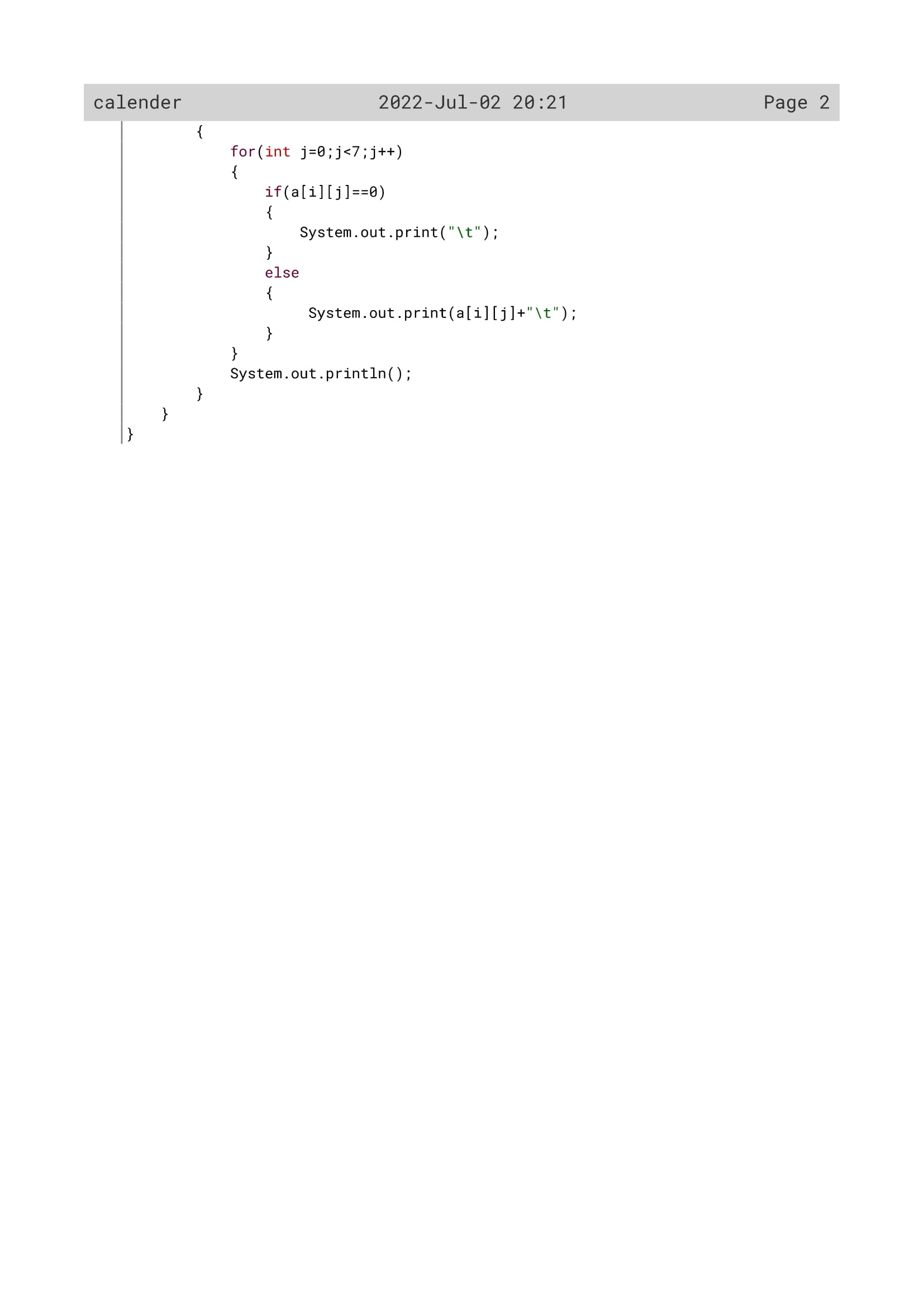
Step 13.1: if a[i][j]=0 then print space else print the value of a[i][j] and display the calendar

Step 14: Stop.

47

Source Code

48



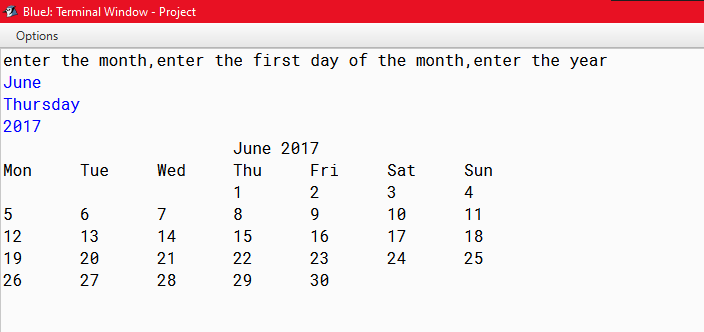
49

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | Description |
| weekdays[] | String | Array to store all the all the week names |
| lmonth[] | Int | keeping total days in a month in an array of a leap year |
| month[] | Int | keeping total days in a month in an array of a normal year |
| m[] | String | keeping the name of the months in an array |
| m1 | String | asking the user to input the month |
| dm | String | asking the user to input the first day of the month |
| y | Int | asking the user to input the year |
| k | Int | Flag variable |
| nod | Int | Flag variable |
| c | Int | Flag variable |
| i | int | Loop variable |
| a[][] | Int | making an array to build the rows and columns of a calendar |

50

Output



51

PROGRAM 2

Write a program in java to input a date and check whether it is a palindromic date.

Sample input:

Enter a date: 10/02/2001

Sample output:

The date is a palindromic date

52

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: The date is accepted from the user in variable str

Step 4: use substring function to separate the day number, monthAnd the year stored in variable d, m, y and convert it to

int type in variable date, month, year

Step 5: The total days of the month of a normal year are kept in Array nod[]

Step 6: concatenate d, m, y and stored in mq variable

Step 7: convert mq to int type in variable pd

Step 8: initialize k=pd for storing the date for later checking if it is a palindrome or not

Step 9: Now check whether the date is valid or not

Step 10: run a while loop until pd is equal to 0 and execute the following steps

Step 10.1: extract the remainder and reverse the number pd by adding it to rev variable

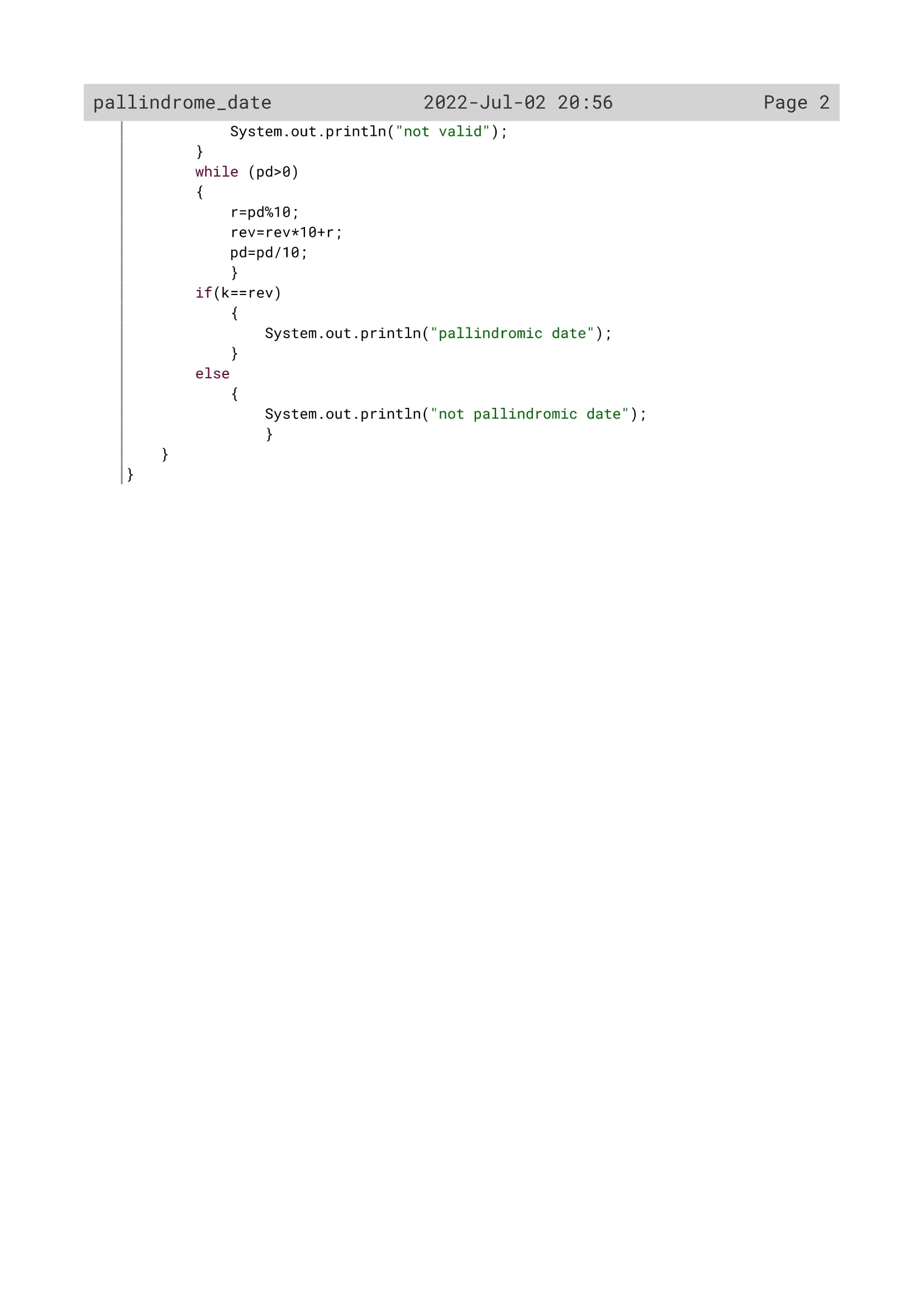
Step 10.2: check if k is equal to rev or not. If it is equal print the number is palindrome date else print not palindrome date

Step 11: Stop

53

Source Code

54



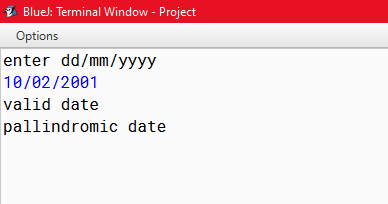
55

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | Description |
| str | String | To accept the date by the user |
| pd | String | converting the date from string to int data type |
| mq | String | concatenating the date, month and year from finding the pallindrome number |
| y | String | To extract the year |
| m | String | To extract the month |
| d | String | To extract the day |
| fd | Int | Flag variable |
| fm | Int | Flag variable |
| r | Int | To find the remainder |
| rev | Int | finding the pallindrome of the date by reversing it |
| nod[] | Int | keeping total days in a month in an array of a normal year |
| year | Int | To convert the extracted year from string to int data type |
| date | Int | To convert the extracted date from string to int data type |
| month | int | To convert the extracted month from string to int data type |

56

OUTPUT



57

PROGRAM 3

Design a program which accepts a date of birth in dd/mm/yyyy format. Check whether the date is a valid date or not. Also display the birth number i.e., 1,2,3,4,5,6,7,8,9,11

Sample Input:

Enter a date: 25/03/2005

Sample Output:

25/03/2005 is a valid date.

The birth number is 8 ( 2+5+0+3+2+0+0+5 = 17, 1+7=8)

58

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: The date is accepted from the user in variable str

Step 4: use substring function to separate the day number, month And the year stored in variable d, m, y and convert it to int type in variable date, month, year

Step 5: concatenate d, m, y and stored in mq variable

Step 6: convert mq to int type in variable mg

Step 7: The total days in a month are kept in an array nod[] of a normal year

Step 8: check if the year is leap year or not

Step 8.1: check if the month is 2 or not, the day is between 1 to 29 or not. If the three conditions satisfy then initialize fd=1;

Step 8.2: check if the date is between 1 and the last date of the Month, nod[month]. If this satisfies then initialize fd=1

Step 9: If the year is not a leap year then check if the date is between 1 and the last date of the Month, nod[month]. If this satisfies then initialize fd=1.

Step 9.1: now check if the month if between 1 to 12 or not. If it is then initialize fm as 1

Step 10: now check if fd=1 and fm=1.If they are then print the date as valid date else print not valid.

Step 11: Inorder to display the birth number pass the mg value to Sumofdgt function until the value of mg is single digit.

Step 12: Store the returned number in x and again reinitialize mg=x

Step 13: Print the birth number

**Algorithm for sumofdigit ()**

Step-1: Start

Step-2: until the number is equal to 0, execute the following steps.

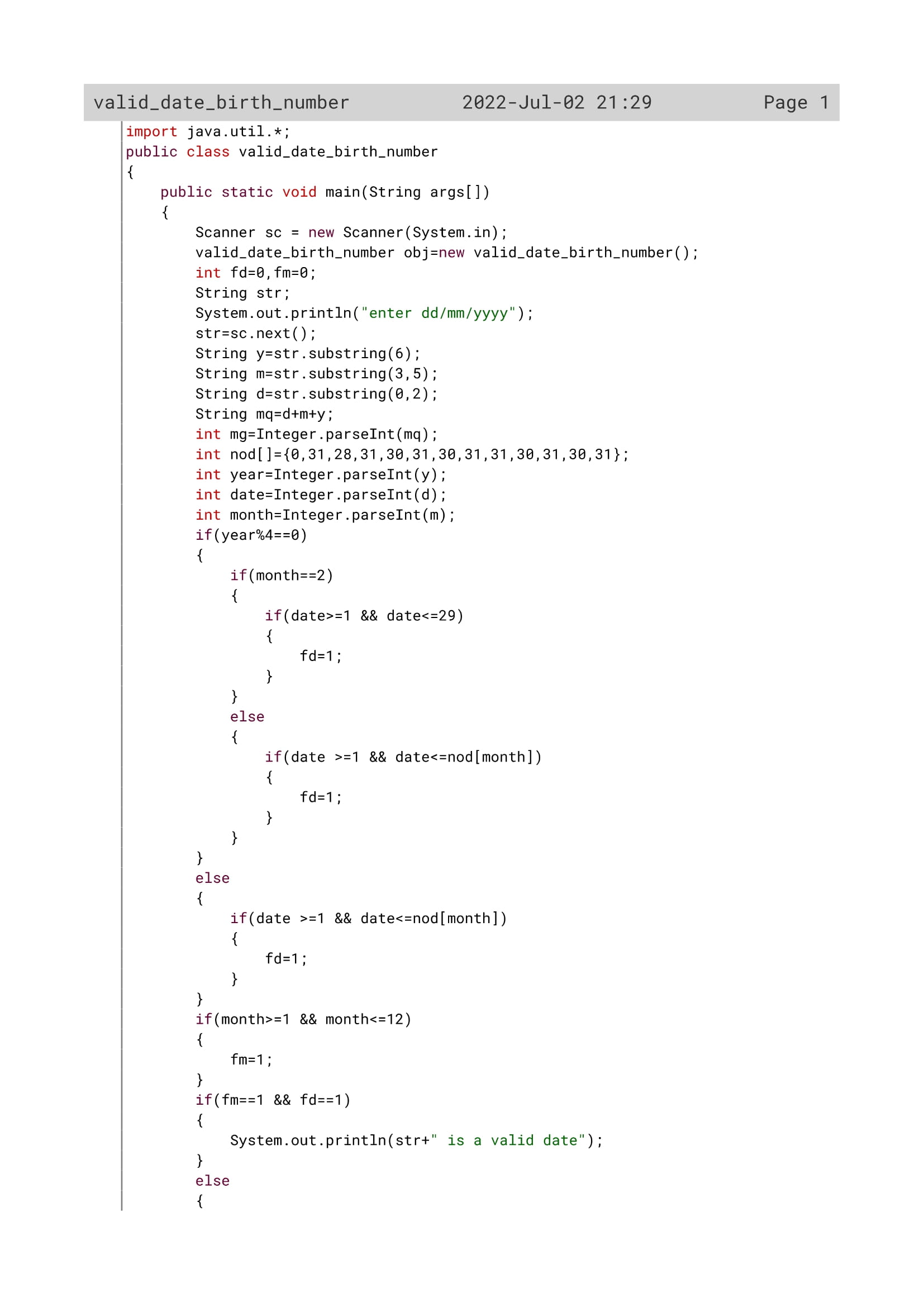
Step-2.1: extract each number and store it to r.

Step-2.2: find the summation by adding r to s variable.

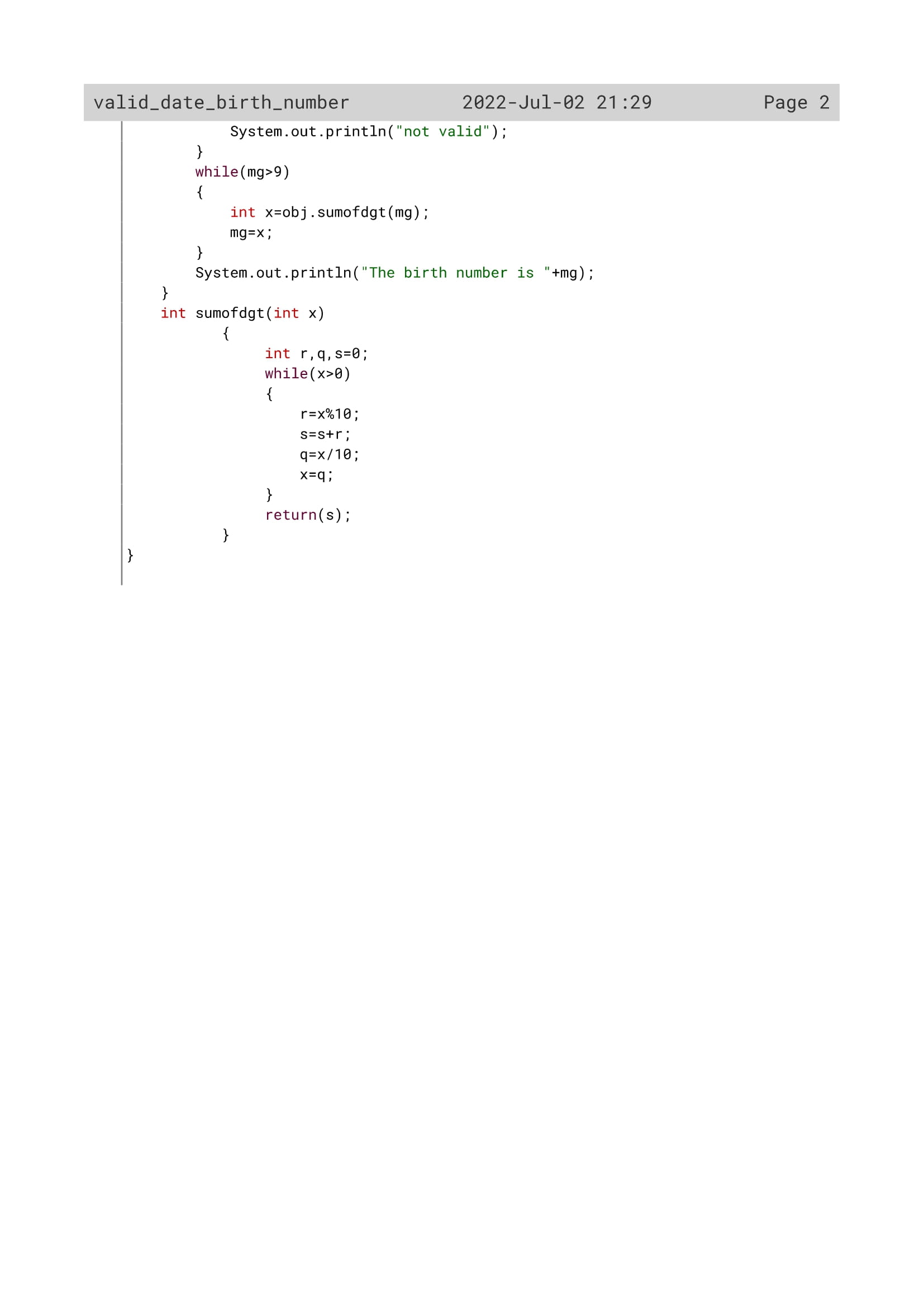
Step-2.3: find the quotient by x/10 operation.

Step-3: Return the variable s as sum of the Number.

Step-4: stop 59

Source Code

60



61

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | Description |
| str | String | accepting the date from the user |
| y | String | extracting the year |
| m | String | extracting the month |
| d | String | extracting the date |
| fd | Int | Flag variable flag variable |
| fm | Int | flag variable |
| nod[] | Int | keeping total days in a month in an array of a normal year |
| year | Int | To convert the extracted year from string to int data type |
| date | Int | To convert the extracted date from string to int data type |
| month | Int | To convert the extracted month from string to int data type |
| mg | Int | For converting date from string to int |
| x | Int | To store sum of digit value |
| r | Int | To store remainder value |
| s | Int | To store sum value |

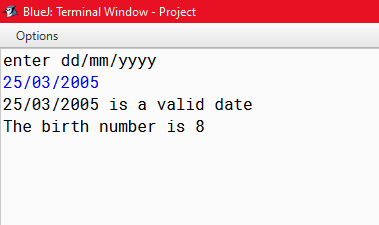
62

Function Description

|  |  |  |
| --- | --- | --- |
| Function name | Data type | description |
| main() | void | To accept a date and check whether it is a valid date or not and find its birth number |
| sumofdgt() | Int | To calculate sum of digits |

63

OUTPUT



64

PROGRAM 4

Write a program in java to accept a day number (between 1 to 366), year (4 digits) to generate and display the corresponding date. The program further accepts ‘n’ (1<=n<=100) from the user to compute and display the future date corresponding to ‘n’ days after the generated date.

Sample input:

Day number: 233

Year: 2008

Days after (n): 17

Sample output:

20 August 2008

Days after 17(n) days: 6 September 2008

65

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: The year, number of days, number of days later is accepted from the user in variable year, day, d1 variable

Step 4: The names of the months are kept in array month[]

Step 5: The total days of the month of a leap year are kept in array nod2[] and the total days of the month of a normal year are kept in nod1[]

Step 6: Execute the following steps from 1 to 12

Step 6.1: check if the year is leap year or not. If it is then add the month’s total number of days nod2[i] to s variable else add the month’s total number of days nod1[i] to s variable and if s exceeds the variable day value then execute break statement

Step 7: perform s-nod[2] and store the value in s variable itself

Step 8: in d variable store the value of day-s and is day variable add the day variable value with d1

Step 9: Print the day month and year as the date.

Step 10: Again execute the following steps from 1 to 12

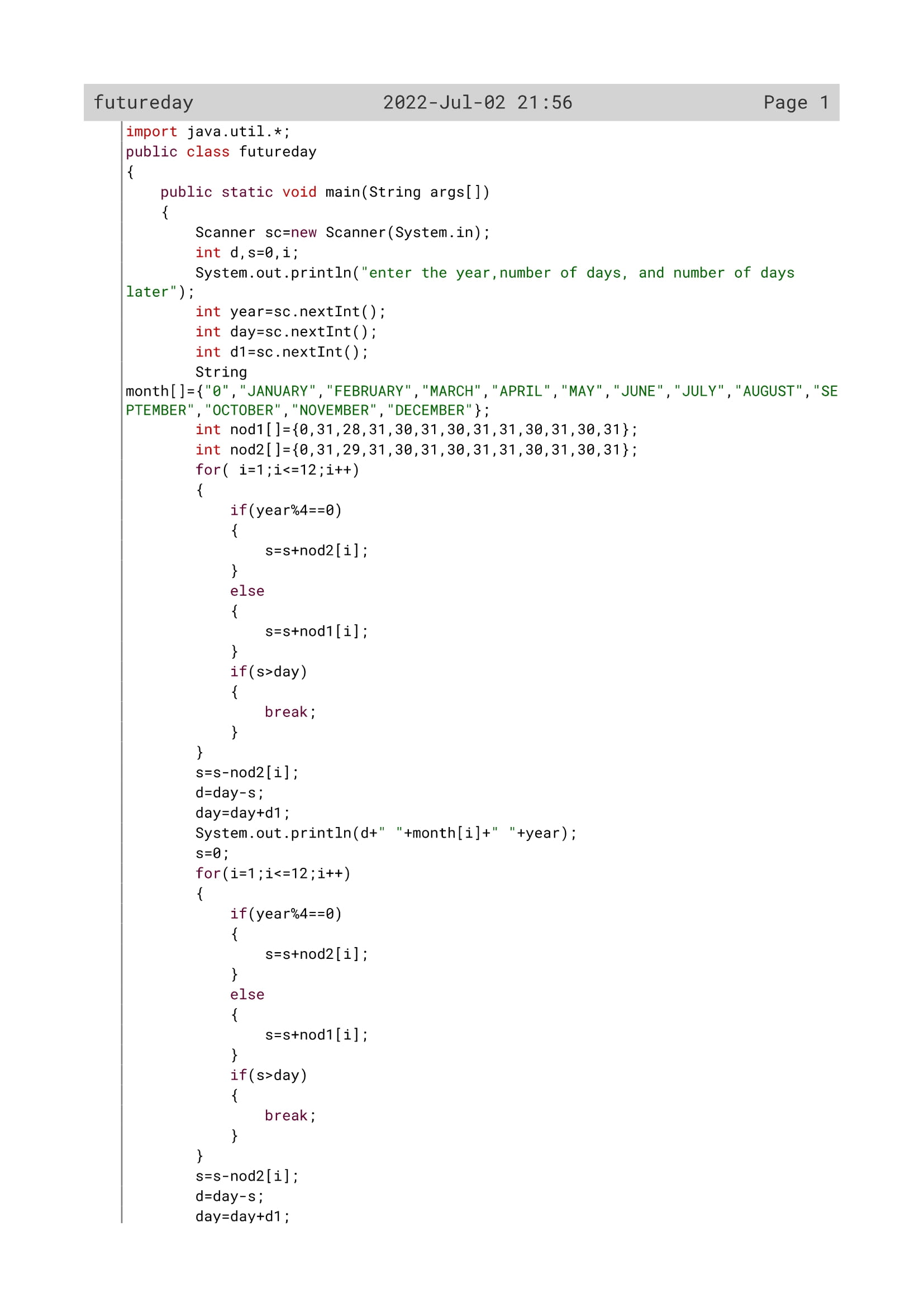
Step 10.1: check if the year is leap year or not. If it is then add the month’s total number of days nod2[i] to s variable else add the month’s total number of days nod1[i] to s variable and if s exceeds the variable day value then execute break statement

Step 11: perform s-nod[2] and store the value in s variable itself

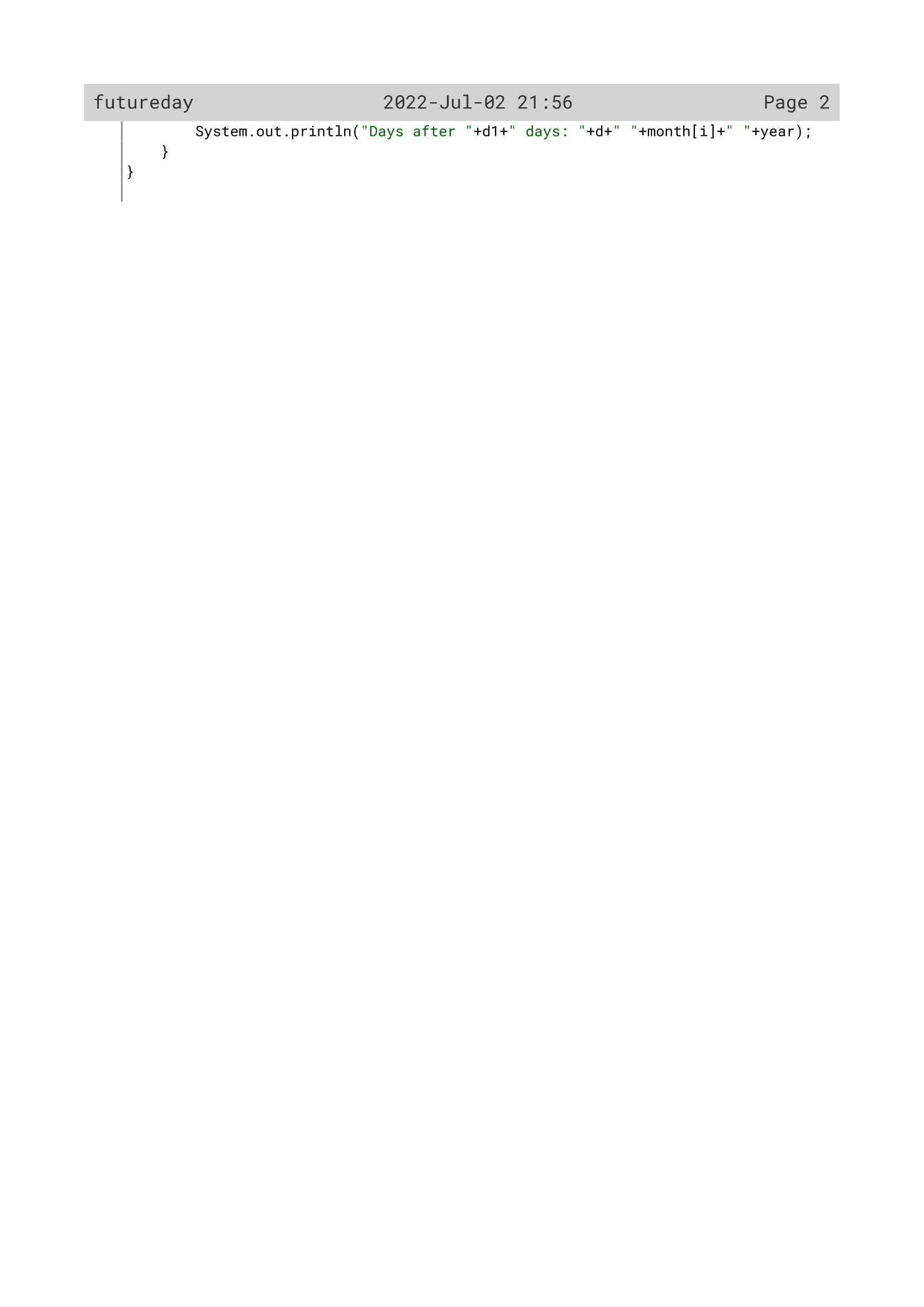
Step 12: in d variable store the value of day-s and is day variable add The day variable value with d1

Step 13: Print the day month and year as the future date .

66

Source Code

67



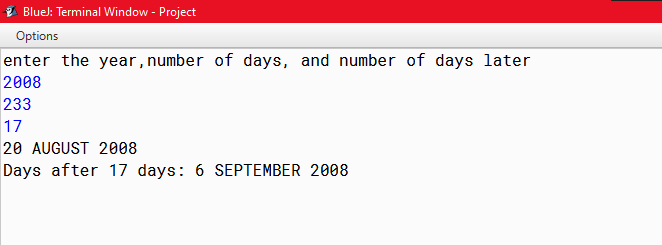
68

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | Description |
| d | Int | To input the number of days by the user |
| s | Int | To calculate the date |
| i | Int | Loop variable |
| year | Int | To input the year by the user |
| day | Int | To input the date that has to be increased with the initial one |
| d1 | Int | To calculate the date |
| nod1 | int | keeping total days in a month in an array of a normal year |
| nod2 | Int | keeping total days in a month in an array of a leap year |
| month[] | String | keeping total days in a month in an array of a normal year |

69

OUTPUT



70

PROGRAM 5

Write a program in java to accept date and weekdays name of the 1st day of that month. Display the weekday of that date.

Sample input:

Enter a date: 27/04/2017

Enter the 1st day of that month: Saturday

Sample output:

Day of the date is Thursday

71

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: The names of the weekdays are kept in array day[]

Step 4: The number of days of each months are kept in array nd[]

Step 5: use substring function to separate the day number, month

And the year stored in variable d, m, y and convert it to

int type in variable dd, mm, yy

Step 6: Check if the number of days is greater than 0 and less than

Equal to 31 and the number of months is greater than 0

and less than equal to 12 or not. If this satisfies then enter

the first day of the month and store it in variable b.

Step 7: Execute the following steps from 1 to 8

Step 7.1: If the first day of the month is equal to any of the weekday

Then initialize wd=i;

Step 8: Execute the following steps from 1 to mm using loop

variable i

Step 8.1: Add s with nd[] variable value based on the value of I

Step 8.2: Add s with dd variable value

Step 9: check if the year is a leap year and if the month is greater

than 2 or not. If it is then add 1 to s variable and get the

remainder by s%7 and store the value to w variable. Then

add wd with w and with that value subtract 1.Store the

result in t variable

Step 10: checks if t is less than equal to 7 or not.If it is print day[t-1];

else if t is greater than 7, then subtract 7 from t and print

day[t].

Step 11: Stop.

72

Source Code

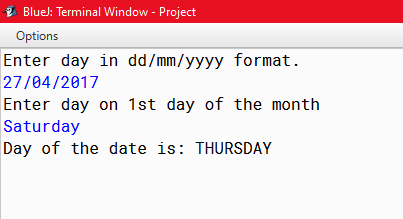
73

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | Description |
| day[] | String | Array to store all the all the week names |
| y | String | extracting the year |
| m | String | extracting the month |
| d | String | extracting the date |
| b | String | To store first day value |
| dd | Int | To convert the extracted date from string to int data type |
| yy | Int | To convert the extracted year from string to int data type |
| i | Int | Loop variable |
| s | Int | To calculate sum of nd[] values |
| w | Int | To store remainder value of s |
| t | Int | To act as array counter |
| wd | Int | To store weekday value |
| mm | int | To convert the extracted month from string to int data type |
| nd[] | Int | To enter the total days in a month in an array |

74

OUTPUT



75

PROGRAM 6

Write a program in java to input long integer data not less than five digits. Your program should reject it if data is less than five digits. In the input, last four digit will be taken as year (validity of the year should be checked) and remaining digit as total number of days. Your program should display the output as actual date.

Sample input: 272008

Sample output: 27 January 2008

Sample input: 362017

Sample output: 5 February 2017

76

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: Enter long integer number not less than 5 digits.

Step 4: Convert the number to String and find its length.

Step 5: The number of days of each months in a leap year are kept

in array nod1[]

Step 6: The number of days of each months in a normal year are

kept in array nod2[]

Step 7: The names of the months are kept in array month[]

Step 8: Using substring function extract the last 4 digits as the year,

And extract the previous digits as the day number. Convert

them to int type

Step 9: check the validity of the year if it is between 1900 to 2300

Step 10: check if the year is leap year or not. If it is then run i loop

From 0 to 12. Add the nod2[i] value based on the value of i

With the s variable. When s is greater than the day

number, break statement is executed.

Step 11: now subtract the day number from s variable and store it

In r variable. Now subtract r variable from nod2[i], to get

The actual day of the date

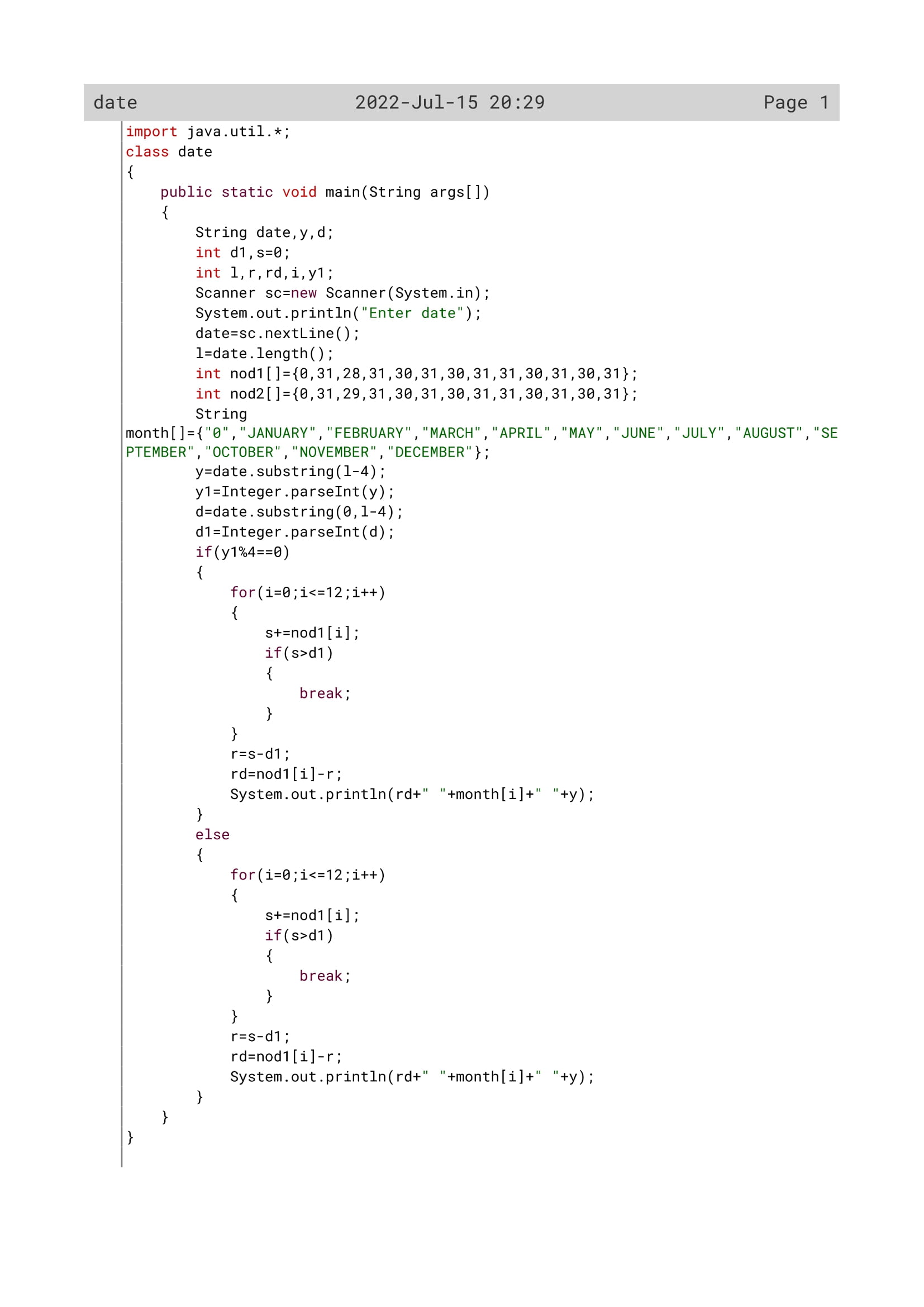
Step 12: display the day, month, and the year.

Step 13: Repeat all the steps from step 10 to 12,if the year is not a

Leap year.

Step 14: Stop.

77

Source Code

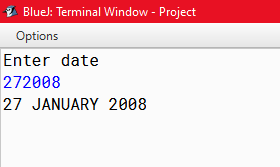
78

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | description |
| date | String | To store date value |
| y | String | To store year value |
| d | String | To store day value |
| d1 | Int | To store day value in integer form |
| l | Int | To store length of date value |
| r | Int | To store s-d1 value |
| rd | Int | To store nod1[i]-r value |
| y1 | Int | To store year value in integer form |
| nod1[] | Int | To store month days in a non leap year |
| nod2[] | int | To store month days in a leap year |
| s | int | To store summation value of the month days |

79

OUTPUT



80

PROGRAM 7

Design a program which accepts your date of birth in dd/mm/yyyy format. Check whether the date entered is valid or not. If it is valid, display “VALID DATE” , also compute and display the day number of the year for the date of birth. If it is invalid, display “INVALID DATE” and then terminate the program.

Input:

Enter your date of birth in dd/mm/yyyy format: 03/04/2010 Output:

Valid Date

93

81

Algorithm

Step 1: Start

Step 2: Scanner Class object is declared

Step 3: Enter a date from the user.

Step 4: use substring function to separate the day number, month and the year stored in variable d, m, y and convert it toint type in variable dd, mm, yy.

Step 5: The number of days of each months in a leap year are kept in array nod1[]

Step 6: The number of days of each months in a normal year are kept in array nod2[]

Step 7: Check for the validity of the program.

Step 7.1: check if the year is leap year or not

Step 7.2: check if the month is 2 or not, the day is between 1 to 29 or not. If the three conditions satisfy then initialize fd=1;

Step 7.3: check if the date is between 1 and the last date of the Month, nod[month]. If this satisfies then initialize fd=1

Step 7.4: If the year is not a leap year then check if the date is between 1 and the last date of the Month, nod[month]. If this satisfies then initialize fd=1.

Step 7.5: now check if the month if between 1 to 12 or not. If it is then initialize fm as 1

Step 7.6: now check if fd=1 and fm=1.If they are then print the date as valid date, else print not valid

Step 8: If the year is leap year then initialize s=0, and run a loop I From 0 to less than mm. Add the total days of the month,nod1[i], based on the ith value with s variable.

Step 9: If the year is not a leap year then initialize s=0, and run a loop From 0 to less than mm. Add the total days of the month, nod1[i], based on the ith value with s variable.

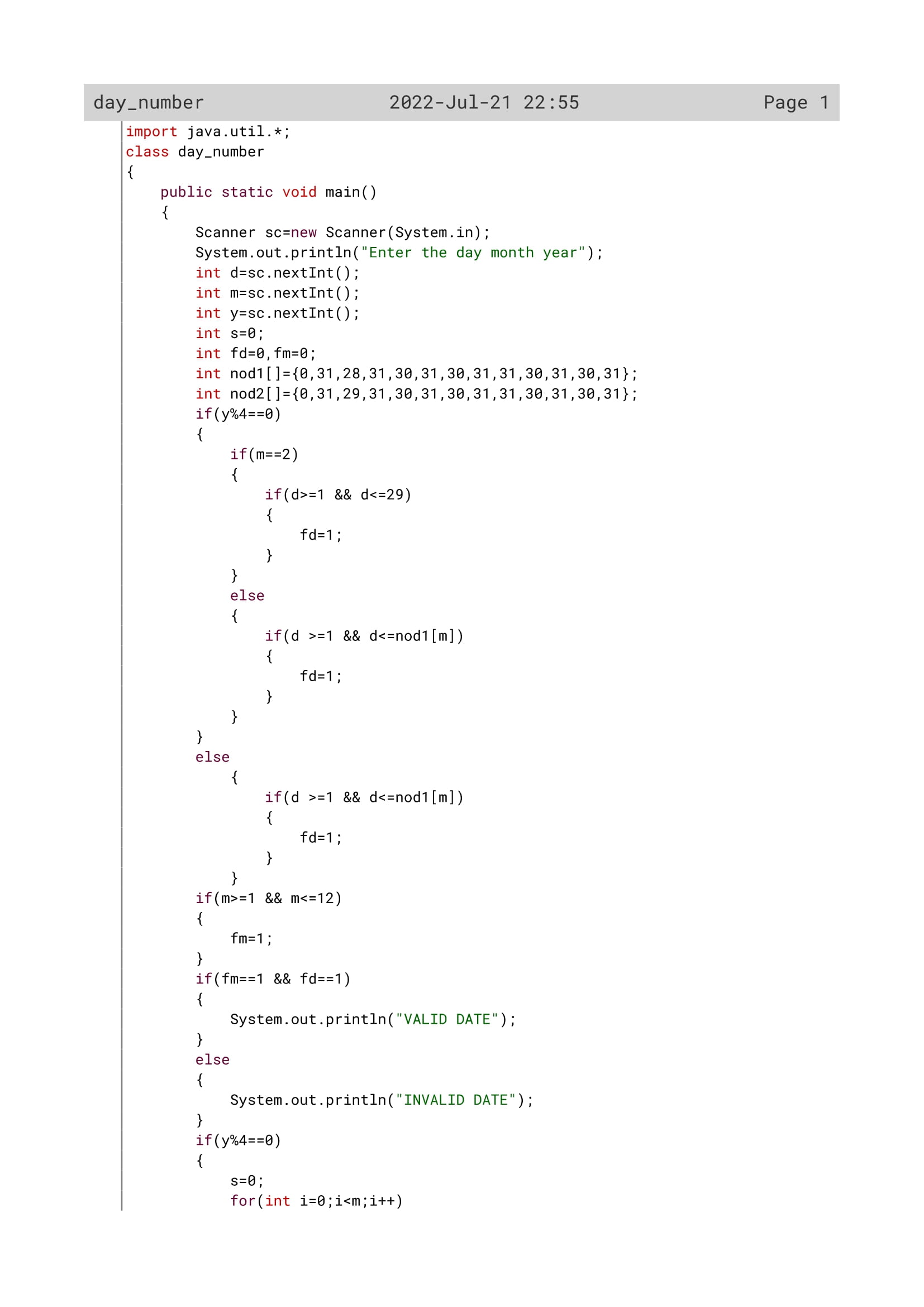
Step 10: now add the s variable with dd variable to get the day Number.

Step 11: Print the day number

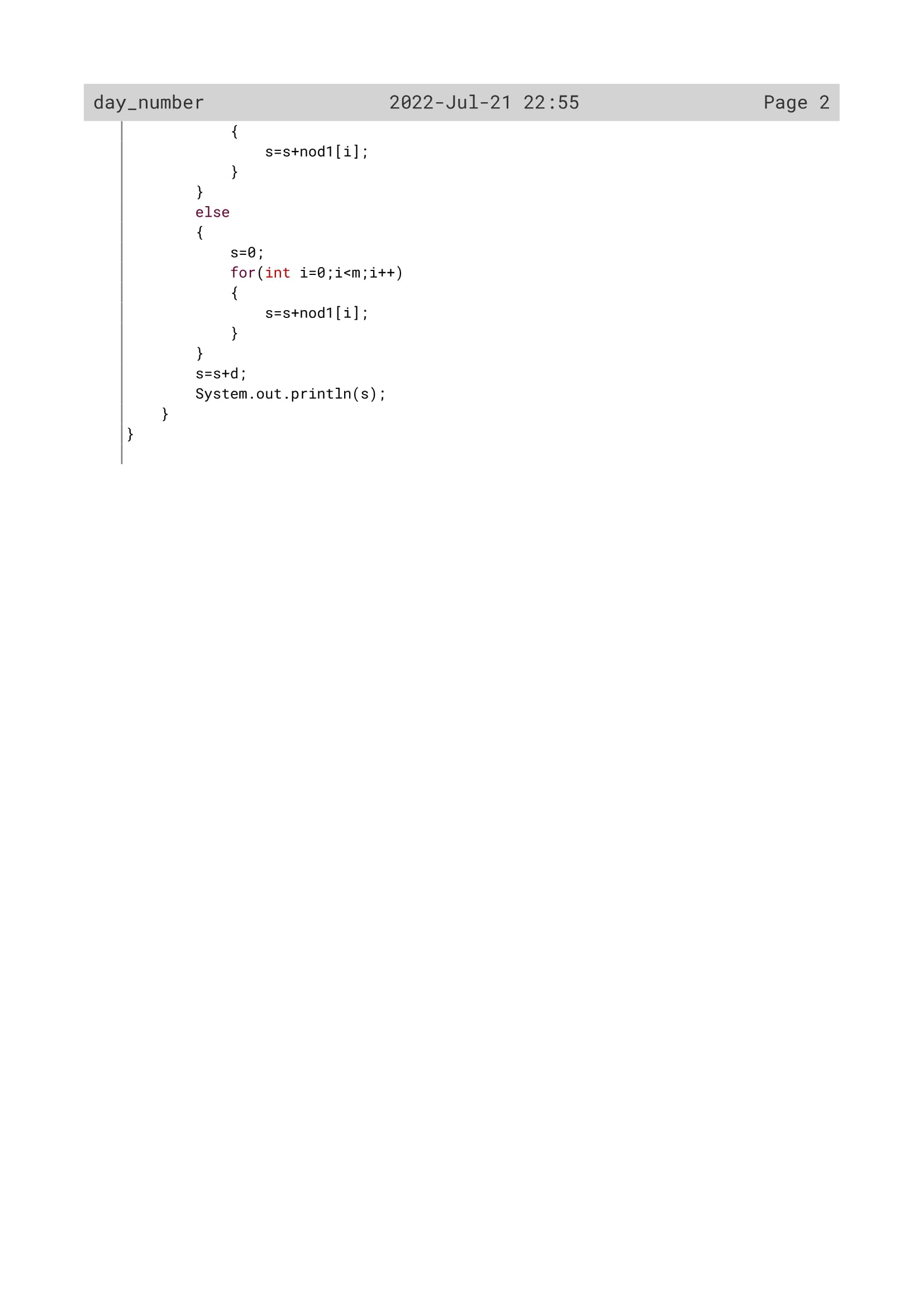
Step 12: Stop.

82

Source Code



83



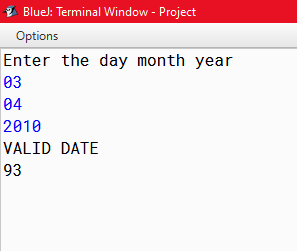
84

Variable Description

|  |  |  |
| --- | --- | --- |
| Variable name | Data type | description |
| y | Int | To store year value |
| d | Int | To store day value |
| m | Int | To store month value |
| fd | Int | To act as flag variable |
| fm | Int | To act as flag variable |
| nod1[] | Int | To store month days in a non leap year |
| nod2[] | int | To store month days in a leap year |
| s | int | To store summation value of the month days |
| i | int | To act as counter variable |

85

OUTPUT



86