*# AddDaysToADate.py  
  
# Program: Given a date in format "YYYY-MM-DD" as a string and n the number of day (between 1 and 5000),  
# output the resulting date after adding n days to the given date.  
# The years are between 1900 and 2100  
# n is between 1 and 50000***def** is\_leap\_year(year):  
 **if** year % 100 == 0:  
 **if** year % 400 == 0:  
 **return True  
 else**:  
 **if** year % 4 == 0:  
 **return True  
 return False  
  
def** add\_days\_to\_date(date, n):  
  
 days\_in\_month = [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]  
  
 date\_list = date.split(**'-'**)  
 year = int(date\_list[0])  
 month = int(date\_list[1])  
 day = int(date\_list[2])  
  
 **if** is\_leap\_year(year):  
 days\_in\_month[2] = 29  
  
 **while** n > 0:  
 **if** day + n <= days\_in\_month[month]:  
 day += n  
 n = 0  
 **else**:  
 n -= (days\_in\_month[month] - day + 1)  
 day = 1  
 month += 1  
 **if** month == 13:  
 month = 1  
 year += 1  
 **if** is\_leap\_year(year):  
 days\_in\_month[2] = 29  
 **else**:  
 days\_in\_month[2] = 28  
 **return f"{**year**:04d}-{**month**:02d}-{**day**:02d}"  
  
def** main():  
 inputDate = **"1999-01-01"** n = 365  
 result = add\_days\_to\_date(inputDate, n)  
 print(**"inputDate = "**, inputDate, **"n = "**, n, **"result = "**, result)  
 print(**"-------------------------------"**)  
 inputDate = **"2000-01-01"** n = 365  
 result = add\_days\_to\_date(inputDate, n)  
 print(**"inputDate = "**, inputDate, **"n = "**, n, **"result = "**, result)  
 print(**"-------------------------------"**)  
 inputDate = **"2000-01-01"** n = 366  
 result = add\_days\_to\_date(inputDate, n)  
 print(**"inputDate = "**, inputDate, **"n = "**, n, **"result = "**, result)  
 print(**"-------------------------------"**)  
 inputDate = **"2000-01-01"** n = 20000  
 result = add\_days\_to\_date(inputDate, n)  
 print(**"inputDate = "**, inputDate, **"n = "**, n, **"result = "**, result)  
 print(**"-------------------------------"**)  
  
**if** \_\_name\_\_==**'\_\_main\_\_'**:  
 main()