

Assignment Date : 7/5/2024

Due Date : 28/5/2024 at 23:59

- Student must do the homework without any collaboration or help.
- Automatic plagiarism detection software will be used for scanning and checking the submitted files.
- If significant similarities are found between submitted files, it will be considered as plagiarism, and those homework grades will be zero.
- File should be submitted to Ninova only, email submission is not accepted.
- Program should be compiled without syntax errors.
- It should work efficiently and generate correct results as expected.
- Solution should be according to the class specifications given below. Any other methods will not be graded.

**BOOK CLASS**

- Write the C++ codes of the UML class diagram given on the right. (Do not make any modifications to the class specifications.)
- The parametered constructor function initializes member data with the parameters.
- The print function displays all member data on screen.

Book	
+ book_id	: int
+ publication_year	: int
+ author_ID	: int
+ publisher_ID	: int
+ book_title	: string
+ Book ()	
+ Book (int, int, int, int, string)	
+ print() : void	

**MAIN PROGRAM**

Write a C++ program to do followings.

By looping and using C++ file input statements, read all records from the ASCII text files described below.

(There are 3 text files in the DATA\_FILES.ZIP file.)

Do not make any data content modifications or format editings in text files.

Your program should be general, so that it works for any number of records in the files.

The following is a sample code for reading the fields from the AUTHORS.TXT file:

```
.....
while ( !dosya.eof() ) {
    dosya >> author_id;
    getline(dosya, author_fullname);
    .....
}
```

PUBLISHERS.TXT file format:

publisher_id	publisher_name
--------------	----------------

AUTHORS.TXT file format:

author_id	author_fullname
-----------	-----------------

BOOKS.TXT file format:

book_id	publication_year	author_ID	publisher_ID	book_title
---------	------------------	-----------	--------------	------------

Refers (link) to  
the author\_id  
in AUTHORS file

Refers to the  
publisher\_id  
in PUBLISHERS file

A book has only one author and one publisher. An author may have one or more books.

The **STANDARD TEMPLATE LIBRARY (STL)** will be used for storage of data.

Store all data that were read from text files into the **STL map variables** defined globally as below.

**map <int, string> publishers;** // First field is publisher\_id, second field is publisher\_name

**map <int, string> authors;** // First field is author\_id, second field is author\_fullname

**map <int, Book> books;** // First field is book\_id, second field is object of Book class

## EXAMPLE SCREEN OUTPUT

By looping through the publishers map, generate a screen output that contains Lists of All Books Grouped By Publishers. The publishers, authors, and books map variables should be used when printing information on screen. In the print member function of Book class, use the built-in `setw` function, defined in `<iomanip>` header, to obtain column alignments on screen output. Also display a dashed line (such as “-----”) as a separator between publishers.

```
LISTS OF ALL BOOKS GROUPED BY PUBLISHERS
=====

PUBLISHER ID      : 001
PUBLISHER NAME    : APRESS MEDIA
NUMBER OF BOOKS   : 2

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1004     2018               Ivor Horton      Beginning C++17 From Novice to Professional
1015     2020               George Leimers    Modern C++ for Absolute Beginners
-----

PUBLISHER ID      : 002
PUBLISHER NAME    : JOHN-WILEY
NUMBER OF BOOKS   : 4

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1012     2014               Mark Allen Weiss  Data Structures and Algorithm Analysis in C++
1016     2017               Lawrence Strutton  Numerical Recipes in C
1022     2018               Brian Kernighan    The C programming Language
1023     2015               Bruce Eckel        Programming in C++
-----

PUBLISHER ID      : 003
PUBLISHER NAME    : MCGRAW-HILL
NUMBER OF BOOKS   : 5

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1002     2016               Timothy McPeice    Beginning Algorithms
1008     2012               Paul Hawwess       C++ for Everyone
1010     2015               Michael Dundredge   C++ Quick Syntax Reference
1011     2018               Edward Schultz      Cross-Platform Development in C++
1019     2019               Harris Wanstead     Object-Oriented Programming Using C++
-----

PUBLISHER ID      : 004
PUBLISHER NAME    : MICROSOFT PRESS
NUMBER OF BOOKS   : 2

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1005     2012               Ivor Horton       Beginning Visual C++
1024     2018               Jeffrey Richter    Windows via C/C++
-----

PUBLISHER ID      : 005
PUBLISHER NAME    : PACKT PUBLISHING
NUMBER OF BOOKS   : 3

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1003     2019               John Sanfield     Beginning C++ Game Programming
1020     2016               Pierre Wieleon     Object-oriented Programming with Java
1021     2017               Steve Newark       Practical C++ Programming
-----

PUBLISHER ID      : 006
PUBLISHER NAME    : PEARSON EDUCATION
NUMBER OF BOOKS   : 3

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1007     2016               Paul Deitel       C How to Program
1009     2016               Harvey Deitel      C++ How to Program
1013     2012               Robert Lafore      Java Programming Language
-----

PUBLISHER ID      : 007
PUBLISHER NAME    : PRENTICE-HALL
NUMBER OF BOOKS   : 5

BOOK_ID  PUBLICATION_YEAR  AUTHOR_FULLNAME  BOOK_TITLE
1001     2014               James Riverton     An Introduction to GCC Toolchain
1006     2013               John Bronson        C for Engineers and Scientists
1014     2017               Herbert Schildt     Java The Complete Reference
1017     2017               Aurelie Linden      Object-Oriented Analysis and Design
1018     2012               Robert Lafore       Object-Oriented Programming in C++
-----

TOTAL NUMBER OF ALL BOOKS : 24

Program finished.
```