|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Task No. | 3 | Task Title | | Practical Tasks | | | | Due Date | 21/06/2019 | |
| Unit / Module details | | | | | | | | | | |
| Qualification/Course | | | Code | ICT60515 | Title | Advanced Diploma of Computer Systems Technology | | | | |
| Unit/Module | | | Code | ICTPRG527 | Title | Apply Intermediate Object-Oriented Language Skills | | | | |
| Assessment details | | | | | | | | | | |
| Assessor | | | Assessor’s name | | Vikram Behal | | Assessor’s phone | | | 03 9354 6130 |
|  | | | Assessor’s email | | vbehal@kangan.edu.au | | | | | |
| Elements / Learning outcomes | | | Click here to enter text. | | | | | | | |
| Task overview*(see below for specific instructions and criteria)* | | | This assessment item requires students to work in individually and accomplish their task professionally and in time. It is their responsibility to inform/contact teacher if they need any clarification to accomplish this task. This assessment item is open book and they can use any available resources. This assessment item is practical task and students need to complete all practical tasks as per Assessment item Description (At the end of this document).Assessment item has 6 tasks and students are required to complete all six tasks to be competent in this assessment. | | | | | | | |
| Assessment conditions | | | * This information is to be handed to each student to outline the assessment requirements. * This is open book assessment answer all questions. * This Assessment Item requires students to work individually and address all requirements as per Assessment item Description * Students are required to accomplish their task professionally and in time * Students must submit this Assessment item at MyLearning * This assessment item is classroom based * This assessment item is due in week 7 * Student can negotiate time with teacher under special circumstances * Read the Plagiarism policy and procedure carefully to understand the consequences that you could face if your work is plagiarized. * You plagiarize if you present other writers' words or ideas as your own. * You must use citations to document all ideas and significant information that are not your own. * Submit this document by the due date (penalties applies) * Insert your name and student id in the footer. * Use Font: Arial Font Size: 10 Heading: Bold Arial Heading Size: 12 Sub Heading: Bold Arial Sub Heading Size: * Marking Criteria for this assessment Item is as under | | | | | | | |
| Resources | | | Resources are available through MyLearning | | | | | | | |
| How to submit | | | Complete Assessment items as per assessment Item description and upload assessment item at MyLearning  All items submitted must be clearly marked with the following details:  • Your full name  • Your student number  • Your class group  • The date  This cover sheet must accompany all items submitted. | | | | | | | |

| Assessment instructions and criteria | | | |
| --- | --- | --- | --- |
| With competency based assessment **all** assessment requirements for each assessment task must be assessed as satisfactory **(S)** for a competent **(CA)** result to be recorded. If an assessment result for any task is assessed as not satisfactory **(NS)** a resubmission will be required for the outstanding **(NS)** assessment task.  Rows can be deleted by left clicking to the left of the row and click *Backspace*  Rows can be added by left clicking to the left of the row and click *CTRL+C* and then *CTRL+V* | | | |
| **Instruction/Task** | **Criteria that you will be assessed on** | **S** | **NS** |
| Student need to complete all practical tasks as per assessment Item Description | Student has completed all practical tasks as per assessment Item Description | ☐ | ☐ |
| 1. Students are required to create SQL File/script to do following:    1. Create one database called books    2. Create user 'user3'@'localhost' password 'Password01' and GRANT him permissions like SELECT, INSERT, UPDATE and DELETE.    3. CREATE TABLE Book as under (bookid, Bookname and Authorid are three fields in this table):   bookid VARCHAR(10) PRIMARY KEY,  Bookname VARCHAR(30),  Authorid VARCHAR(10)   * 1. CREATE TABLE Author as under (Author\_id, FirstName, LastName and email are fields):   Author\_id VARCHAR(10) PRIMARY KEY,  FirstName VARCHAR(30),  LastName VARCHAR(30),  email VARCHAR(30),   * 1. ALTER TABLE book ADD FOREIGN KEY Authorid and link it with Author\_id of Author (Books database) as under:   (Authorid) REFERENCES Author(Author\_id);   * 1. Inset at least four records in both tables   2. Import script in your WAMP make sure you logged in as root in WAMP.   3. Upload SQL file/script with this file | Students created SQL File/script to do following:   * 1. Create one database called books   2. Create user 'user3'@'localhost' password 'Password01' and GRANT him permissions like SELECT, INSERT, UPDATE and DELETE.   3. CREATE TABLE Book as under (bookid, Bookname and Authorid are three fields in this table):      1. bookid VARCHAR(10) PRIMARY KEY,      2. Bookname VARCHAR(30),      3. Authorid VARCHAR(10)   4. CREATE TABLE Author as under (Author\_id, FirstName, LastName and email are fields):      1. Author\_id VARCHAR(10) PRIMARY KEY,      2. FirstName VARCHAR(30),      3. LastName VARCHAR(30),      4. email VARCHAR(30),   5. ALTER TABLE book ADD FOREIGN KEY Authorid and link it with Author\_id of Author (Books database) as under:   (Authorid) REFERENCES Author(Author\_id);   * 1. Inset at least four records in both tables   2. Import script in your WAMP make sure you logged in as root in WAMP.   3. Upload SQL file/script with this file | ☐ | ☐ |
| 1. Students are required to create a java file to display all records of table books. Make sure it should display following fields:    1. Student must add table header.    2. Student must display following fields :   bookid, Bookname   * 1. Organise this list on bookid | Students create java class to display all records of table books. As per requirements ( it displayed following fields:   * 1. Student must add table header.   2. Student must display following fields :   bookid, Bookname   * 1. Organise this list on bookid | ☐ | ☐ |
| 1. Students are required to create a java file to display all records of table books. Make sure it should display following fields:    1. Student must add table header.    2. Student must display following fields :   bookid, Bookname, FirstName, LastName and email  (bookid, Bookname fields are from book, however FirstName, LastName and email are from table Author. Make sure list must include all books)   * 1. Organise this list on bookid | Students createed a java java to display all records of table books. Make sure it should display following fields:   * 1. Student must add table header.   2. Student must display following fields :   bookid, Bookname, FirstName, LastName and email  (bookid, Bookname fields are from book, however FirstName, LastName and email are from table Author. Make sure list must include all books)   * 1. Organise this list on bookid | ☐ | ☐ |
| 1. Students are required to create user documentation for above program. User documentation must include following:    1. User documentation must have screen prints to help users    2. User documentation what user should not do    3. Student must add test results in user documentation    4. Table of Content should be part of user documentation    5. Index should be part of user documentation. | Students are required to create user documentation for above program. User documentation must include following:   * 1. User documentation must have screen prints to help users   2. User documentation what user should not do   3. Student must add test results in user documentation   4. Table of Content should be part of user documentation   5. Index should be part of user documentation. | ☐ | ☐ |
| 1. Students are required to create a java Program to read data from Provided file “Data\_R\_01.xls”. Student must do following:    1. Convert data into correct format    2. Add screen print of your output here    3. Upload your java code with this file | Students created a java Program to read data from Provided file “Data\_R\_01.xls”. Student did following:   * 1. Convert data into correct format   2. Add screen print of your output here   3. Upload your java code with this file | ☐ | ☐ |
| 1. Students are required to create a java program to write data using keyboard input into text file “Data\_W\_01.txt”    1. Text File should be created for Student Id, First Name, Last name and Mobile Number    2. User must enter input (Id, Name, lastname and Mobile) through keyboard    3. Program must allow user to enter data until user wants    4. Add screen print of your output here    5. Upload your java code with this file    6. Upload created text file Data\_W\_01.txt | Students created a java program to write data using keyboard input into text file “Data\_W\_01.txt”. Student did following:   * 1. Text File should be created for Student Id, First Name, Last name, Postcode and Mobile Number   2. User must enter input (Id, Name, lastname, Postcode and Mobile Number) through keyboard   3. Program must allow user to enter data until user wants   4. Add screen print of your output here   5. Upload your java code with this file   6. Upload created text file Data\_W\_01.txt | ☐ | ☐ |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Student details | | | | | | | | | | | |
|  | Student’s name | | | Click here to enter text.  Benjamen Calleja | | Student’s ID | | | Cal14385330 | | |
| Unit/Module details | | | | | | | | | | | |
| Unit/Module | Code | | ICTPRG527 | Title | Apply Intermediate Object-Oriented Language Skills | | | | | | |
|  | Task title | | Practical Tasks | | | | | | | | |
| Assessment feedback / comments | | | | | | | | | | | |
| Click here to enter text. | | | | | | | | | | | |
| Re-submission (where applicable) | | | | | | | | | | | |
| Unsatisfactory tasks are to be re-submitted according to the details below. Type NA if not applicable. | | | | | | | | | | | |
| **Task (or component of task) to be re-submitted** | | **Additional evidence required** | | | | | **Re-Submission date** | | | **S** | **NS** |
| Task for re-submission | | Evidence | | | | | Date | | |  |  |
| Task for re-submission | | Evidence | | | | | Date | | |  |  |
| Task for re-submission | | Evidence | | | | | Date | | |  |  |
| Result for this assessment task | | | | | | | | | | | |
|  | Your final result for this unit will be provided on the ***Unit Result Record*** at completion of all assessment tasks | | | | | | | | | | |
|  | Result | | | Choose an item. | | | | | | | |
|  | Assessor’s signature | | |  | | | **Date** |  | | | |

**Assessment item Description: To be competent in this assessment item students are required to complete all tasks.**

1. Students are required to create SQL File/script to do following:
   1. Create one database called books
   2. Create user 'user3'@'localhost' password 'Password01' and GRANT him permissions like SELECT, INSERT, UPDATE and DELETE.
   3. CREATE TABLE Book as under (bookid, Bookname and Authorid are three fields in this table):

bookid VARCHAR(10) PRIMARY KEY,

Bookname VARCHAR(30),

Authorid VARCHAR(10)

* 1. CREATE TABLE Author as under (Author\_id, FirstName, LastName and email are fields):

Author\_id VARCHAR(10) PRIMARY KEY,

FirstName VARCHAR(30),

LastName VARCHAR(30),

email VARCHAR(30),

* 1. ALTER TABLE book ADD FOREIGN KEY Authorid and link it with Author\_id of Author (Books database) as under:

(Authorid) REFERENCES Author(Author\_id);

* 1. Inset at least four records in both tables
  2. Import script in your WAMP make sure you logged in as root in WAMP.
  3. Upload SQL file/script with this file

[ add screen print of output of your import here and upload SQL file created by you with this document]

Create database books;

create user 'user3'@'localhost' identified by 'Password01';

use books;

grant SELECT, INSERT, UPDATE, DELETE on books.\* TO 'user2'@'localhost' identified by 'Password01';

CREATE TABLE Book(

bookid VARCHAR(10) PRIMARY KEY,

Bookname VARCHAR(30),

Authorid VARCHAR(10));

CREATE TABLE Author(

Author\_id VARCHAR(10) PRIMARY KEY,

FirstName VARCHAR(30),

LastName VARCHAR(30),

email VARCHAR(30));

ALTER TABLE book ADD FOREIGN KEY (Authorid) REFERENCES Author(Author\_id);

INSERT INTO Author VALUES('A001','James','Miller','JamesLMiller@armyspy.com');

INSERT INTO Author VALUES('A002','Patricia','Anderson','PatriciaSAnderson@dayrep.com');

INSERT INTO Author VALUES('A003','Christopher','Akbar','ChristopherMAkbar@rhyta.com');

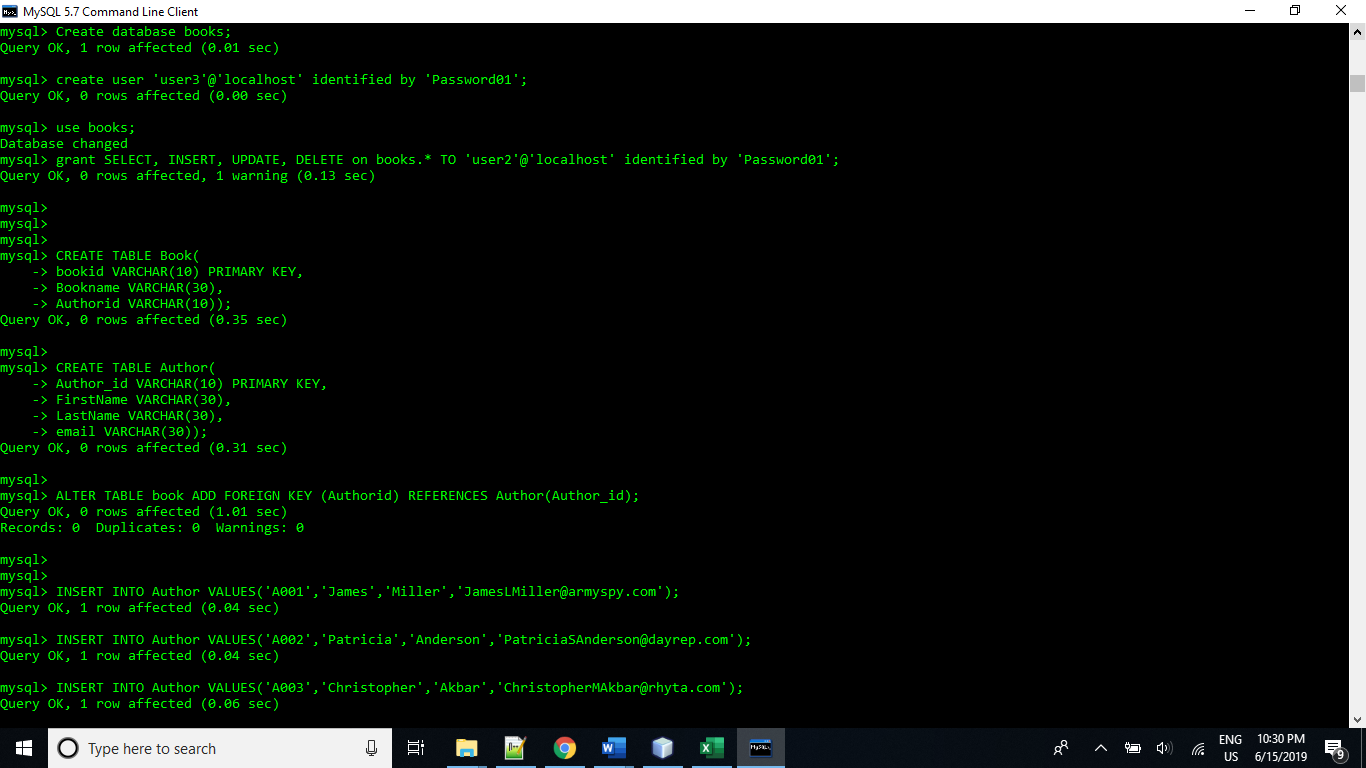
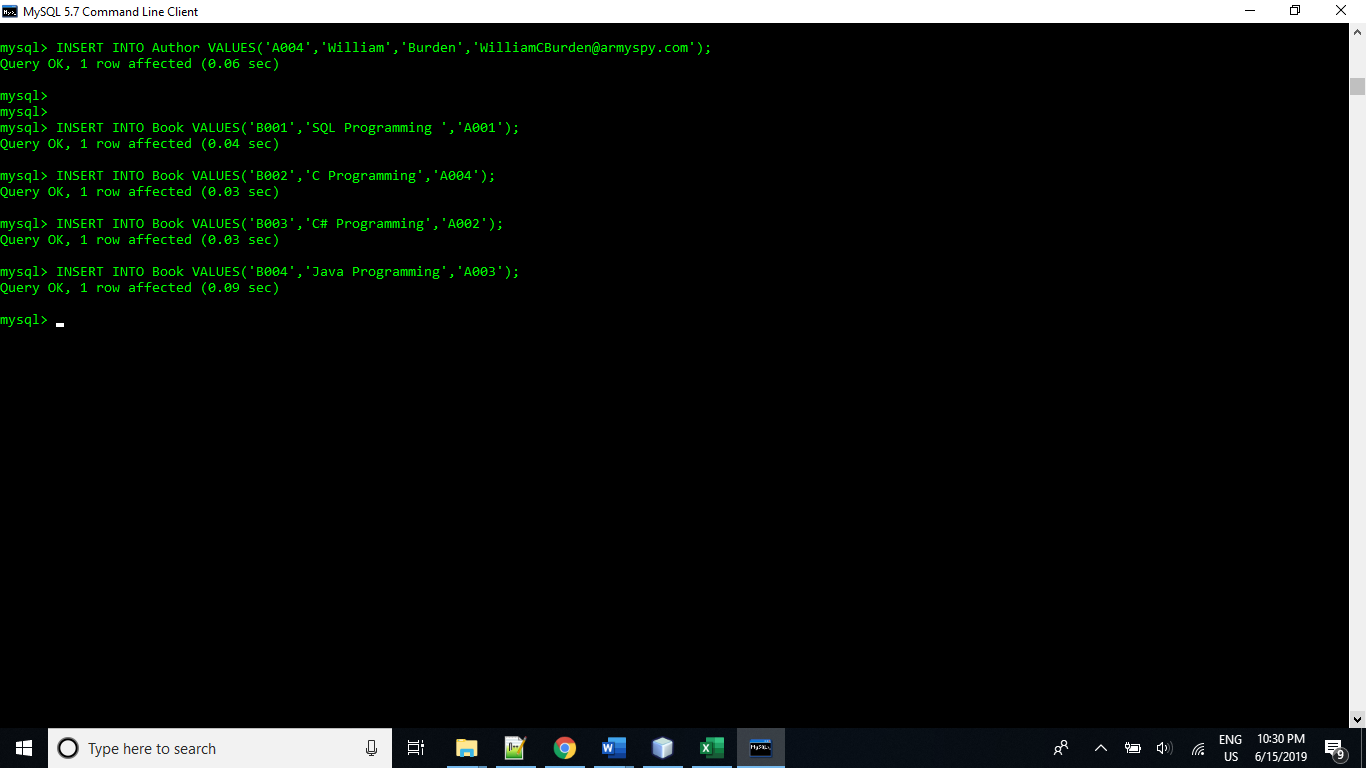
INSERT INTO Author VALUES('A004','William','Burden','WilliamCBurden@armyspy.com');

INSERT INTO Book VALUES('B001','SQL Programming ','A001');

INSERT INTO Book VALUES('B002','C Programming','A004');

INSERT INTO Book VALUES('B003','C# Programming','A002');

INSERT INTO Book VALUES('B004','Java Programming','A003');

1. Students are required to create a java class to display all records of table books. Make sure it should display following fields:
2. Student must add table header.
3. Student must display following fields :

bookid, Bookname

1. Organise this list on bookid

[ add screen print of output of your program here and upload java class along with this document]

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

public class Task2 {

public static void main(String[] args) {

Connection connection = null;

try {

connection = DriverManager.getConnection("jdbc:mysql://localhost/books", "root", "root");

System.out.println("List of books");

String query= "SELECT \* FROM book order by bookid";

Statement statement = connection.createStatement();

ResultSet resultSet = statement.executeQuery(query);

System.out.println(String.format("%-10s %s", "BookID","BookName"));

while (resultSet.next()){

System.out.println(String.format("%-10s %s", resultSet.getString("bookid"),resultSet.getString("Bookname")));

}

}

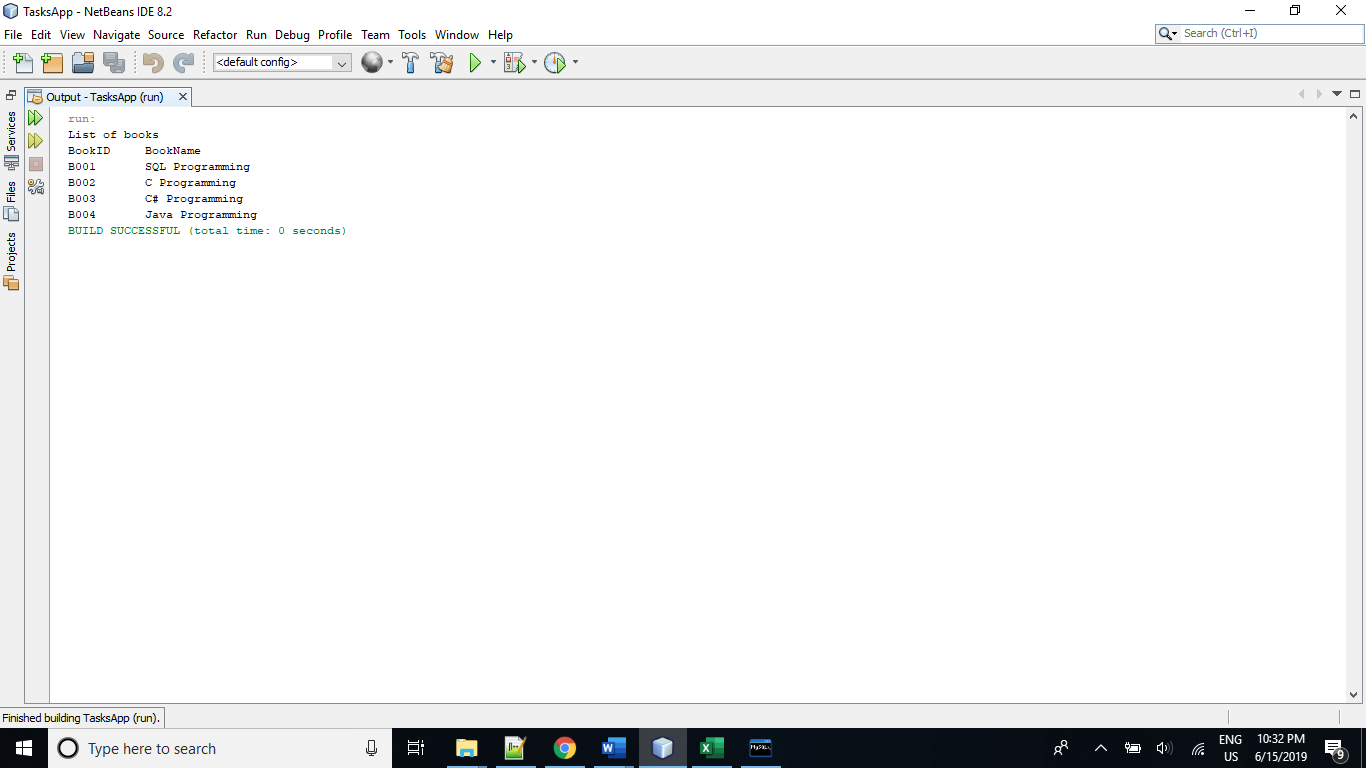
catch (Exception e) {

System.err.println(e);

}

}

}



1. Students are required to create a java class to display all records of table books. Make sure it should display following fields:
2. Student must add table header.
3. Student must display following fields :

bookid, Bookname, FirstName, LastName and email

(bookid, Bookname fields are from book, however FirstName, LastName and email are from table Author. Make sure list must include all books)

1. Organise this list on bookid

[ add screen print of output of your program here and upload java class along with this document]

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

public class Task3 {

public static void main(String[] args) {

Connection connection = null;

try {

connection = DriverManager.getConnection("jdbc:mysql://localhost/books", "root", "root");

System.out.println("List of books and their authors");

String query= "SELECT \* FROM book inner join author on author.Author\_id=Book.Authorid order by bookid";

Statement statement = connection.createStatement();

ResultSet resultSet = statement.executeQuery(query);

System.out.println(String.format("%-10s %-30s %-20s %-20s %s", "BookID","BookName","FirstName", "LastName", "Email"));

while (resultSet.next()){

System.out.println(String.format("%-10s %-30s %-20s %-20s %s",

resultSet.getString("bookid"),resultSet.getString("Bookname"),

resultSet.getString("FirstName"),resultSet.getString("LastName"),

resultSet.getString("email")));

}

}

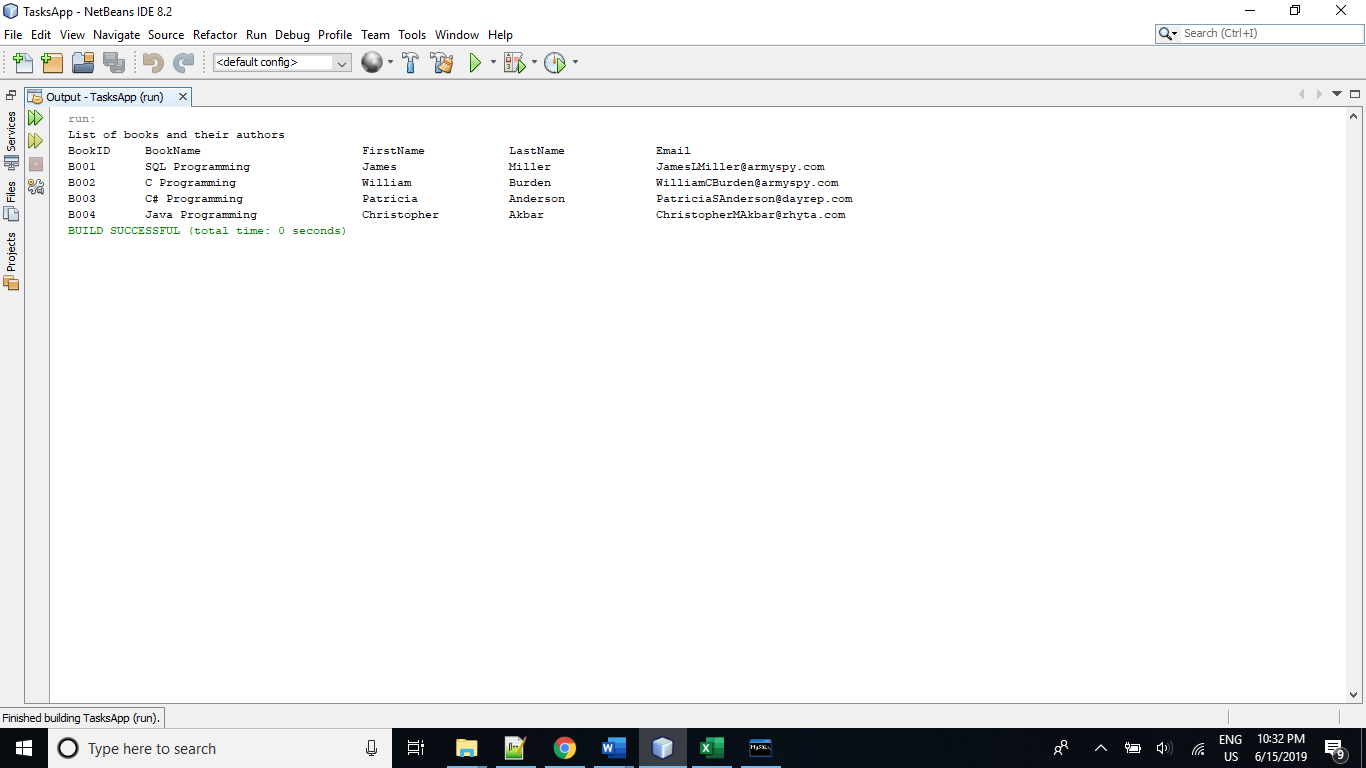
catch (Exception e) {

System.err.println(e);

}

}

}



1. Students are required to create user documentation for above program. User documentation must include following:
2. User documentation must have screen prints to help users
3. User documentation what user should not do
4. Student must add test results in user documentation
5. Table of Content should be part of user documentation
6. Index should be part of user documentation.

[ Student must upload word document or pdf version of user documentation ]

# References:

Schildt, H. (2017). *Java 2: The Complete Reference*. Boston, Massachusetts: McGraw Hill Professional.

Lassoff, M. (2017). *Java Programming for Beginners: Learn the fundamentals of programming with Java*. Birmingham: Packt Publishing.

1. Students are required to create a java Program to read data from Provided Text file “Data\_R\_01.xls”
   1. Convert data into correct format
   2. Add screen print of your output here
   3. Upload your java code with this file

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

import java.util.Scanner;

public class Task5 {

public static void main(String args[]){

try{

Scanner inFile=new Scanner(new FileReader(new File("Data\_R\_01.txt")));

while(inFile.hasNextLine()){

String line=inFile.nextLine();

String[] data=line.split("\t");

for(int i=0;i<data.length;i++){

System.out.print(String.format("%-20s",data[i]));

}

System.out.println();

}

}

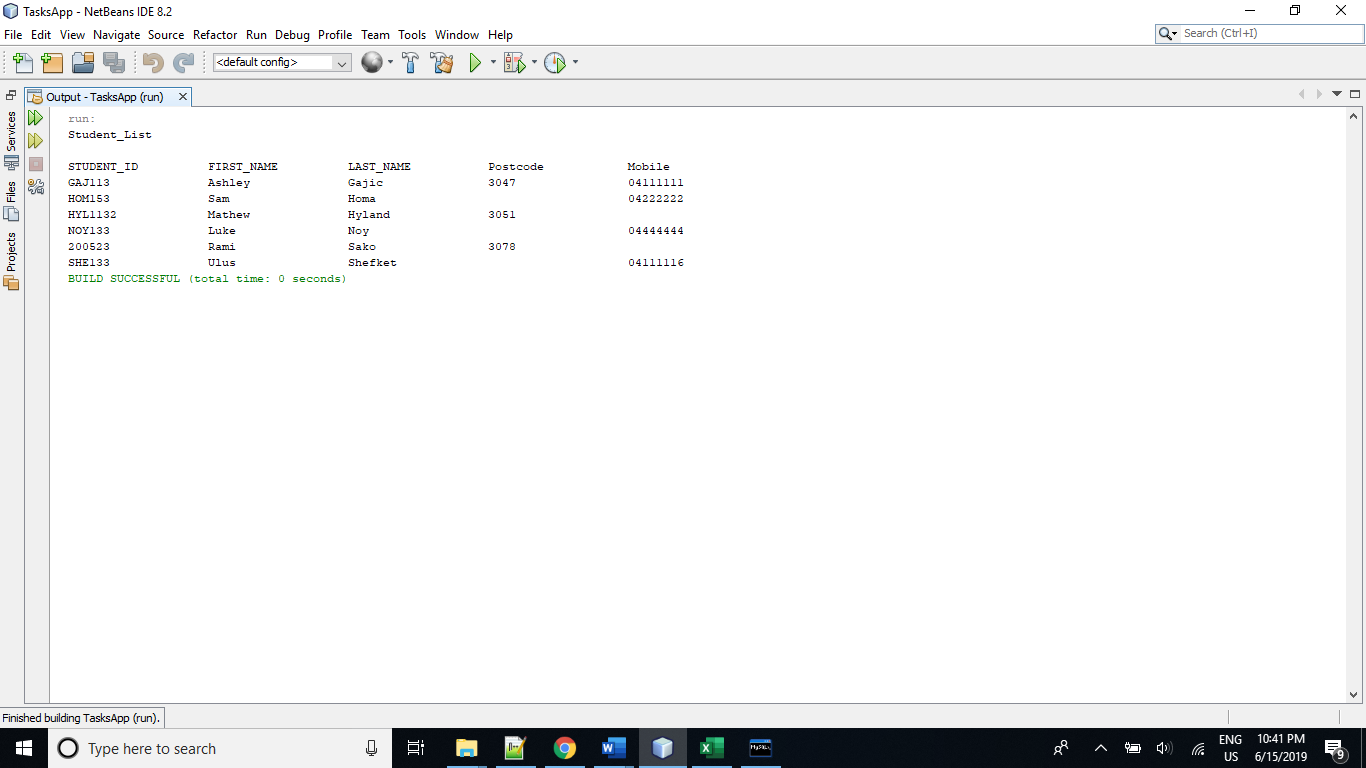
catch(IOException ex){

System.out.println("Error in reading data from file");

}

}

}



1. Students are required to create a java program to write data using keyboard input into text file “Data\_W\_01.txt”
   1. Text File should be created for Student Id, First Name, Last name Postcode and Mobile Number
   2. User must enter input (Id, Name, lastname Postcode and Mobile Number) through keyboard
   3. Program must allow user to enter data until user wants
   4. Add screen print of your output here
   5. Upload your java code with this file
   6. Upload created text file Data\_W\_01.txt

import java.io.FileWriter;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Scanner;

public class Task6 {

public static void main(String args[]){

try{

Scanner sc=new Scanner(System.in);

PrintWriter writer=new PrintWriter(new FileWriter("Data\_W\_01.txt"));

writer.println("Student Id,First Name, Last name, Postcode, Mobile Number");

writer.flush();

boolean loop=true;

while(loop){

System.out.print("Enter the Student ID: ");

String studentID=sc.nextLine();

System.out.print("Enter the Student First Name: ");

String firstName=sc.nextLine();

System.out.print("Enter the Student Last Name: ");

String lastName=sc.nextLine();

System.out.print("Enter the Student Postcode: ");

String postcode=sc.nextLine();

System.out.print("Enter the Student Mobile Number: ");

String mobileNumber=sc.nextLine();

writer.println(studentID+","+firstName+","+ lastName+","+postcode+","+mobileNumber);

writer.flush();

System.out.print("Do you want to enter another customer details (y/n)? ");

String option=sc.nextLine().toLowerCase();

if(option.equals("n")){

loop=false;

}

System.out.println();

}

}

catch(IOException ex){

System.out.println("Error in writing data to file");

}

}

}

