Page 1 of 47

Melbourne Polytechnic

Assessment – ICTWEB451 Apply structured query language in relational databases Assessor Guide



#### DEPARTMENT OF BUSINESS, ADVANCED MANUFACTURING AND LOGISTICS

ICT50120 Diploma of Information Technology

Assessment

**Learner**

**ICTWEB451 Apply structured query language in relational databases**

Assessment Book

Learner name:WangYiZhuo ID:S1554654

**Department:** Business, Advanced Manufacturing and Logistics

**Course:** ICT50120 Diploma of Information Technology

**Unit of Competency:** ICTWEB451 Apply structured query language in relational databases



Prepared by: Graham Gleich Melbourne Polytechnic Document creation date: November 2021

Version: 1.0

##### © Melbourne Polytechnic 2021 National Provider no. 3075

**Authors:** Graham Gleich, Teacher, Melbourne Polytechnic

**Disclaimer Statement:**

**PLEASE NOTE that by clicking on a link you may be directed to a third party’s site.**

**You should respect the intellectual property on that site. You may be leaving the Melbourne Polytechnic website. Melbourne Polytechnic does not endorse a linked site or guarantee the accuracy or currency of any information contained on the third party linked site.**

##### Acknowledgments

National training packages attributed as ***‘© Commonwealth of Australia 2013’*** Training packages are copied and communicated under [Creative Commons](https://creativecommons.org/licenses/by-nd/3.0/au/) [Attribution-Non Derivative 3.0 Australia](https://creativecommons.org/licenses/by-nd/3.0/au/) ([CC BY-ND 3.0 AUS](https://creativecommons.org/licenses/by-nd/3.0/au/)) license.

##### For information regarding material in this document, contact:

Graham Gleich Teacher

Business, Advanced Manufacturing & Logistics [grahamgleich@melbournepolytechnic.edu.au](mailto:grahamgleich@melbournepolytechnic.edu.au)

Contents

[Assessment Requirements 4](#_bookmark4)

[Introduction 4](#_bookmark5)

[Resubmission 5](#_bookmark6)

[Assessment appeals 5](#_bookmark7)

[Reasonable adjustment 5](#_bookmark8)

[Safety 5](#_bookmark9)

[Assessment Methods 6](#_bookmark10)

[Assessment Task 1: City Bookshop Database Proposal 7](#_bookmark0)

[Project Scenario 8](#_bookmark0)

[Create E.R. Diagram 9](#_bookmark0)

[Create Data Dictionaries 10](#_bookmark1)

[Assessment Checklist: Task 1, Part 1&2 Database 12](#_bookmark2)

[Assessment Task Summary – Task 1, Part 1&2 Database 14](#_bookmark3)

[Assessment Task 2: Knowledge Questions 15](#_bookmark0)

[Questions 16](#_bookmark0)

[Assessment Task 3: SQL Project 22](#_bookmark11)

[Scenario 23](#_bookmark12)

[Part 1: Create and populate database 23](#_bookmark13)

[Part 2: Test and verify query results 30](#_bookmark14)

[Assessment Checklist 3: Part 1&2 Database Assessment Task 3 45](#_bookmark15)

[Assessment Task Summary – Part 1&2 Database Assessment Task 3 47](#_bookmark16)

# Assessment Requirements

## Introduction

For this unit **ICTWEB451 Apply structured query language in relational databases** you will receive the following assessment books:

1. Assessment Book – Assessment (this book)

To achieve competence, you must satisfactorily complete all assessment tasks in the assessment books. The full set of tasks will enable you to demonstrate the skills and knowledge required for each unit.

All assessment tasks adhere to the unit of competency requirements from the training package and the principles of assessment and rules of evidence as required by Standards for Registered Training Organisations (RTOs) 2015.

You will be given clear instructions for all tasks. The assessment tasks ensure that you can provide sufficient evidence to demonstrate competence in each unit. If you need any variation to the assessment, you can discuss this with your assessor. See Reasonable adjustment section below.

Assessment tasks are designed and managed to allow you to demonstrate your skills and knowledge and verify that all work is your own.

To ensure fairness, consistency and reliability when marking assessments, assessors will be provided with checklists of expected outcomes, behaviours and required answers.

### Resubmission

If you do not achieve a satisfactory result for a task, you will be given the opportunity to be re-assessed and/or to provide additional evidence. You can be given multiple opportunities to resubmit or undertake an assessment task. The number of attempts will be limited by department resources and the end date of the unit/module. Arrangements will be made on an individual basis to ensure the process is valid, fair and reliable in line with the VET Assessment Policy and Procedures.

### Assessment appeals

If you are dissatisfied with the outcome of your assessment, discuss your concerns with the teacher/assessor and/or program leader. If the issue is not resolved, you may appeal the decision(s) by following the Complaints and Appeals process as outlined on the Melbourne Polytechnic Learner Portal.

### Reasonable adjustment

If you have a disability or long-term medical or mental health condition you can register with Melbourne Polytechnic Disability Support Services to develop a Disability Support Plan and access appropriate academic support.

You may also speak with teaching staff about other circumstances impacting your capacity to complete an evidence-based assessment and seek a reasonable adjustment. It is important to ensure the integrity of the assessment is maintained and the intent is not compromised. Reasonable adjustment may include but is not limited to extra time or extensions for assessments, an alternate assessment task, note-taking support or varying the venue.

### Safety

If for any reason you feel unsafe, you can stop participating in the assessment and inform your Assessor.

If at any time during the assessment process the Assessor considers that the safety of any person is at risk, they will **stop** the assessment immediately.

## Assessment Methods

The following assessments will be used to collect evidence of the knowledge and skills you have gained from your Learning Program. You will be required to demonstrate your ability to perform to the standard required in the workplace, as specified within the assessment task criteria as detailed below.

|  |  |  |
| --- | --- | --- |
| **Book** | **Task Number & Name** | **Method** |
| Assessment book  Practical | Assessment Task 1: **City Bookshop Database Proposal** | Written Assessment - Case Study |
| Assessment book  Knowledge | Assessment Task 2: **Knowledge Questions** | Written Questions |
| Assessment book  Practical | Assessment Task 3: **SQL Project** | Written Assessment - Assignment |

# Assessment Task 3: SQL Project

|  |  |
| --- | --- |
| Course code  and title | **ICT50120 Diploma of Information Technology** |
| Unit code and  title | **ICTWEB451 Apply structured query language in relational databases** |
| Due date | DD/MM/YYYY Week 8 |
| Resources required | Provided:   * ICTWEB451 Learner resources guide * Access to computer and internet * Microsoft Word, Microsoft Visio, M.S. SQL Server. |
| Decision making  rules | To achieve an overall satisfactory result for this assessment task:   * Learners must achieve a satisfactory result for each item in the Assessment   Checklist. |
| Learner instructions | * General Instructions   + This assessment must be completed by each learner, in the learners own time.   + The learners have three weeks to complete this task.   + The learner must read the case study and then address and complete Part 1 and to Part 2.   + At this time any learner who requires reasonable adjustments can discuss it with the assessors. It is important to ensure the integrity of the assessment is maintained and the intent is not compromised. E.g. extension of time, oral questions and answers etc.   + The learners must complete the two parts electronically using the space provided **within** the assessment document that they have downloaded from Moodle (**Database - Assessment Task 3.docx**) Please include student Full Name, Student ID and Group Name in the footer of the answer document   + Submit the completed file (**Database - Assessment Task 3.docx**) into the Assessment Task 3 folder in ICT50120 – Database – 2021 within the Moodle LMS site. * The learners must agree ( by clicking on the ‘I confirm radio button) with the assessment submission terms and condition in Moodle LMS prior to the submission |

**CITY BOOKSHOP DATABASE**

## Scenario:

You have recently accepted employment positions with the MP Tech Solution Company. One of the services provided by MP Tech Solutions to its clients includes providing a database solution.

You have been given a new assignment to work on a project entitled ‘City Bookshop Database. The project involves creating and populating the tables and generating reports as per client’s requirements.

The project manager of MP Tech has provided a set of procedure that need to be followed in the process of database development.

* Review the structure of the proposed database for City Bookshop
* Create the database as per the design structure provided
* Populate the tables along with the relationship based on the information provided
* Test and verify query results
* Obtain sign-off upon successfully meeting the requirements of the project

In the process of completing the City Bookshop Database the two main task includes:

1. Part 1 : Create and populate database
2. Part 2 : Test and verify query results

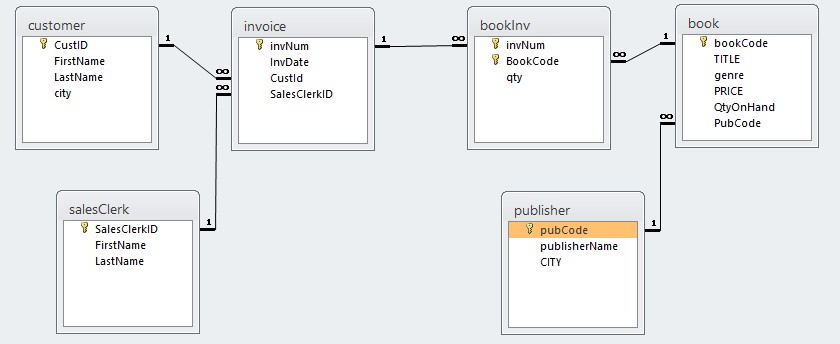
From this information, you are to:

## Part 1: Create and populate database

**Step 1**: Create a database entitled: **Book Database** using appropriate SQL queries within **M.S.SQL Server**. The database includes the following tables: **customer, invoice, bookInv, book, publisher** and **salesclerk**

**Step 2**: Based on the **relationship diagram** provided below determine the primary and foreign keys of each table for the **Book Database**.

**E.R. Diagram**



**Step 3**: Based on the **Data Dictionaries** provided below, you will have to structure the columns (fields) within each of the database tables for the **BookDatabase** in **M.S.SQL Server.** This structuring will be done using appropriate SQL queries on the relational database software.

#### Data Dictionaries

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: customer** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| **custID** | varchar | 6 | **Primary key** | Customer Id |
| **firstName** | varchar | 20 | Not null | Customer First Name |
| **lastName** | varchar | 20 | Not null | Customer Last Name |
| **city** | varchar | 15 | Not null | City |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: invoice** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| **invNum** | int | 8 | **Primary key** | Invoice Number |
| **invDate** | varchar | 10 | Not null | Invoice Date |
| **custId** | varchar | 6 | **Foreign key** | Customer Id |
| **salesClerkID** | int | 8 | **Foreign key** | Sales Clerk ID |

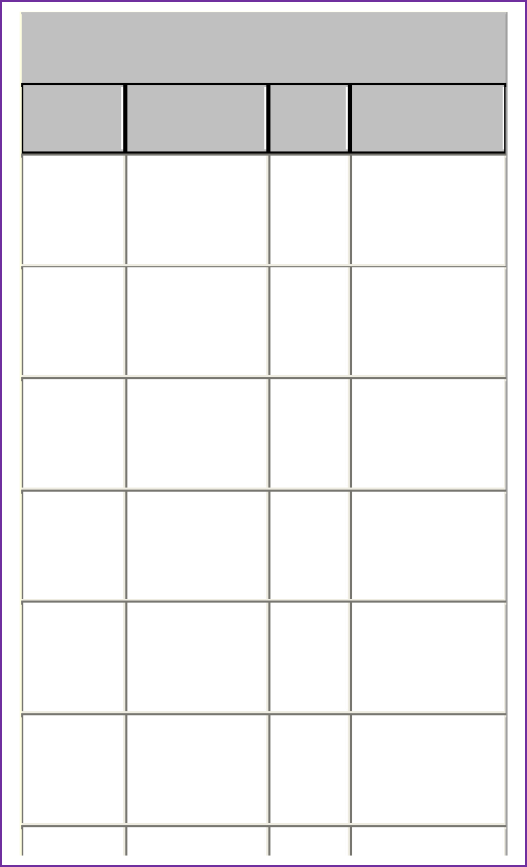
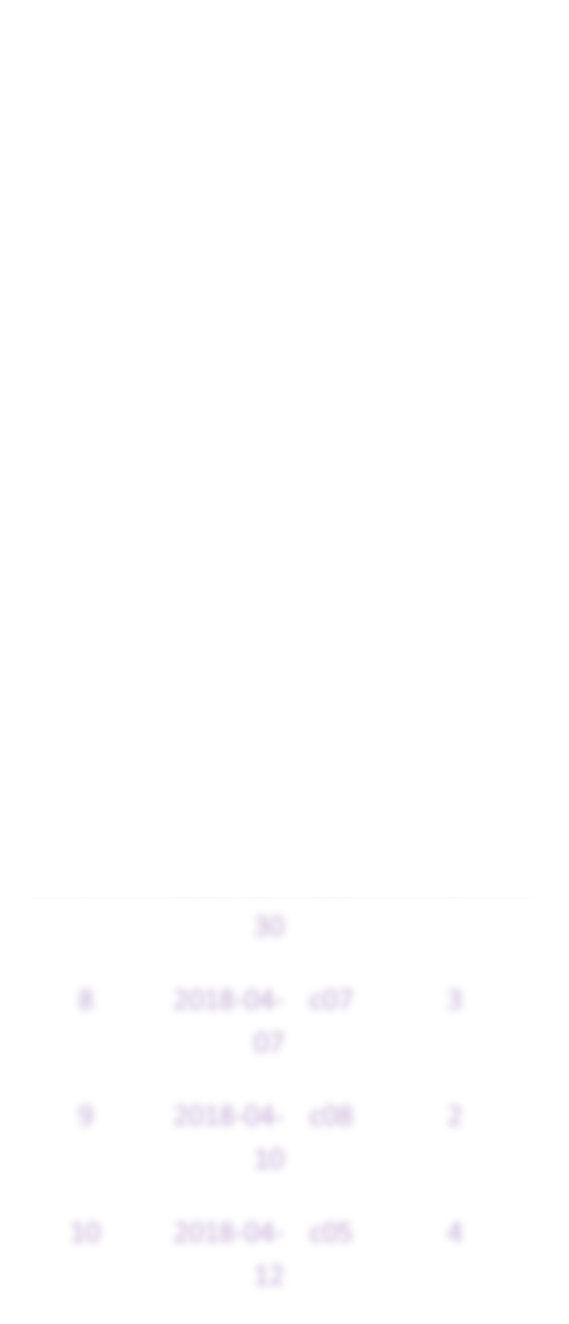
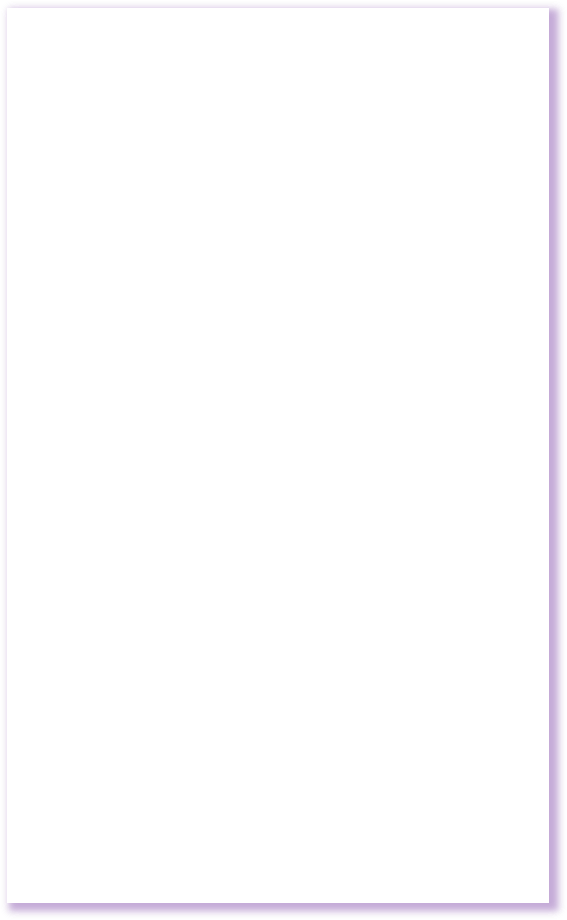
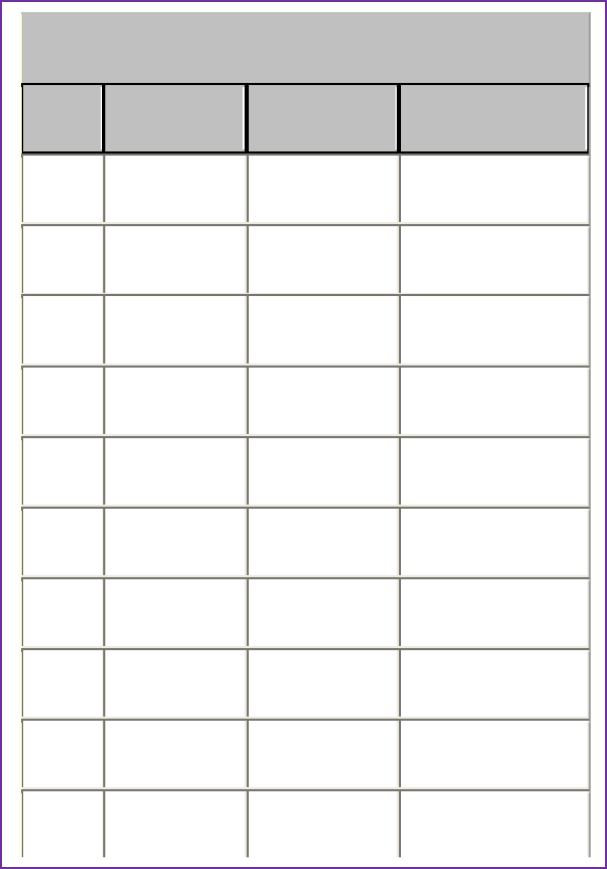
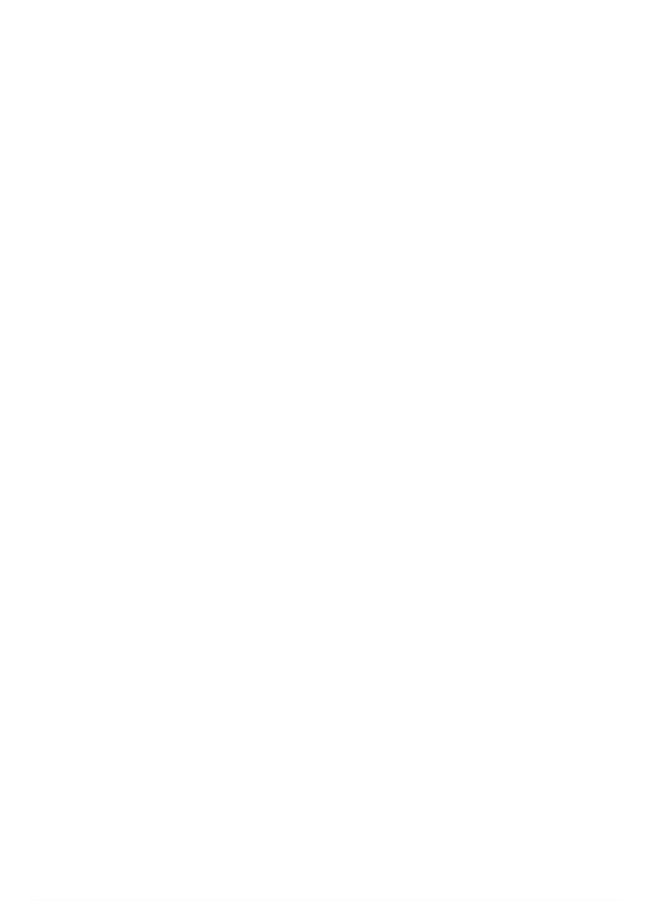
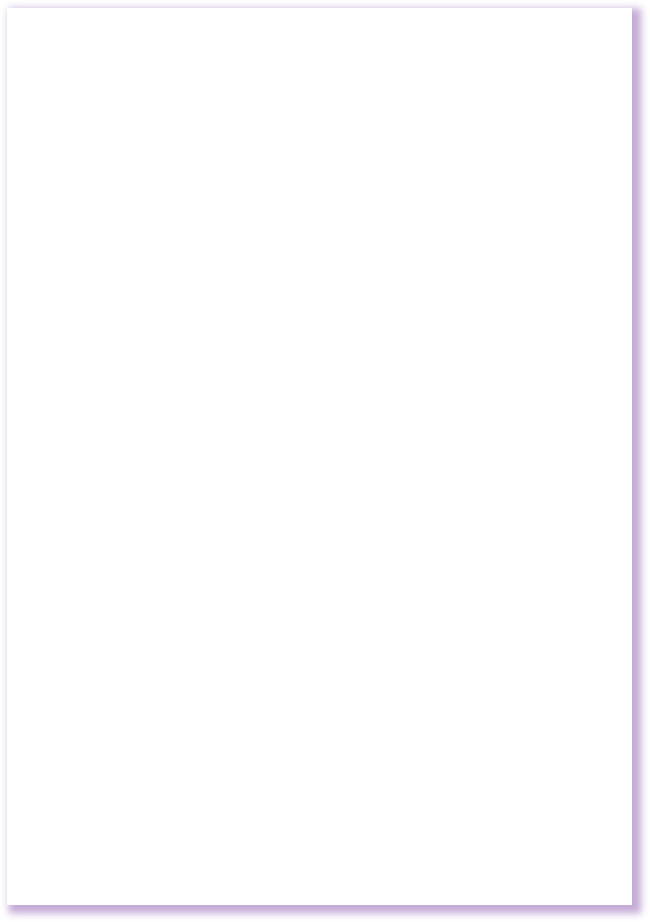
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: salesClerk** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| **salesClerkID** | int | 2 | **Primary key** | Sales Clerk ID |
| **FirstName** | varchar | 15 | Not null | Sales Clerk First Name |
| **LastName** | varchar | 15 | Not null | Sales Clerk Last Name |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: bookInv** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| invNum | int | 8 | **Primary key/Foreign Key** | Invoice Number |
| bookCode | int | 6 | **Primary key/Foreign Key** | Book Code |
| qty | int | 4 | Not null | Quantity Bought |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: book** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| **bookCode** | int | 6 | **Primary key** | Book Code |
| **title** | varchar | 60 | Not null | Book Title |
| **genre** | varchar | 6 | Not null | Genre / Type |
| **price** | decimal | 6,2 | Not null | Price of Book |
| **qtyOnHand** | int | 4 | Not null | Number of books in stock |
| **pubCode** | varchar | 5 | **Foreign key** | Publisher Code |

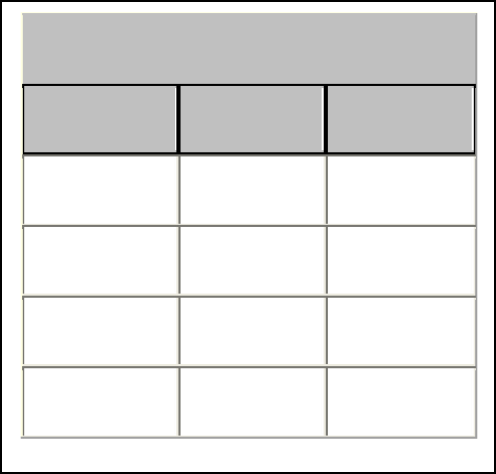
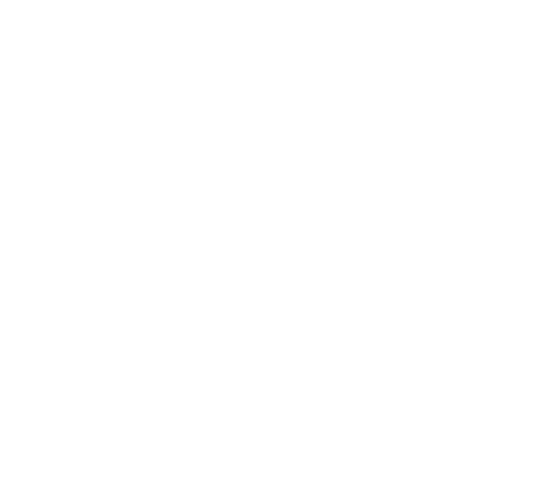
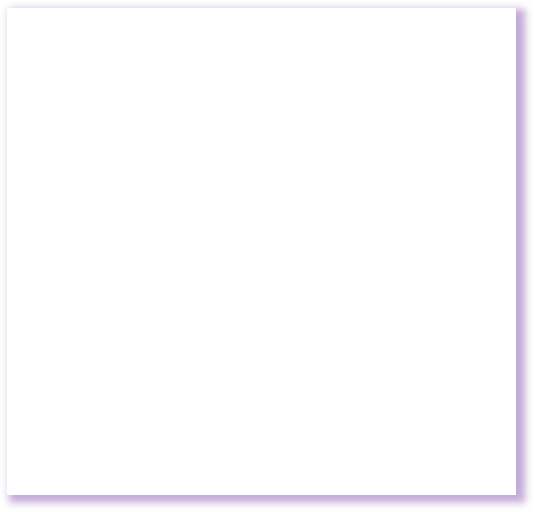
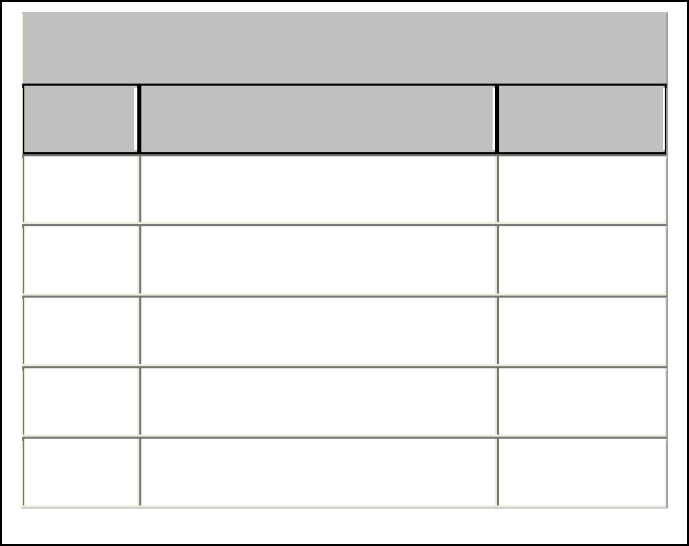
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: publisher** | | | | |
| **Field Name** | **Data Type** | **Field Length** | **Constraints** | **Description** |
| **pubCode** | varchar | 3 | **Primary key** | Client Id |
| **publisherName** | varchar | 25 | Not null | Client name |
| **city** | varchar | 20 | Not null | Client name |

**Step 4**: Based on the **Tables** provided below, you are to populate the tables using appropriate SQL queries within the **BookDatabase** with the data shown in the tables below**.**



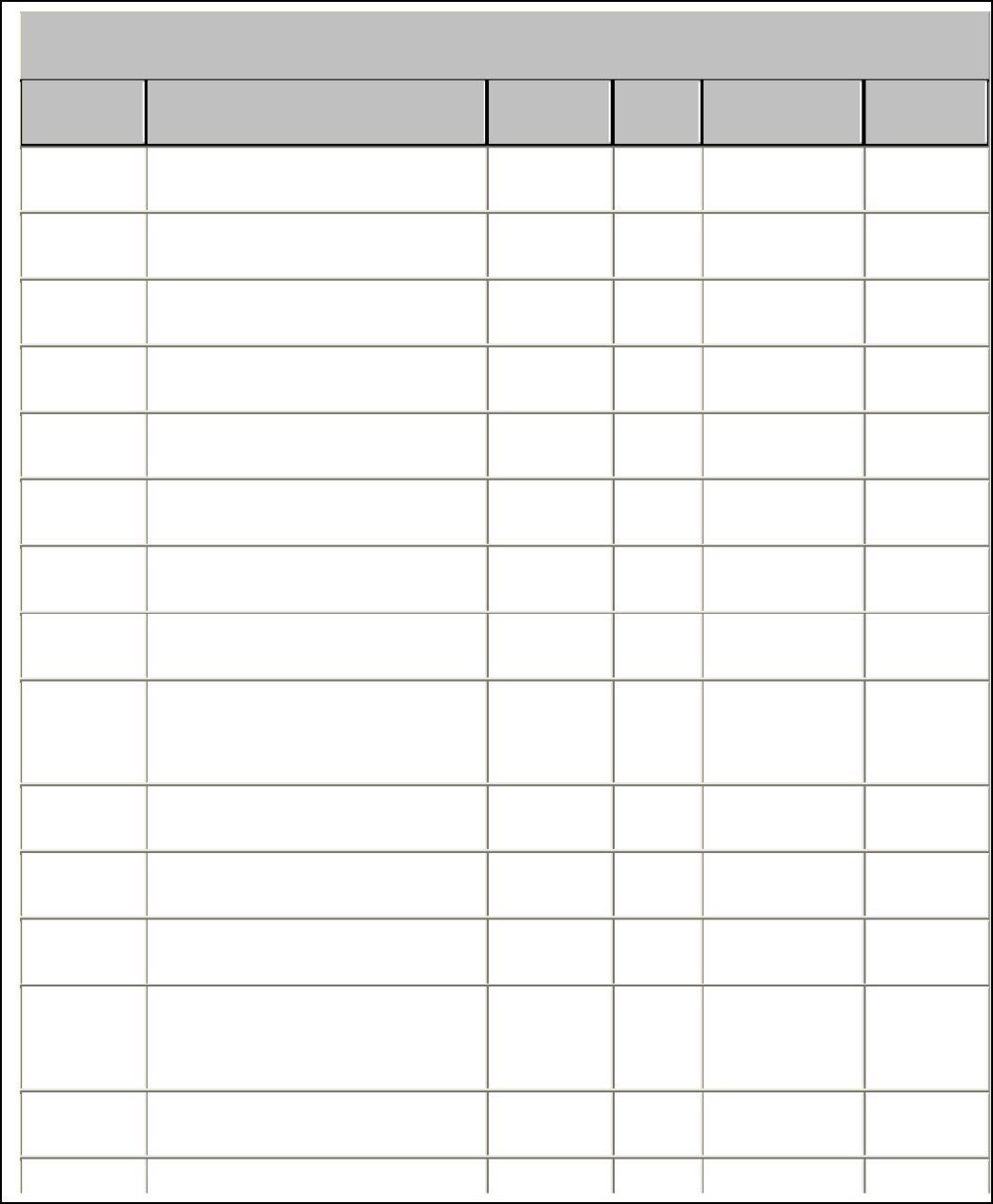
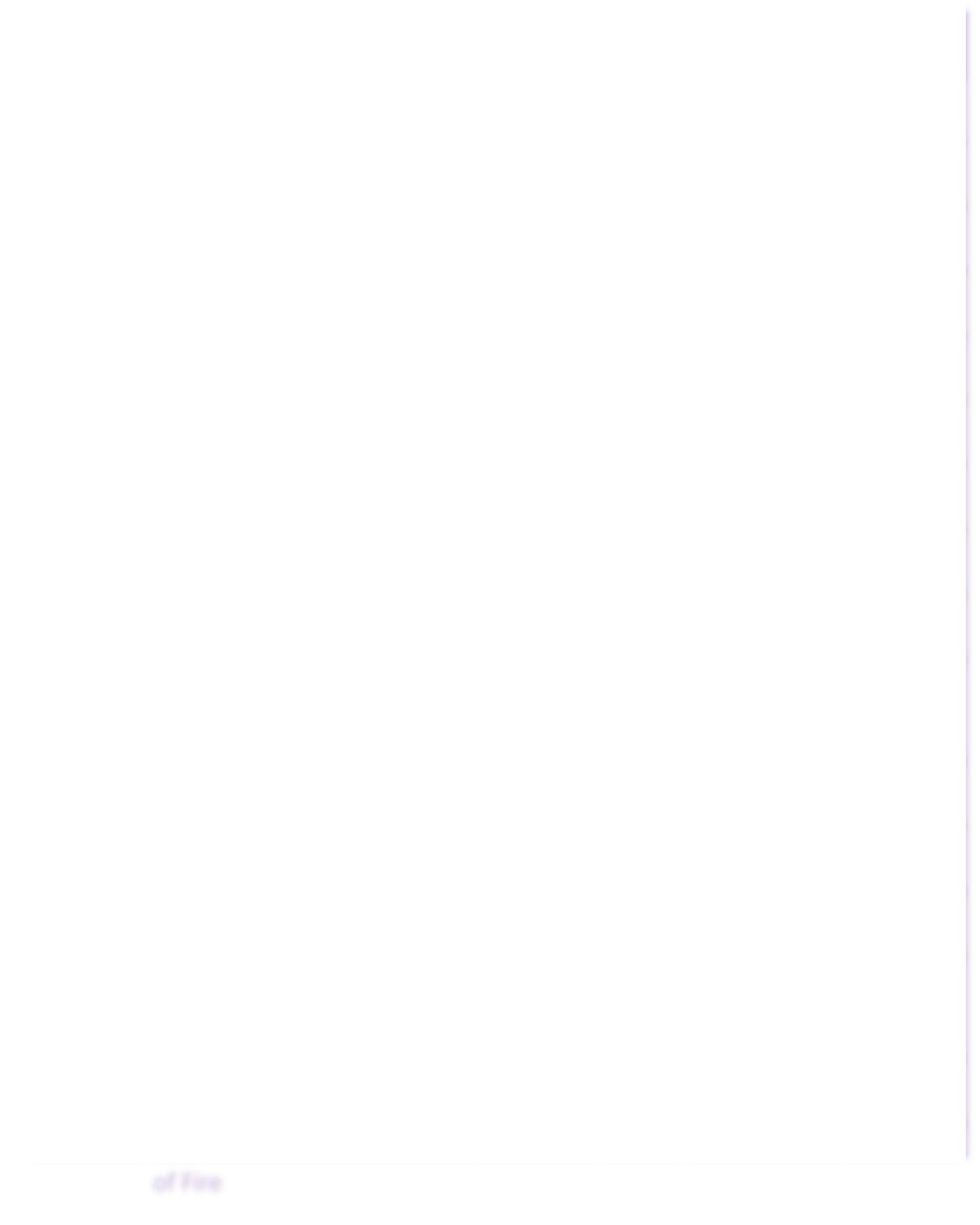
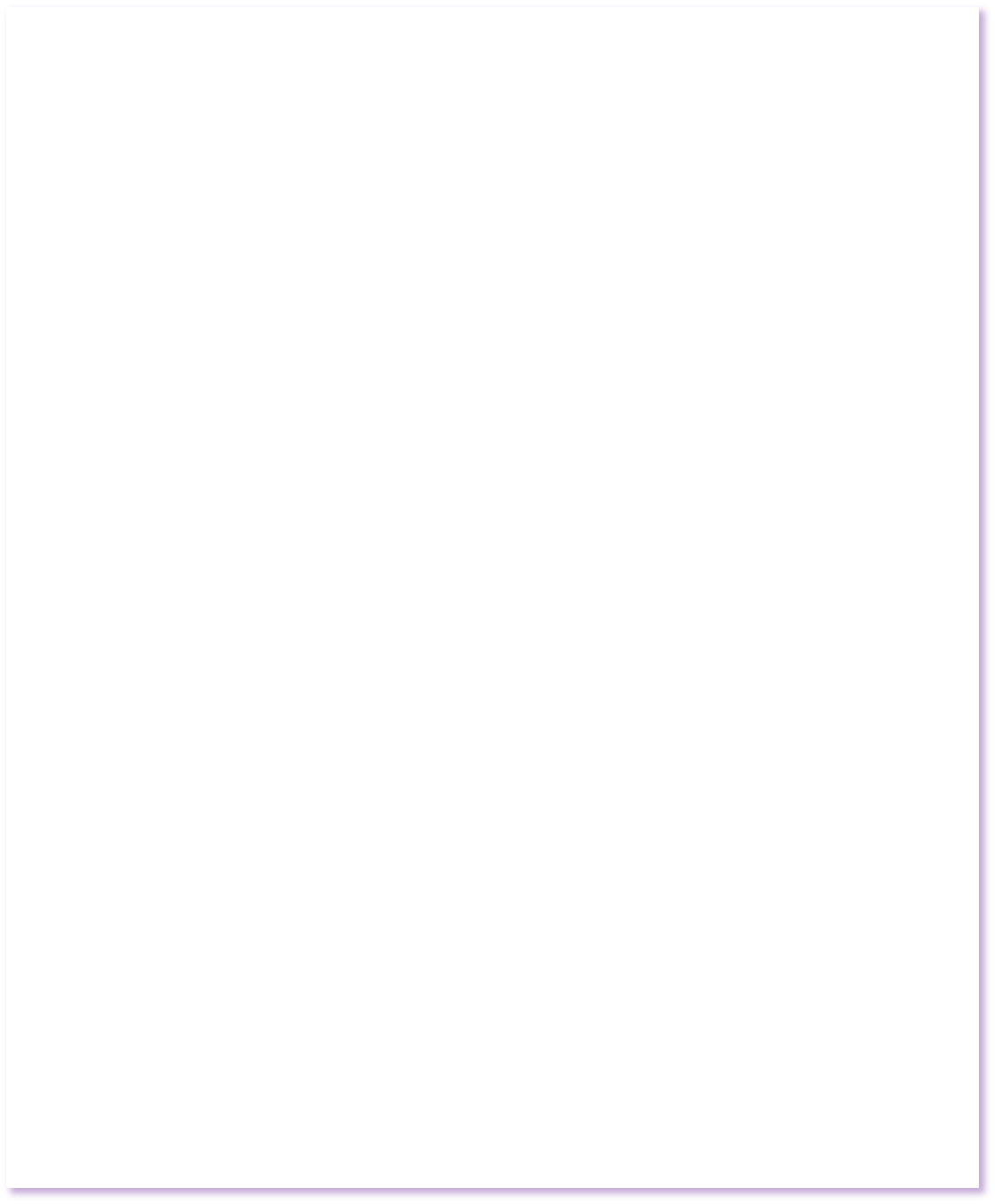
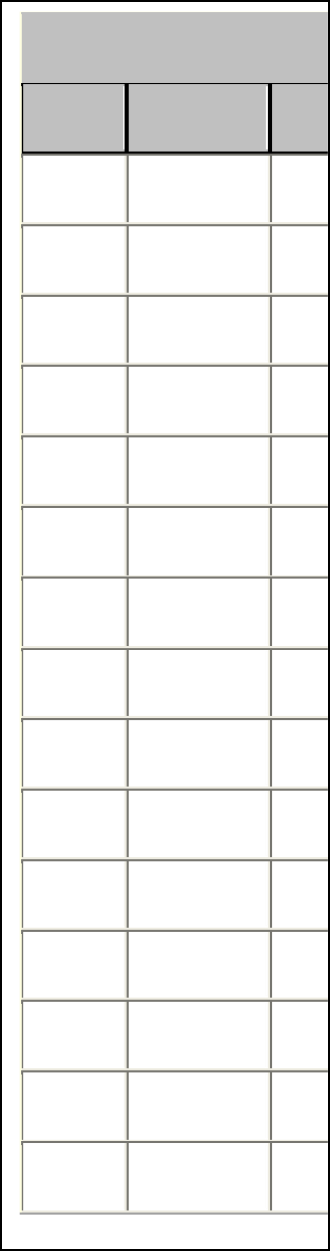
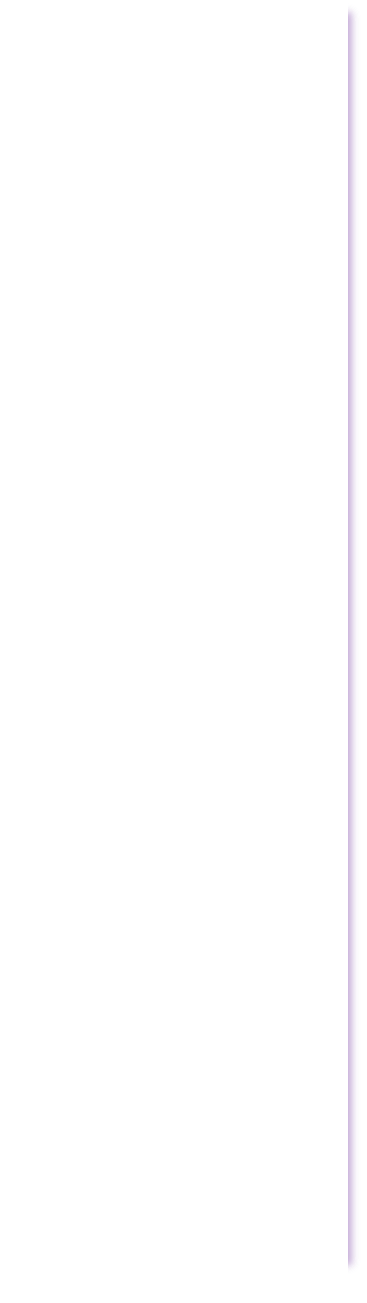
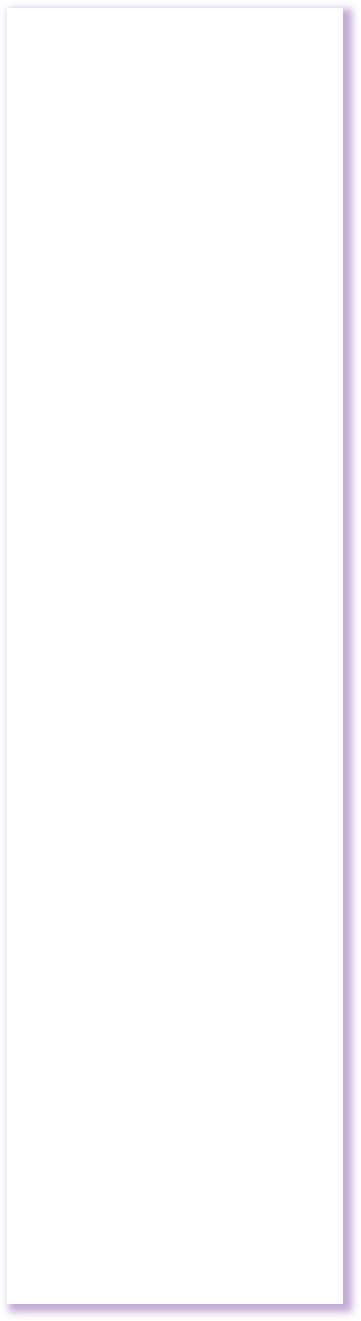
|  |  |  |  |
| --- | --- | --- | --- |
|  |  | customer |  |
| custID | firstName | lastName | city |
| c01 | Toni | Wu | Hampton |
| c02 | Paul | Chen | Carlton |
| c03 | Peter | Li | Springvale |
| c04 | Troy | Smith | Hampton |
| c05 | Peter | Straub | Toorak |
| c06 | Stephen | Zhoa | Oakliegh South |
| c07 | Philip | Pratt | Preston |
| c08 | Truddi | Chase | Werribee |
| c09 | Bradley | Jones | Geelong |
| c10 | Joseph | Yang | Lara |

|  |  |  |  |
| --- | --- | --- | --- |
| **invoice** | | | |
| **invNum** | **invDate** | **custId** | **salesClerkID** |
| 1 | 2018-01-  12 | c05 | 1 |
| 2 | 2018-02-  01 | c02 | 4 |
| 3 | 2018-02-  15 | c07 | 2 |
| 4 | 2018-03-  03 | c01 | 4 |
| 5 | 2018-03-  14 | c02 | 4 |
| 6 | 2018-03-  16 | c09 | 3 |
| 7 | 2018-03-  30 | c10 | 3 |
| 8 | 2018-04-  07 | c07 | 3 |
| 9 | 2018-04-  10 | c08 | 2 |
| 10 | 2018-04-  12 | c05 | 4 |



|  |  |  |
| --- | --- | --- |
|  | **publisher** |  |
| **pubCode** | **publisherName** | **city** |
| AP | Arcade Publishing | New York |
| CT | Course Technology | Boston |
| MP | McPherson & Co. | Kingston |
| PE | Penguin USA | Sauk City |
| RH | Random House | New York |

|  |  |  |
| --- | --- | --- |
|  | **salesClerk** |  |
| **salesClerkID** | **firstName** | **lastName** |
| 1 | Sally | Huang |
| 2 | Bob | Smith |
| 3 | Joanne | Yang |
| 4 | Amit | Hussien |



|  |  |  |
| --- | --- | --- |
| **bookInv** | | |
| **invNum** | **bookCode** | **qty** |
| 1 | 3350 | 1 |
| 1 | 2766 | 2 |
| 2 | 180 | 1 |
| 3 | 808 | 1 |
| 4 | 7443 | 3 |
| 5 | 3906 | 1 |
| 6 | 378 | 1 |
| 6 | 2908 | 5 |
| 7 | 5163 | 2 |
| 8 | 7249 | 1 |
| 9 | 3906 | 2 |
| 9 | 6328 | 1 |
| 10 | 378 | 1 |
| 10 | 3350 | 2 |
| 10 | 7405 | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **book** | | | | | |
| **bookCode** | **title** | **genre** | **price** | **qtyOnHand** | **pubCode** |
| 180 | A Deepness in the Sky | SFI | 7.19 | 10 | RH |
| 200 | The Stranger | FIC | 8.00 | 12 | AP |
| 378 | Venice | ART | 24.50 | 10 | MP |
| 808 | The Edge | MYS | 6.99 | 25 | AP |
| 1351 | Dreamcatcher: A Novel | HOR | 19.60 | 55 | RH |
| 1382 | Treasure Chests | ART | 24.46 | 15 | PE |
| 2766 | Of Mice and Men | FIC | 6.95 | 23 | RH |
| 2908 | Electric Light | POE | 14.00 | 26 | RH |
| 3350 | Group: Six People in Search of a Life | PSY | 10.40 | 12 | CT |
| 3906 | The Soul of a New Machine | SCI | 11.16 | 32 | AP |
| 5163 | Travels with Charley | TRA | 7.95 | 29 | PE |
| 6328 | Band of Brothers | HIS | 9.60 | 56 | MP |
| 7249 | One flew over the Chooks Nest | FIC | 17.00 | 17 | PE |
| 7405 | East of Eden | FIC | 12.95 | 49 | CT |
| 7443 | Harry Potter and the Goblet  of Fire | SFI | 18.16 | 78 | RH |

**Step 5**: Upon the completion of the database and its tables, copy and paste the code from Step 1 to Step 4 into the answer section provided below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **What is the SQL Code that will create the database for Part 1** | | | | |  |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY | |
| -- Assessment Task 3 Database Code  -- Drop BookDatabase if it's already exists  DROP DATABASE IF EXISTS BookDatabase;  CREATE DATABASE BookDatabase;  -- select BookDatabase  USE BookDatabase;  -- `customers` Table  DROP TABLE IF EXISTS customers;  CREATE TABLE customers (  custID varchar(6) NOT NULL Primary Key,  firstName varchar(20) NOT NULL,  lastName varchar(20) NOT NULL,  city varchar(15) NOT NULL  );  -- `Customers` Table Data  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c01','Toni','Wu','Hampton');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c02','Paul','Chen','Carlton');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c03','Peter','Li','Springvale');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c04','Troy','Smith','Hampton');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c05','Peter','Straub','Toorak');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c06','Stephen','Zhoa','Oakliegh South');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c07','Philip','Pratt', 'Preston');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c08','Truddi','Chase', 'Werribee');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c09','Bradley','Jones', 'Geelong');  insert into `customers`(`custID`,`firstName`,`lastName`,`city`) values('c10','Joseph','Yang', 'Lara');  -- `invoice`Table  DROP TABLE IF EXISTS `invoice`;  CREATE TABLE `invoice` (  `invNum` int(8) NOT NULL PRIMARY KEY,  `invDate` varchar(10) NOT NULL,  `custId` varchar(5) NOT NULL,  `salesClerkID` int(8) NOT NULL,  FOREIGN KEY (custId) REFERENCES Customers(custId),  FOREIGN KEY (salesClerkID) REFERENCES SalesClerk(salesClerkID)  ) ;  -- `invoice` Table Data  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (1,'2018-01-12','c05',1);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (2,'2018-02-01','c02',4);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (3,'2018-02-15','c07',2);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (4,'2018-03-03','c01',4);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (5,'2018-03-14','c02',4);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (6,'2018-03-16','c09',3);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (7,'2018-03-30','c10',3);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (8,'2018-04-07','c07',3);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (9,'2018-04-10','c08',2);  insert into `invoice`(`invNum`,`invDate`,`custId`,`salesClerkID`) values (10,'2018-04-12','c05',4);  -- `salesClerk` Table    DROP TABLE IF EXISTS `salesClerk`;  CREATE TABLE `salesClerk` (  `salesClerkID` int(2) NOT NULL PRIMARY KEY,  `firstName` varchar(15) NOT NULL,  `lastName` varchar(15) NOT NULL  ) ;  -- `salesClerk` Table Data  insert into `salesClerk`(`salesClerkID`,`firstName`,`lastName`) values(1,'Sally','Huang');  insert into `salesClerk`(`salesClerkID`,`firstName`,`lastName`) values(2,'Bob','Smith');  insert into `salesClerk`(`salesClerkID`,`firstName`,`lastName`) values(3,'Joanne','Yang');  insert into `salesClerk`(`salesClerkID`,`firstName`,`lastName`) values(4,'Amit','Hussien');  -- `bookInv` Table  DROP TABLE IF EXISTS `bookInv`;  CREATE TABLE `bookInv` (  `invNum` int(8) NOT NULL,  `bookCode` int(6) NOT NULL,  `qty` int(4) NOT NULL,  PRIMARY KEY (`invNum`,`bookCode`),  FOREIGN KEY (`invNum`) REFERENCES `invoice` (`invNum`),  FOREIGN KEY (`bookCode`) REFERENCES `books` (`bookCode`)  );  -- `bookInv` Table Data  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(1,3350,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(1,2766,2);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(2,180,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(3,808,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(4,7443,3);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(5,3906,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(6,378,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(6,2908,5);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(7,5163,2);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(7,7249,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(9,3906,2);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(9,6328,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(10,378,1);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(10,3350,2);  insert into `bookInv`(`invNum`,`bookCode`, `qty`) values(10,7405,1);  -- `books`Table  DROP TABLE IF EXISTS `books`;  CREATE TABLE `books` (  `bookCode` int(6) NOT NULL Primary Key,  `title` varchar(60) NOT NULL,  `genre` varchar(6) NOT NULL,  `price` decimal(6,2) NOT NULL,  `qtyOnHand` int(4) NOT NULL,  `pubCode` varchar(5) NOT NULL,  FOREIGN KEY (`pubCode`) REFERENCES `publisher` (`pubCode`)  );  -- `books` Table Data  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('180','A Deepness in the Sky','SFI',7.19,10,'RH');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('200','The Stranger','Fic',8.00,12,'AP');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('378','Venice','ART',24.50,10,'MP');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('808','The Edge','MYS',6.99,25,'AP');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('1351','Dreamcatcher: A Novel','HOR',19.60,55,'RH');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('1382','Treasure Chests','ART',24.46,15,'PE');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('2766','Of Mice and Men','FIC',6.95,23,'RH');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('2908','Electric Light','POE',14.00,26,'RH');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('3350','Group: Six People in Search of a Life','PSY',10.40,12,'CT');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('3906','The Soul of a New Machine','SFI',11.16,32,'PL');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('5163','Travels with Charley','TRA',7.95,29,'TB');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('6328','Band of Brothers','HIS',9.60,56,'MP');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('7249','One flew over the Chooks Nest','FIC',17.00,17,'BY');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('7405','East of Eden','FIC',12.95,49,'CT');  insert into `books`(`bookCode`,`title`,`genre`,`price`,`qtyOnHand`,`pubCode`)  values('7443','Harry Potter and the Goblet of Fire','SFI',18.16,78,'JP');  -- `publisher` Table  DROP TABLE IF EXISTS `publisher`;  CREATE TABLE `publisher` (  `pubCode` varchar(3) NOT NULL PRIMARY KEY,  `publisherName` varchar(25) NOT NULL,  `city` varchar(20) NOT NULL  ) ;  -- `publisher` Table Data    insert into `publisher`(`pubCode`,`publisherName`,`city`) values('AP','Arcade Publishing','New York');  insert into `publisher`(`pubCode`,`publisherName`,`city`) values('CT','Course Technology','Boston');  insert into `publisher`(`pubCode`,`publisherName`,`city`) values('MP','McPherson & Co','Kingston');  insert into `publisher`(`pubCode`,`publisherName`,`city`) values('PE','Penguin','Sauk City');  insert into `publisher`(`pubCode`,`publisherName`,`city`) values('RH','Random House','New York'); | | | | | | |

## Part 2: Test and verify query results

For each of the questions below, use the following steps to test and verify your query results.

Step 1: Review the required information from the questions. Construct the test data (SQL statement) and the expected results.

Step 2: Execute the SQL statement according to the task.

Step 3: Test and verify the result of the executed SQL statement against the expected results. Document your findings.

Step 4: Make the necessary changes to ensure the task requirements are met and obtain sign off.

Below is an example on how to answer the questions.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Example of Question:**  **Q15** | **List the Publishers Name and city for all the publishers residing in New York.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT  SATISFACTORY |

#### Test Data and Results Tables

**Answer ☐ SATISFACTORY**

**NOT SATISFACTORY**

**Pass** means the expected results are the same as the SQL Query results

**Pass/ Fail**

**SQL Query (Test) Results**

**Expected Result**

**Test Data**

**Test Case Name**

**Test Case Num.**

**Date :**

**Tester:**

**Example For Student**

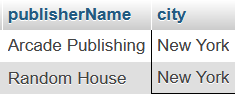
**Project Name: Assessment 3**

**☐**

**Fail** means the expected results does not match the SQL Query results or vice versa

|  |  |
| --- | --- |
| **publisherName** | **city** |
| Arcade  Publishing | New York |
| Random House | New York |

**Comment**: **The green highlighted row** is required if the outcome of a particular test case number is a **fail**. The process of recording information in the respective columns for that row needs to be repeated until the outcome is a **pass.**



**Pass**

**SELECT**

**publisherName,city FROM publisher**

**WHERE city = 'New York'**

**List the Publishers Name and city for all the publishers residing in New York.**

**15**

Page 31 of 47

Melbourne Polytechnic

|  |  |  |
| --- | --- | --- |
| **invNum** | **invDate** | **salesClerkID** |
| 6 | 2018-03-16 | 3 |
| 7 | 2018-03-30 | 3 |
| 8 | 2018-04-07 | 3 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1** | **For sales clerk id 3, list all the columns from Invoice excluding the customer Id column.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **1** | **For sales clerk id 3, list all the columns from Invoice excluding the customer Id column.** | SELECT invNum,invDate,salesClerkID  FROM invoice  WHERE salesClerkID = 3 |  | |  | | pass |

Page 32 of 47

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **2** | **Display the title, price and genre of the books that has the price between 8 and 10 dollars. Sort it by the genre.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | | | | |
| **Tester:** | | | | | | | | **Date :** |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | | | **SQL Query (Test) Results** | | | **Pass/ Fail** |
| **2** | **Display the title, price and genre of the books, which has the price between 8 and**  **10 dollars. Sort it by the genre.** | SELECT title,price,genre  FROM books  WHERE price BETWEEN 8.00 AND 10.00  ORDER BY genre; |  | | | |  | | | pass |
|  | **title** | **price** | **genre** |
| The Stranger | 8.00 | FIC |
| Band of Brothers | 9.60 | HIS |
|  | | | |

|  |  |
| --- | --- |
| **genre** | **NumberOfBooks** |
| ART | 25 |
| Fic | 101 |
| HIS | 56 |
| HOR | 55 |
| MYS | 25 |
| POE | 26 |
| PSY | 12 |
| SFI | 120 |
| TRA | 29 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **3** | **List each genre once and the total of how many books there are on hand in the store for each of the genre.**  **(Hint: use Group by and sum)** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test**  **Case Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **3** | **List each genre once and the total of how many books there are on hand in the store for each of the genre.**  **(Hint: use Group by and sum)** | SELECT genre,SUM(qtyOnHand) AS NumberOfbooks  FROM books  GROUP BY genre; |  | |  | | pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **4** | **List the publisher name and their city who has offices in New York, Kingston and Sauk City. Do this using the IN operator. Sort by City in alphabetical order.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | | **Date :** |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | **SQL Query (Test) Results** | | | **Pass/ Fail** |
| **4** | **List the publisher name and their city who has offices in New York, Kingston and Sauk City. Do this using the IN operator. Sort by City in alphabetical order.** | SELECT publisherName,city  FROM publisher  WHERE city IN( 'New York','Kingston','Sauk City')  ORDER BY city; |  |  | | | pass |

Page 35 of 47

|  |  |
| --- | --- |
| **publisherName** | **city** |
| McPherson&Co | KingSton |
| Arcade Publishing | New York |
| Random House | New York |
| Penguin | Sauk City |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **5** | **Display the first name, last name and invoice date for customers who received invoices before 2018-03-16.**  **Note: You need to use an inner join in this query and that dates must be enclosed in single quotes within SQL statements.** | | | | | |
|  |  | | | | |
|  | ANSWER | | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **5** | **Display the first name, last name and invoice date for customers who received invoices before 2018-03-16. Note: You need to use an inner join in this query and that dates must be enclosed in single quotes within SQL statements.** | SELECT firstName,lastName,invDate  FROM customers  INNER JOIN invoice ON customers.custID=invoice.custId  WHERE invDate<'2018-03-16'; |  | |  | | Pass |

Page 36 of 47

|  |  |  |
| --- | --- | --- |
| **firstName** | **lastName** | **invDate** |
| Peter | Straub | 2018-01—12 |
| Paul | Chen | 2018-02-01 |
| Philip | Pratt | 2018-02-15 |
| Toni | Wu | 2018-03-03 |
| Paul | Chen | 2018-03-14 |

Melbourne Polytechnic

|  |  |  |  |
| --- | --- | --- | --- |
| **firstName** | **lastName** | **title** | **qty** |
| Bradley | Jones | Electric Light | 5 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **6** | **List the first name and last name of customers who have purchased a Poetry book ( i.e. genre: ‘POE’) and also the title of the book and what quantity (i.e. qty) they purchased.**  **Note : You need to use inner joins in this query** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test**  **Case Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **6** | **List the first name and last name of customers who have purchased a Poetry book ( i.e. genre: ‘POE’) and also the title of the book and what quantity (i.e. qty) they purchased. Note : You need to use inner joins in this query** | SELECT firstName,lastName,books.title,bookinv.qty  FROM customers  INNER JOIN invoice ON customers.custID = invoice.custId  INNER JOIN bookinv ON invoice.invNum = bookinv.invNum  INNER JOIN books ON books.bookCode = bookinv.bookCode  WHERE books.genre = 'POE'; |  | |  | | Pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **7** | **List the customer ID, book title and total spent on each purchase for each customer where the total spent is more than 60 dollars.**  ***(Hint: You will need a calculated field in this sql statement )* Note: You need to use inner joins in this query.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **7** | **List the customer ID, book title and total spent on each purchase for each customer where the total spent is more than 60 dollars. *(Hint: You will need a calculated field in this sql statement )* Note: You need to use inner joins in this query.** | FROM customers  INNER JOIN invoice ON customers.custID = invoice.custId  INNER JOIN bookinv ON invoice.invNum = bookinv.invNum  INNER JOIN books ON books.bookCode = bookinv.bookCode  WHERE bookinv.qty \*books.price>60; | **custID title TotalSpent**  C09 Electric Light 70.00 |  | Pass |

|  |  |  |
| --- | --- | --- |
| **firstName** | **lastName** | **TotalSpent** |
| Toni | WU | 54.48 |
| Paul | Chen | 18.35 |
| Peter | Straub | 82.5.5 |
| Philip | Pratt | 6.99 |
| Truddi | Chase | 31.92 |
| Bradley | Jones | 94.50 |
| Joseph | Yang | 32.90 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **8** | **List the customer’s first name, last name and total spent by each of the customers on books from the store. ( *Hint: You will need to use a group by and an aggregate function*) Note : You need to use inner joins in this query** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **8** | **List the customer’s first name, last name and total spent by each of the customers on books from the store. ( *Hint: You will need to use a group by and an aggregate function*) Note : You need to use inner joins in this query** | SELECT  firstName,  lastName,  SUM(bookinv.qty \* books.price) AS TotalSpent  FROM  customers  INNER JOIN  invoice ON customers.custID = invoice.custId  INNER JOIN  bookinv ON invoice.invNum = bookinv.invNum  INNER JOIN  books ON bookinv.bookCode = books.bookCode  GROUP BY  customers.custID; |  | |  | | Pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **9** | **Modify the previous query to show the grand total spent by all customers in the book store.** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **9** | **Modify the previous query to show the grand total spent by all customers in the book store.** | SELECT SUM(bookinv.qty\*books.price) AS TotalSpent  FROM bookinv  INNER JOIN books ON books.bookCode = bookinv.bookCode; | **TotalSpent**  321.69 | |  | | Pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **10** | **Write the SQL command to delete the book record titled “East of Eden” from the book Table.**  **(Hint: use Delete)** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **10** | **Write the SQL command to delete the book record titled “East of Eden” from the book Table.**  **(Hint: use Delete)** | DELETE  FROM books  WHERE title = 'East of Eden'; | **The'East of Eden’ entry is deleted**  **From the Books table in the Data base** | | **The'East of Eden’ entry is deleted From the Books table in the Data base** | | Pass |

|  |  |
| --- | --- |
| **Title** | **Price** |
| Venice | 25.00 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **11** | **Write the SQL command to change the price, of the book titled “Venice” to 25.00 dollars.**  **(Hint: use Update)** | | | | |
|  | ANSWER | **☐** | SATISFACTORY | **☐** | NOT SATISFACTORY |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name: Assessment 3** | | | | | | | |
| **Tester:** | | | | **Date :** | |  | |
| **Test Case**  **Num.** | **Test Case Name** | **Test Data** | **Expected Result** | | **SQL Query (Test) Results** | | **Pass/ Fail** |
| **11** | **Write the SQL command to change the price, of the book titled “Venice” to**  **25.00 dollars. (Hint: use Update)** | UPDATE books  SET price = 25.00  WHERE title = 'Venice' |  | |  | | Pass |

Page 43 of 47

|  |  |  |
| --- | --- | --- |
| **Sign off for project closure:** | | |
|  | | |
| **Tester Name** | **Test Signature** | **Date** |
| **Student Name** | **Student Signature** |  |
| **Supervisor Name** | **Supervisor Signature** | **Date** |
| Graham Gleich | Graham Gleich |  |

## Assessment Checklist 3: Part 1&2 Database Assessment Task 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Learner name** | |  | **Student ID** |  | |
| **Assessor name** | |  | **Date** |  | |
| ASSESSMENT CHECKLIST  **ASSESSOR TO COMPLETE THE FOLLOWING** | | | | | |
| **THE LEARNER:** | | | **SATISFACTORY** | | **NOT SATISFACTORY** |
| Part 1 | | |  | |  |
| 1 | The learner has constructed the appropriate code to create the populated book store database. | | ☐ | | ☐ |
| Part 2 | | |  | |  |
| 3 | The learner has correctly answered all of the SQL questions | | ☐ | | ☐ |
| **Feedback -** Assessor must include feedback. | | | | | |
|  | | | | | |

Learner Declaration

(only required for hard copy submissions)

|  |  |  |  |
| --- | --- | --- | --- |
| PLEASE READ, TICK AND SIGN BELOW | | | |
| * I declare that the attached assessment I have submitted is my own original work and any contributions from and references to other authors are clearly acknowledged and noted. * This document has been created for the purpose of this assessment only and has not been submitted as another form of assessment at Melbourne Polytechnic or any other tertiary institute. * I have retained a copy of this work for my reference in the event that this application is lost or damaged. * I give permission for Melbourne Polytechnic to keep, make copies of and communicate my work for the purpose of investigating plagiarism and/or review by internal and external assessors. * I understand that plagiarism is the act of using another person’s idea or work and presenting it as my own. This is a serious offence and I will accept that penalties will be imposed on me should I breach Melbourne Polytechnic’s plagiarism policy. | | | |
| LEARNER SIGNATURE | X | DATE |  |
| PLEASE NOTE THAT YOUR ASSESSMENT WILL NOT BE ACCEPTED UNLESS YOU HAVE:   * COMPLETED ALL SECTIONS OF THE ASSESSMENT * ACKNOWLEDGED ALL SOURCES OF OTHER PEOPLE’S CONTRIBUTIONS INCLUDING REFERENCES AND LEARNERS’ NAMES FOR GROUP WORK ASSESSMENTS * COMPLETED ALL AREAS OF THIS LEARNER ASSESSMENT COVER SHEET. | | | |

## Assessment Task Summary – Part 1&2 Database Assessment Task 3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TRAINER/ASSESSOR TO COMPLETE THE FOLLOWING:  **THE LEARNER:** | | | | | | YES | NO |
| 1 | Satisfactorily completed all items in Part 1 | | | | | ☐ | ☐ |
| 2 | Satisfactorily completed all items in Part 2 | | | | | ☐ | ☐ |
| FEEDBACK **-** Assessor must include feedback | | | | | | | |
|  | | | | | | | |
| OVERALL TASK RESULT | | | | | | | |
| * Satisfactory * Not Satisfactory (resubmission required) – Due date: | | | | | | | |
| DATE ASSESSMENT RETURNED | | |  | | | | |
| TRAINER/ASSESSOR NAME | | |  | | | | |
| TRAINER/ASSESSOR SIGNATURE | | | X | | | | |
| **LEARNER DECLARATION**: Please read, tick and sign below | | | | | | | |
| * I, have been advised of the outcome of this assessment task.   PRINT NAME | | | | | | | |
| LEARNER SIGNATURE | | WangYiZhuo | | DATE | 2024．1．4 | | |