COMP3322B Modern Technologies on World Wide Web

Project- An Online Bookstore

Total 20 points

Overview:

In this project, you are going to design and implement an online bookstore for customers to purchase books according to their preferences. In the project, you will implement the website using different techniques which include both the client side and the server side. The website consists of a few web pages for collecting users' input on the client-side and passing the data to the server-side for processing. You can build your own database for storing data, but we have also provided one to you. Note: if you are interested in designing your own database schema, you are welcome to do so.

Objectives:

1. A learning activity to support ILO 2.
2. The goals of this programming project are:
   - to get solid experience in using client-side, server-side techniques and database system to design and implement a web-based application which contains some common features of an online bookstore;
   - to get a good understanding of how the layout of a web-based application can be implemented by using CSS and responsive design.

Specifications:

This project consists of two parts:

- Part I- Implement the functionalities of the system, which mainly contains the following parts: the main page, the login/logout page, the create account page, the refine by category function, the sort by Price (Highest) function, the search bar function, the cart function, the checkout page, and the invoice page. It is expected that the abovementioned pages and functions are created by the client-side and the server-side technologies and you can use any client-side and server-side technologies that were learnt in this course. **Please note that in the examples shown in this specification, HTML, CSS, JavaScript, jQuery, MySQL, AJAX, and PHP are used as the client and server side technologies but you are welcome to use other technologies which were learnt from this course.**
- Part II- Design the layout of the web pages by using CSS and responsive web design. It is expected that in this part, your website should have the feature of responsive web and modern CSS design.

Part I (Total: 16 points)

You are expected to build a login system which enables the functionalities of accepting user login ('Login') and creating a new user account ('Create Account').

On the Login page, it is expected the following items and functions are included (**0.5 points**):

- [Login] form – A login form which uses the method of "POST" to send the data that is entered by the user in the username and password textboxes. It is expected that the data will be passed to the server (e.g. a php file called "verifyLogin") which will be implemented later. Besides, upon submitting the form, it should be able to check whether the username and password textboxes are empty; if yes, an alert message (e.g., popup or alert box) showing "Please do not leave the fields empty" will be displayed in the page. You should also include the name of the bookstore at the top of the page (The name is designed by you).

- [Username] textbox- A textbox which allows the user to enter his/her account name with a placeholder "Username".

- [Password] textbox- A textbox which allows user to enter his/her account password with a placeholder "Password". The password should not be visible to others when the user is entering the password.

- [Submit] button- A submit button for sending the form data (the username and password) to the server (e.g. a php file called "verifyLogin") once it is clicked. The button should have a display name "SUBMIT".

- [Create account] button- A button which when clicked, will direct to the Create account page (which will be implemented later). The button should have a display name "CREATE".

Figure 1 shows the sample of the index page. Note that in Figure 1, in addition to the above features, some other features such as search bar, shopping cart, etc. are included, and some CSS styling rules are applied to the page, but you DO NOT need to implement those functions and the CSS **in this stage**, implementing the basic skeleton as stated above is fine.

Fig. 1 Sample of the index page (you only need to implement the basic skeleton as stated in the previous page at the current stage)

On the Create account page, it is expected the following items and functions are included (**0.5 points**):
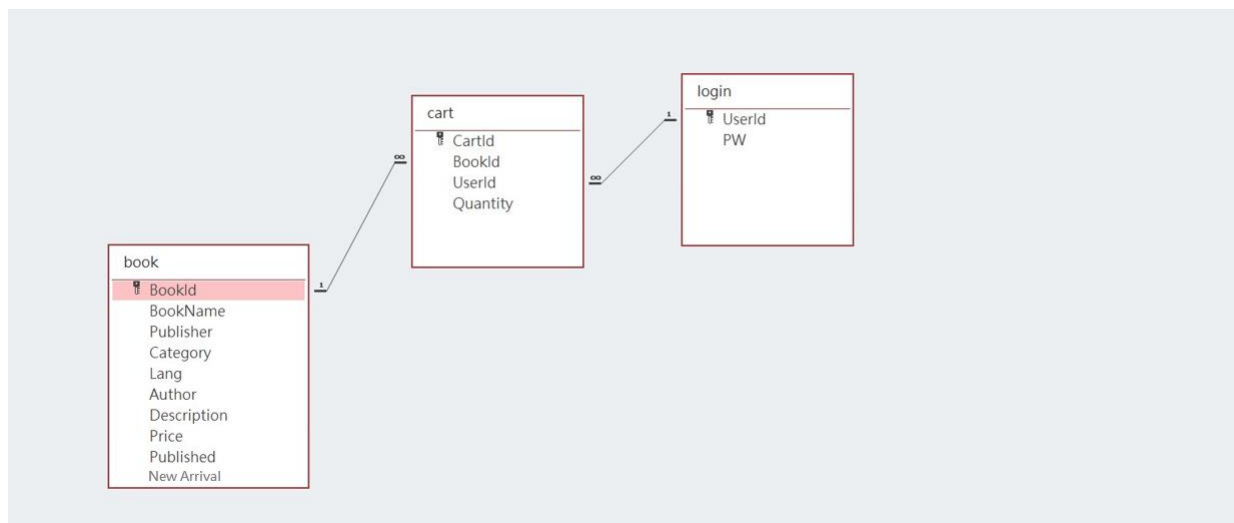
- [Create] form- A create form which uses the method of "POST" to send the data that is entered by the user in the New username textbox and the New password textbox. It is expected that the data will be passed to the server (e.g. a php file called "create.php") which will be implemented later. Besides, upon submitting the form, the following conditions are needed to be checked:
  - o Whether the New username textbox and/or New password textbox is left empty, if yes, display an alert message "Please do not leave the fields empty". The form data entered by the user should not be submitted to the server.

- [New username] textbox- For the user to enter his/her new account name with the placeholder "Desired Username".

- [New password] textbox- For the user to enter his/her new account password with the placeholder "Desired Password". The password should not be visible to others when the user is entering the password in the textbox.

- [Back] button- Upon this button is clicked, it will direct the browser back to the Login page. The button should have a display name "BACK".

- [Sign up] button- A submit button on click would trigger the form to send the data (New username and New password) to the server (e.g. a php file "create.php"). The button should have a display name "CONFIRM".

Figure 2 below illustrates the sample of the Create account page. Note that in Figure 2, we add additional features that you are going to implement **in a later stage**, just implement the basic skeleton as stated above is fine at this stage.

Fig. 2 Sample of Create account page

Before creating the server part to handle the login and the account creation matters, you should create a database schema in the server first (in the example here, we use MySQL Server). The suggested database schema is as follows (you can simply adopt the schema, or you can create your own database schema, as long as you can fulfill the specifications mentioned in this part).

Fig. 3 Suggested database schema

In the book table, there are the following fields:

- BookId (Primary key)
- BookName
- Publisher
- Category
- Lang
- Author
- Description
- Price
- Published
- New Arrival

Note: Lang means the language of the book; Published means the publication place and date of the book which is separated by comma (e.g. United Kingdom, 12 December 2019).

In the cart table, there are the following fields:
- CartId (Primary key)
- BookId (Foreign key, reference to the book table)
- UserId (Foreign key, reference to the login table)
- Quantity

In the login table, there are the following fields:

- UserId (Primary key)
- PW

  Note: PW means the login password.

You are assumed to have the information of at least 12 books stored in the MySQL database, as well as their book images, which should be hosted on the server (The book images can be downloaded from the course's Moodle site). Please visit Appendix A for the information about the 12 books provided by us.

In the example shown in this specification, we would use php for processing the data on the server side. The first 2 php files are "verifyLogin.php" and "create.php". The former is used to verify whether the username and password entered by the user in the Login page match with the username and password stored in the MySQL database while the latter is used to check whether the username entered by the user in the create account page already exists in the MySQL database.

In the "verifyLogin.php", it is expected the following items and functions are included: **(0.5 points)**

★ Receive the username and password passed from the login form in the login page and check whether the user's input matches with the username and password stored in the login table; if yes, store the username in a session (i.e. create a session) and redirect the browser to the main page (which will be implemented later) immediately. If no, display the message "Invalid login, please login again." (with <h1> tag) and redirect back to the login page after 3 seconds. (Note: you may adopt a different design to present/show the login result so long as the main page is shown when logged on successful and remain in the login page when logged on unsuccessful.)

In the "create.php", it is expected the following items and functions are included: **(0.5 points)**

★ Receive the username and password passed from the create form in the create account page and check whether the username inputted by the user already exists in the login table; if yes, display the message "Account already existed" (with <h1>tag) and redirect back to the create account page after 3 seconds. If no, insert the username and password into the login table as a new record, print the message "Account created! Welcome" (with <h1>tag) and redirect the browser back to the login page after 3 seconds. (Again, you may adopt your own design to provide the same service.)

Besides, you should implement a main page of the online bookstore. Note that this main page allows any user to access no matter he/she has logged on or not. This page mainly serves as a place to display books that are on sales (on the right) and link to the category page which implements the refine by category function (on the left). This page also contains the search bar function (at the top of the page) and has a Sort by Price (Highest) function for reordering the book list. The marks **(total: 4 points)** are divided into 3 sections and the requirements are as follows.

Section I (1 point):

On this page, it is expected the following items and functions are included:

★ On the left column of the page, there should be a heading "Category" and all the book categories stored in the database should be displayed in this area and they act as links to jump to the category page (the category page will be implemented later).

★ On the right column of the page, by default, there would be a heading "All Books" and follow it shows all the books (with their title, author, publisher, price information and new arrival (if the book is a new arrival, the word "NEW ARRIVAL !" should also be shown) stored in the database. For each book, there should be a link in the book title which would direct the browser to a page that shows the details of the book (which will be implemented later). Also, there should be a navigation link that shows the current directory location "Home" of the whole website (Fig. 4).

Section II (2 points):

★ On the top of the page, it is expected that the search bar function should be implemented:
  ✓ There should be a textbox with a placeholder "Keyword(s)" which allows users to input keywords (the keywords can **either** be the **book name** or the **author** of the book) for searching a book. It is expected that the user can input single or multiple keyword(s), with space separates each keyword for inputting multiple keywords.
  ✓ There should be a button called "Search", which upon is clicked, would search the books which contain the keyword(s) (**case-sensitive**) and the matched books would be displayed on the right-column of the page (the original content of all books would be replaced by the matched book(s), as well as the heading would be changed from "All Books" to "Searching Results") **without reloading the whole page**.
  ✓ An example: Assume the database only contains 3 books, with the name "Have You Filled A Bucket Today? : A Guide to Daily Happiness for Kids", "The Rule of the Land"

and "Olive, Again". If the user inputs a single keyword "Olive", then the book "Olive, Again" as well as its related information should be displayed. If two keywords "Olive" and "Have" are inputted (i.e. Olive Have), then the book "Have You Filled A Bucket Today? : A Guide to Daily Happiness for Kids" and the book "Olive, Again" and their related information should be displayed in the searching result (Fig. 5.1); if the user inputs a single keyword "Elizabeth", then the book "Olive, Again" as well as its related information should be displayed. If two keywords "Elizabeth" and "Garrett" are inputted (i.e. Elizabeth Garrett), then the book "The Rule of the Land" and the book "Olive, Again" and their related information should be displayed in the searching result (Fig. 5.2).
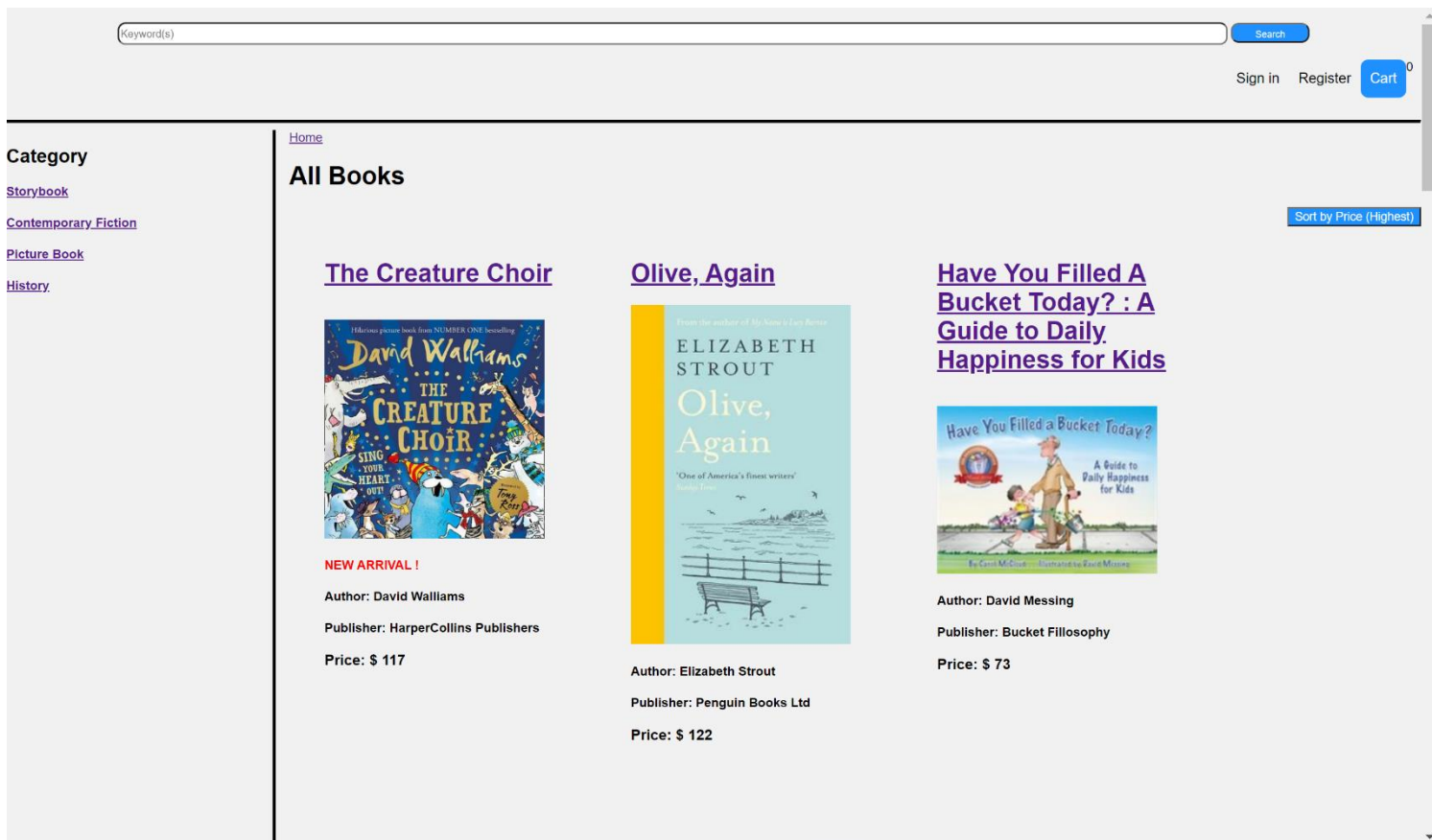
✓ Besides, it is also expected that there would be 2 or 3 buttons on the right side below the search button. If the user has not logged in, there should be 3 buttons - Sign in (link to the login page), Register (link to the create account page), and Cart (link to the cart page which would be implemented later). If the user has logged in, there should be 2 buttons – Logout and Cart (link to the logout php file and the cart page which would be implemented later).

Note: In addition to the main page, the search bar function and the above buttons should also be applied to the login page, the create account page, the cart page, and the category page (the cart and the category page will be implemented later).

Section III (1 point)

★ This page should also implement the Sort by Price (Highest) function. The button – "Sort by Price (Highest)" should be displayed on the right side of the book list. Upon clicking that button, the heading would be changed to "All Books (Sort by Price Highest)" and all the books would be sorted by price, with the highest price displayed first (Fig. 6), **without reloading the whole page**.

Note: The Sort by Price (Highest) function should also be implemented on the category page.

Fig. 4 Sample of the main page by default showing All Books (Note: Only part of the book list is shown for illustration)
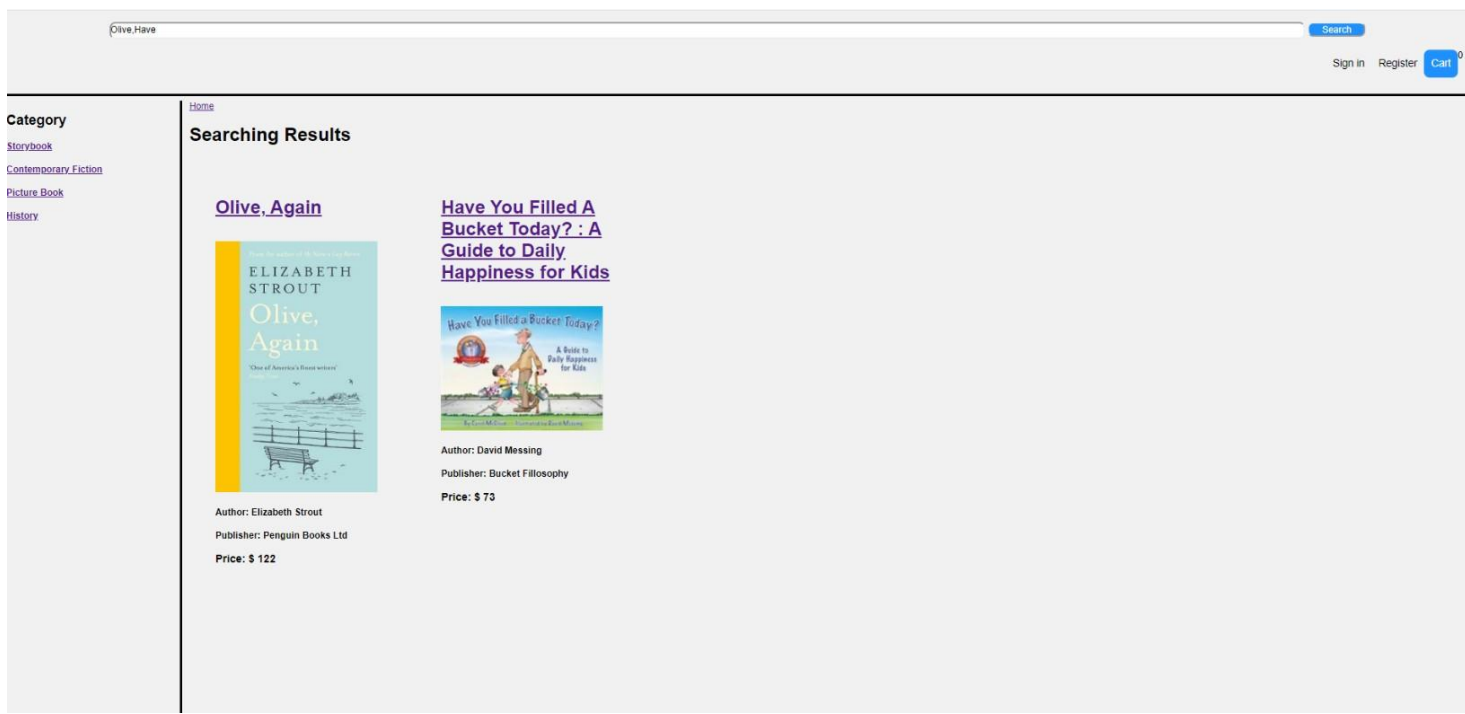
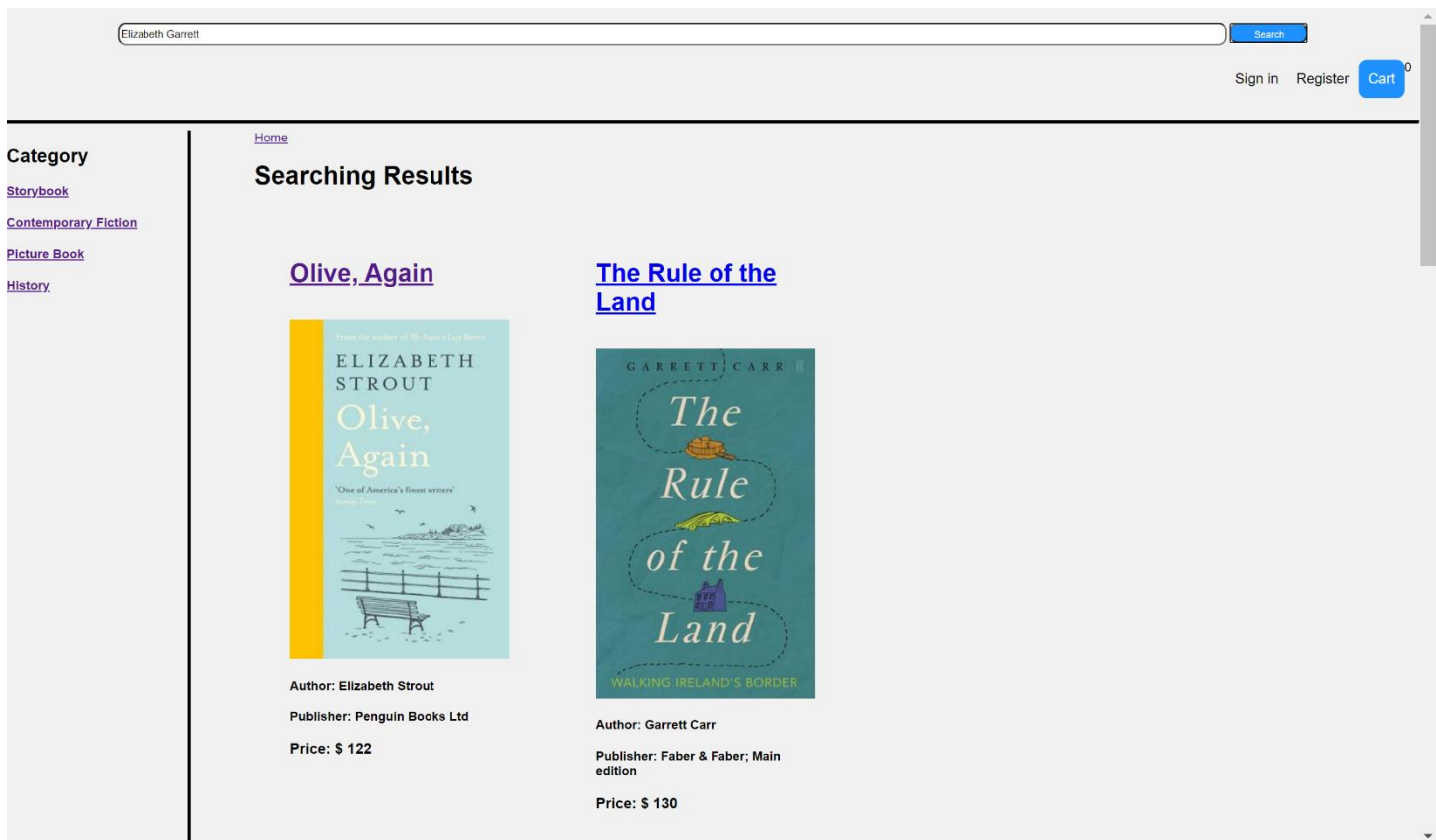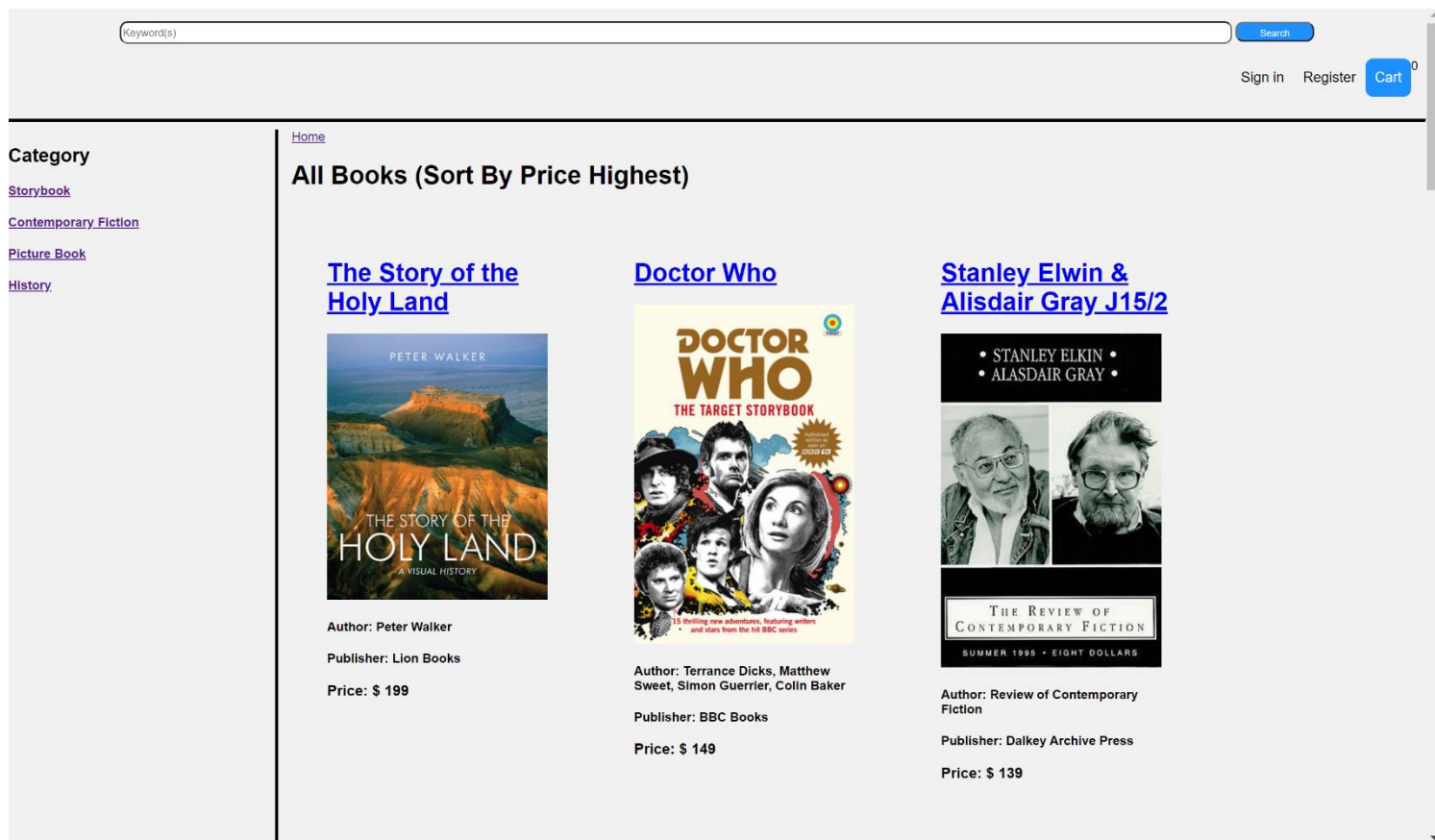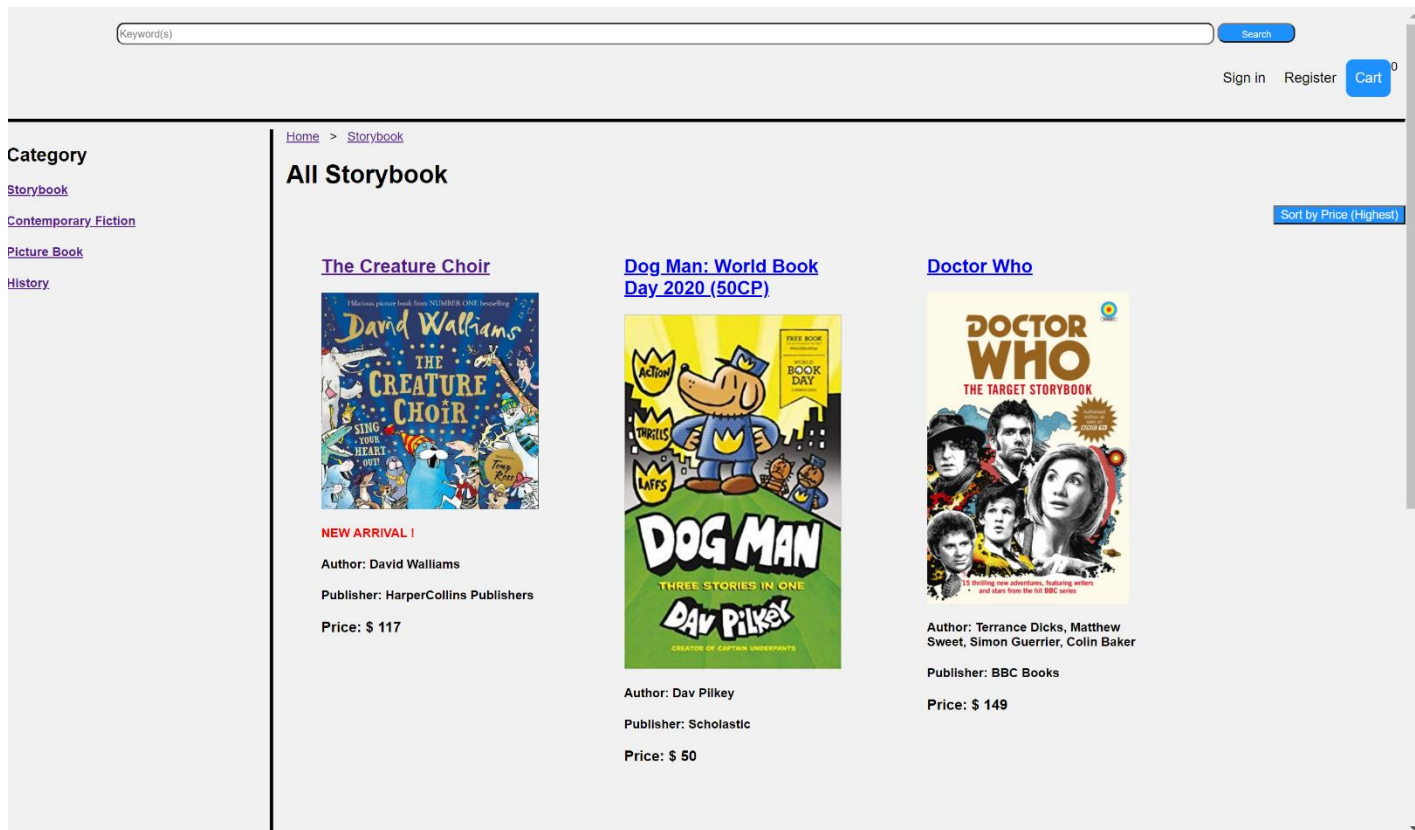**Fig. 5.1 Sample of the main page with searching results by book names**

**Fig. 5.2 Sample of the main page with searching results by book
authors**

Fig. 6 Sample of the main page with Sort by Price Highest on All Books

In the category page, it is expected that all books which belong to that category which the user selected will be displayed on the right column of the page, while on the left column as well as the top of the page, the content displayed should be the same as the one shown in the main page. Besides the <mark>navigation</mark> link should be changed to "Home" > "(the Category selected)" **(0.5 points)** (Fig. 7).
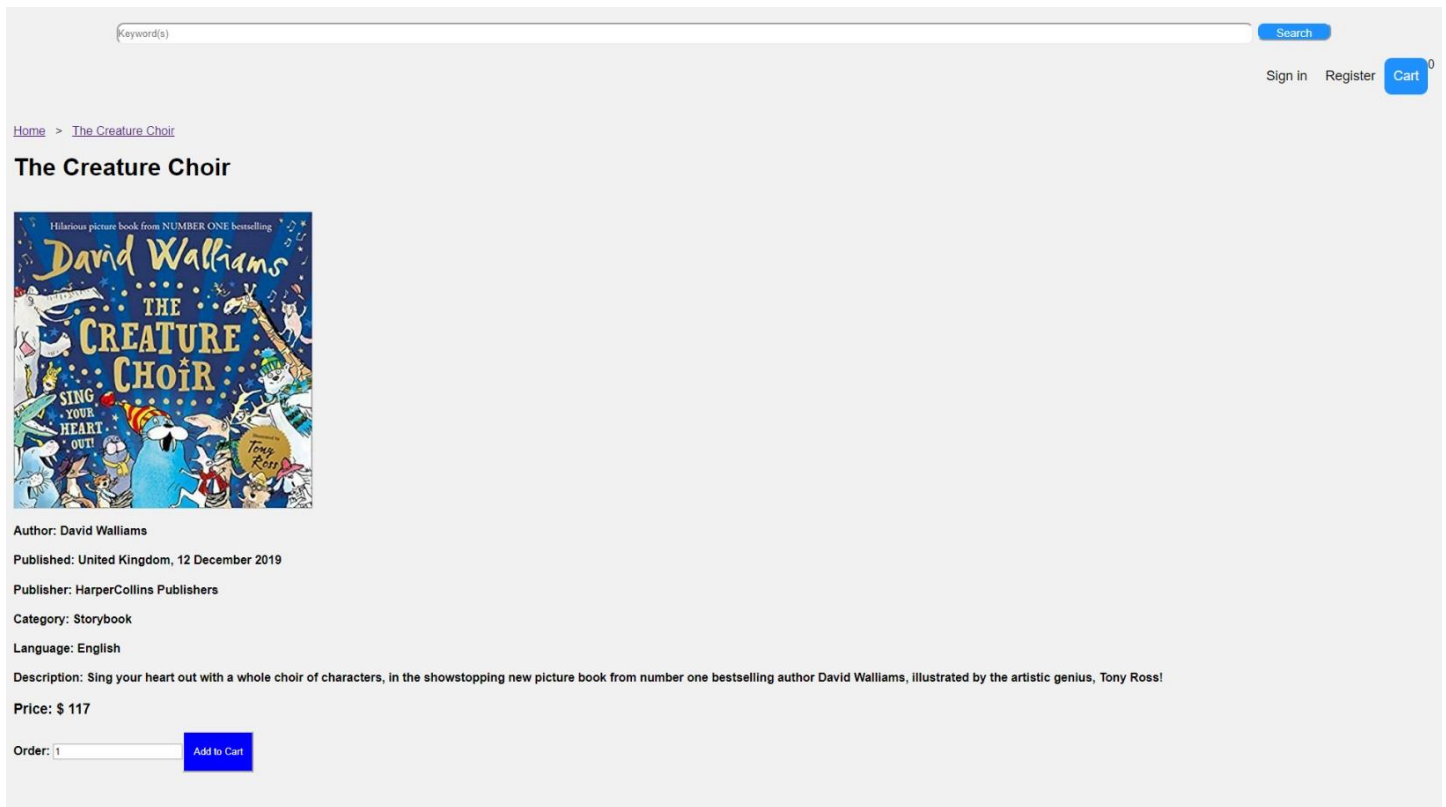
Fig. 7 Sample of the category page when the user selects the category- "Storybook"

As mentioned, for each book, there should be a link in the book title which upon is clicked, would direct to a page which shows the details of the book. It is expected that in addition to the book title, author, publisher, and price; the publication place and date of the book, the category of the book, the Language, and the Description of that book should also be shown. **Please follow the order of the display as shown in Fig. 8 for easy marking**. Besides, below the price, there should be an input textbox for the user to order the book. It allows the user to enter the amount of the book to purchase and it will be added to the shopping cart once submitted. The default value of the textbox would be 1. There should also be a button "Add to Cart" which upon is clicked, would send the buying information to the cart page (implemented later).

Please note that the <mark>navigation</mark> link should be changed to "Home" > "(the book selected)" and the top search bar function and the corresponding buttons should also be shown at the top of the page **(0.5 points).**

Fig. 8 Sample of the book information page

On the cart page, it is expected the following items and functions are included **(4 points):**

- The cart page should display all the information (the book title and the corresponding quantity) of all books that the user has added to the cart (i.e. through clicking the button "Add to cart" on the book page). Note that **no matter the user has logged in or not**, he/she can still add his/her desired books into the cart and they should be able to display them on the cart page.

- Note that the total number of books added to the cart would be displayed next to the cart button on every page that has the cart button.

- Note that the total price of the books added into the cart should be calculated and displayed after displaying all the books in the cart.

- Note that when displaying the cart information, the quantity of each book should be summed up.

- On the cart page, the user can delete his/her cart record by pressing the "Delete" button, and the corresponding record in the cart should be removed. The cart page should be updated, without reloading the whole page.

- Besides, the books that were added into the cart before the user login must be added into the cart associated with the user account after the user has successfully logged in. The original cart information that has been stored in the session (before logged in) should be deleted.

- Note that there would be two buttons, namely "Back" and "Checkout" below the total price, which would direct the browser back to the main page upon the "Back" button is clicked, and would direct to the checkout page (which will be implemented in the next stage) upon the "Checkout" button is clicked.

- You are reminded that at the top, the search bar and the corresponding buttons should be appeared on the page (Fig. 9).

My Shopping Cart

Book Name: The Creature Choir

Quantity: 2

Delete

Book Name: Olive, Again

Quantity: 1

Delete

Book Name: Brave Irene: A Picture Book

Quantity: 3

Delete

Total Price: $ 617

Back Checkout

Fig. 9 Sample of the cart page

Next, we would implement the checkout page, it is expected the following items and functions are included **(4 points):**

★ If the user has not logged in, the checkout page should consist of 3 sections, you can refer to Fig. 10 for references:

✓ The first section would ask whether the user has an existing account, if not, he/she should complete the remaining sections, which consist of "Fast track" registration and Delivery info. In the section of "Fast track registration", the user should input his/her desired username and password for registration. Both the username and password textboxes should not leave empty, otherwise an alert message "Please do not leave the fields empty" would be displayed upon clicking the Confirm button. In addition, the system would check whether the username inputted by the user has already been registered in the database. If yes, after the user entered the username and left the textbox, a red warning message "Username Duplicated!" will be displayed next to the username textbox and the textbox should be clear, without reloading the page. In contrast, if the user has an existing account, he/she should click Sign In, which is a hyperlink that would direct the user to the login page.

✓ It should be noted that in the second section, there are five fields (Full Name, Address Line 1, City, Country, Postcode/Zip Code) which are mandatory for the user to input. There should be a placeholder "Required" in each of these 5 fields in order to remind the user. Similar to the case of encountering the empty username and password textboxes, an alert message (or of any form) " Please do not leave the fields empty" would be displayed in case any one of these 5 fields is left empty when pressing the Confirm button.

✓ In the third section, the user should be able to change the book items inside the cart if he/she wants to, by clicking the "change" hyperlink, which would direct back to the cart page. Besides, it would display the order summary (book titles, quantities, and total price), as well as the confirm button, which upon is clicked, would pass the information inputted on this page and direct the browser to the invoice page.

★ If the user has logged in, the checkout page should only contain 2 sections, you can refer to Fig. 11 for references:

✓ The first section would ask for the Delivery info, which is similar to the second section when the user has not logged in. All the requirements would be the same.

✓ The second section would be the same as the third section when the user has not logged in. All

the requirements would be the same.

Fig. 10 Sample of the checkout page (User has not logged in)

Fig. 11 Sample of the checkout page (User has logged in)

On the invoice page, it would simply display the Delivery info received from the checkout page, as well as the order summary. In addition, there should also be a button "OK" which upon is clicked, would direct back to the main page. You can refer to Fig. 12 for references **(0.5 points)**.

Invoice Page

Full Name: c3322   **Company:** NA
Address Line 1: Oxford, England, United Kingdom
Address Line 2: NA
City: Oxford   **Region:** NA   **Country:** United Kingdom
Postcode: OX1 4BH

2 x The Creature Choir   HK$ 117

Total Price: HK$ 234

Thanks for ordering. Your books will be delivered within 7 working days.

OK

Fig. 12 Sample of the invoice page

The last part would implement the logout feature (e.g. create a file "logout.php").
It is expected the following items and functions are included when the logout button is clicked **(0.5 points):**

> ★ The page would destroy the current session, display the message "Logging out" (with <h2> tag) and redirect the user back to the main page after 3 seconds.

Part II (Total: 4 points)

In this part, you are going to implement the responsive web design (i.e. automatically resizing, hiding, shrinking, or realigning components in the website, to make it looks good on all devices (desktops, tablets, and phones)) for every page that you haved implemented in Part I. You need to make every element in the pages to be responsive to different mobile/computer screen resolutions in order to score the points **(2 points)**.

The remaining **2 points** regarding the use of CSS to decorate the pages that you have implemented in Part I. For CSS implementation, you are free to use any CSS formatting to make the pages look nice. Note: **You are not required to have a fancy design; just use suitable styling should be fine**.

Submission:

Please finish this project before May 4, 2021 Tuesday 23:59.

1. If you are using PHP+MySQL to implement the project, here are the instructions to prepare for the submission:

   i.   You should only use the Department's MySQL server (Sophia) as your project database server, please make sure that you have included a valid MySQL password in your PHP code so that we can access your database for testing and grading. **Using localhost (your own MySQL server) is NOT accepted**.

   ii.   **Upload all your files of the project to the i7.cs.hku.hk server** and place them (in your origin file structure) in your preset directory under the course account (c3322b) for this project, i.e., /course/home/c3322b/public_html/Project/[your_CSID] (which is similar to the setting for the

assignments) and **submit the link to Moodle**.

iii. **You need to make sure that you have viewed your product on the i7 server before submitting your link to Moodle**. We will go to your individual webpage to check your work. Please note that this project will mainly be **marked by features** and **DO NOT** expected that we will evaluate your source code during marking.

iv. Create a text file named "Readme.txt" that contains:
   a. List the version number of the JQuery library if appropriate.
   b. Indicate how much work you have completed for the project. You are recommended to make use of the table shown in the Grading Policy section as the checklist. Using it to show us how much you have accomplished.

2. If you are using Node.js+Express+MongoDB or Node.js+Express+MongoDB+ReactJS to implement the project, here are the instructions to prepare for the submission.

i. You must create a project folder "Project" and place all HTML, CSS, JS, and the database 'data' folder in this project folder. Use a compression or archive tool to compress the Project folder and name the file as Project-yourStudentNo.zip (or tgz or other common compression formats). Before creating the project zip file, **make sure all source files and data for the MongoDB are all placed inside the project folder. Besides, remove the node_modules folder(s) to save space and compression time.**

ii. **Create a text file named "MDBdata.txt" and list all MongoDB statements in creating the database**, the collections, and the documents for testing your program. This is important as the database 'data' folder in the submitted file may be missing or empty. Given these MongoDB statements, tutors can recreate your database to test and grade your project.

iii. Create a text file named "Readme.txt" that contains:

   a. List the version numbers of the jQuery, Node.js, Express.js, MongoDB, and ReactJS installed and used on your platform.
   b. **Indicate how to (1) start the MongoDB database, (2) start the Web server (and ReactJS if is appropriate), and (3) access the main page of the Web application.**
   c. Indicate how much work you have completed for the project. You are recommended to make use of the table shown in the Grading Policy section as the checklist. Using it to show us how much you have accomplished.

iv. **You need to make sure that you have viewed your product before uploading all the files** to Moodle. Please note that this project will mainly be **marked by features** and **DO NOT** expected that we will evaluate your source code during marking.

Grading Policy:

Part I (16 points)	• Correctness of the login page (0.5 points)
	• Correctness of the create account page (0.5 points)
	• Correct functionality of verifying user's login into the system (verifyLogin.php) (0.5 points)
	• Correct functionality of users' account creation (create.php) (0.5 points)
	• Correct functionality of the main page (4 points)
	• Correct functionality of the category page (0.5 points)
	• Correct functionality of the book information page (0.5 points)
	• Correct functionality of the cart page (4 points)
	• Correct functionality of the checkout page (4 points)
	• Correct functionality of the invoice page (0.5 points)
	• Correct functionality of logout in "logout.php" (0.5 points)
Part II (4 points)	• Correct implementation of responsive web features in web pages (2 points)
	• CSS implementation in webpages (2 points)

Plagiarism:

Plagiarism is a very serious academic offence. Students should understand what constitutes plagiarism, the consequences of committing an offence of plagiarism, and how to avoid it. **Please note that we may request you to explain to us how your program is functioning as well as we may also make use of software tools to detect software plagiarism.**

Appendix A

Book 1:

Book Name	The Creature Choir
Publisher	HarperCollins Publishers
Category	Storybook
Lang	English
Author	David Walliams
Description	Sing your heart out with a whole choir of characters, in the showstopping new picture book from number one bestselling author David Walliams, illustrated by the artistic genius, Tony Ross!
Price	117
Published	United Kingdom, 12 December 2019
New Arrival	Yes
Book Image	book_1.jpeg

Book 2:

Book Name	Olive, Again
Publisher	Penguin Books Ltd
Category	Contemporary Fiction
Lang	English
Author	Elizabeth Strout
Description	An extraordinary new novel by the Pulitzer Prize-winning, Number One New York Times bestselling author of Olive Kitteridge and My Name is Lucy Barton...
Price	122
Published	United Kingdom, 31 October 2019
New Arrival	No
Book Image	book_2.jpeg

Book 3:

Book Name	Have You Filled A Bucket Today? : A Guide to Daily Happiness for Kids
Publisher	Bucket Fillosophy
Category	Picture Book
Lang	English
Author	David Messing
Description	Children are not the only ones that need to learn how to be truly happy. It's all in the bucket, that invisible bucket that follows you everywhere... teaches young readers valuable lessons about giving, sharing, and caring...
Price	73
Published	USA, 1 October 2015
New Arrival	No
Book Image	book_3.jpeg

Book 4:

Book Name	Dog Man: World Book Day 2020 (50CP)
Publisher	Scholastic
Category	Storybook
Lang	English
Author	Dav Pilkey
Description	A BRILLIANT new DOG MAN book for World Book Day 2020 packedwith three humorous stories. Dav Pilkey's wildly popular DOG MAN series appeals to readers of all ages and explores universally positive themes...
Price	50
Published	United Kingdom, 27 Feb 2020
New Arrival	No
Book Image	book_4.jpg

Book 5:

Book Name	Brave Irene: A Picture Book
Publisher	Square Fish
Category	Picture Book
Lang	English
Author	William Steig
Description	Brave Irene is Irene Bobbin, the dressmaker's daughter. Her mother, Mrs. Bobbin, isn't feeling so well and can't possibly deliver the beautiful ball gown she's made for the duchess to wear that very evening.
Price	87
Published	USA, 1 January 1996
New Arrival	No
Book Image	book_5.jpeg

Book 6:

Book Name	Stanley Elwin & Alisdair Gray J15/2
Publisher	Dalkey Archive Press
Category	Contemporary Fiction
Lang	English
Author	Review of Contemporary Fiction
Description	Arthur M. Saltzman, Stanley Elkin: An Introduction/Peter J. Bailey, 'A Hat Where There Never Was a Hat': Stanley Elkin's Fifteenth Interview/Stanley Elkin, Words and Music/William Gass, Stanley Elkin: An Anecdote/Jerome Charyn...
Price	139
Published	United States, 8 January 1995
New Arrival	No
Book Image	book_6.jpg

Book 7:

Book Name	The Rule of the Land
Publisher	Faber & Faber; Main edition
Category	History
Lang	English
Author	Garrett Carr
Description	In the wake of the EU referendum, the United Kingdom's border with Ireland has gained greater significance: it is set to become the frontier with the European Union. Over the past year, Garrett Carr has travelled this border...
Price	130
Published	United Kingdom, 1 May 2017
New Arrival	No
Book Image	book_7.jpeg

Book 8:

Book Name	The Story of the Holy Land
Publisher	Lion Books
Category	History
Lang	English
Author	Peter Walker
Description	This is a highly illustrated, visually led guide through the story of the Holy Land, from Bible times to the present day. The Holy Land frequently features in today's headlines as a much fought-for territory.
Price	199
Published	United Kingdom, 22 June 2018
New Arrival	No
Book Image	book_8.jpeg

Book 9:

Book Name	Doctor Who
Publisher	BBC Books
Category	Storybook
Lang	English
Author	Terrance Dicks, Matthew Sweet, Simon Guerrier, Colin Baker
Description	In this exciting collection you'll find all-new stories spinning off from some of your favourite Doctor Who moments across the history of the series. Learn what happened next, what went on before...
Price	149
Published	United Kingdom, 24 October 2019
New Arrival	No
Book Image	book_9.jpeg

Book 10:

Book Name	Festivals of Love
Publisher	Createspace Independent Publishing Platform
Category	Contemporary Fiction
Lang	English
Author	Midwest Fiction Writers
Description	Minnesota: A state blessed with 10,000 lakes, four beautiful seasons, and countless romantics looking for love in all the right places! What better place to find a perfect happily-ever-after than one of Minnesota's uniquely fun...
Price	131
Published	United States, 1 May 2016
New Arrival	No
Book Image	book_10.jpeg

Book 11:

Book Name	The Jungle Book
Publisher	Palazzo Editions Ltd
Category	Picture Book
Lang	English
Author	Rudyard Kipling
Description	Originally published as part of Kipling's famous The Jungle Book, the story of Rikki Tikki Tavi, the little mongoose rescued by a family when he was half-drowned in a storm, has been an enduring favourite with young and old for more than half a century.
Price	80
Published	United Kingdom, 1 September 2018
New Arrival	No
Book Image	book_11.jpeg

Book 12:

Book Name	Sapiens
Publisher	Vintage Publishing
Category	History
Lang	English
Author	Yuval Noah Harari
Description	THE MILLION COPY BESTSELLER Fire gave us power. Farming made us hungry for more. Money gave us purpose. Science made us deadly. This is the thrilling account of our extraordinary history - from insignificant apes to rulers of the world.
Price	103
Published	UK, 29 April 2015
New Arrival	No
Book Image	book_12.jpg