```
File Edit View Navigate Code Refactor Run Tools VCS Window Help main_menu.py - books.py
  Class 12 Computer Project > Project 
             books.py ×
<del>: |</del>
                         import mysgl.connector as conn1
         2
                         import string
          3
                                  This function will help the administrator to add as many books they want in the 'books' table '''
          4
          5
                         def Add_book():
          6
                                  mycon1 = conn1.connect(host="localhost", user="root", password="Rinshu@03", database="book_shop")
          7
          8
                                  cursor = mycon1.cursor(buffered=True)
                                  no_books = int(input("Enter number of books you want to add: "))
         9
                                  for i in range(no_books):
       10
                                            try:
       11
                                                                                                                    Enter the book name: ")).lower()
       12
                                                    book_name = str(input("
       13
                                                    writer_fname = str(input("
                                                                                                                           Enter the first name of author: ")).lower()
                                                    writer_lname = str(input("
                                                                                                                           Enter the last name of author: ")).lower()
       14
                                                                                                         Enter the year of Release: "))
                                                    year = int(input("
       15
                                                                                                       Enter the category of book: ")).lower()
                                                    cat = str(input("
       16
                                                                                                       Enter the Publisher's Name: ")).lower()
       17
                                                    pub = str(input("
       18
                                                    stock = int(input("
                                                                                                           Enter the number of books in stock: "))
                                                    page = int(input("
                                                                                                         Enter number of pages of book: "))
       19
                                                    words = int(input("
                                                                                                           Enter the number of words per page (average) approximate: "))
       20
                                                    price = int(input("
                                                                                                           Enter the price of the book: "))
       22
                                                    cursor.execute(
       23
                                                             f"INSERT INTO books (title, author_fname, author_lname, released_year, stock_quantity, category, "
                                                             f"publication, pages, pg_word, price) VALUES('{book_name}', '{writer_fname}', '{writer_lname}', {year}, "
       24
       25
                                                             f"{stock}, '{cat}', '{pub}', {page}, {words}, {price})")
                                                    mycon1.commit()
       26
ċίι
      27
                                                    print("Successfully Added the book.")
       28
                                           except:
                                                    print("Wrong Data Given.")
       29
Explorer
       30
                                                    Add_book()
                                  mycon1.close()
       31
       32
                                  return
```

```
Project
       books.py X
∷1 34
    35
    36
    37
    38
    39
    40
    41
    42
    43
    44
```

```
''' This function will help the administrator to update books' stock quantity,
               price, number of pages, release year '''
           def Update_book():
               mycon1 = conn1.connect(host="localhost", user="root", password="Rinshu@03", database="book_shop")
               cursor = mycon1.cursor(buffered=True)
               no_books = int(input("Enter number of books you want to update: "))
               count = 0
               while count < no_books:</pre>
                   book_name = str(input("Enter the name of the book you want to update: ")).lower()
                   cursor.execute("SELECT * FROM books WHERE title = '{}'".format(book_name))
   45
                   data = cursor.fetchone()
   46
                   if data == None:
   47
                       print("No book like '{}' exists in database.".format(book_name)) # Checking if the book exists in database
   48
                       Update_book()
                                                                                           # If not found, function runs again.
   49
                   else:
   50
                                   1. Update the book's year: \n",
                       print("
   51
                                   2. Update the book's stock: \n",
   52
                                   3. Update the book's number of pages: \n"
   53
                                   4. Update the book's price: ")
   54
                       choice = int(input("Enter your choice: "))
   55
   56
                       if choice == 1:
                           year = int(input("Enter the new year of the book: "))
   57
Favorites
                           cursor.execute("UPDATE books SET released_year = {} where title = '{}'".format(year, book_name))
   58
                           mycon1.commit() # Updating the book's release year
   59
iί
                           print("Successfully Updated")
  60
                       elif choice == 2:
   61
                           stock = int(input("Enter the stock of the book: "))
   62
AWS Explorer
                           cursor.execute("UPDATE books SET stock_quantity = {} where title = '{}'".format(stock, book_name))
   63
                           mycon1.commit() # Updating the book's stock quantity
   64
                           print("Successfully Updated")
```

```
Project
      books.pv >
                       elif choice == 3:
   66
                           page = int(input("Enter the number of pages of the book: "))
   67
                           cursor.execute("UPDATE books SET pages = {} where title = '{}'".format(page, book_name))
   68
                           mycon1.commit() # Updating the book's number of pages as per new release
   69
                           print("Successfully Updated")
   70
                       elif choice == 4:
   71
                           price_n = int(input("Enter the new price: "))
   72
                           cursor.execute("UPDATE books SET price = {} where title = '{}'".format(price_n, book_name))
   73
                           mycon1.commit() # Updating the book's price
   74
                           print("Successfully Updated")
   75
   76
                       else:
                           print("Wrong Input")
   77
                           Update_book()
   78
                       count += 1
   79
               mycon1.close()
   80
   81
               return
   82
   83
              This function will help the administrator to delete books from the books table'''
   84
   85
           def Delete_book():
   86
               mycon1 = conn1.connect(host="localhost", user="root", password="Rinshu@03", database="book_shop")
   87
               cursor = mycon1.cursor(buffered=True)
   88
               no_books = int(input("Enter number of books you want to delete: "))
   89
               count = 0
   90
               while count < no books:
   91
ĊΪ
                   book_name = str(input("Enter the name of the book you want to delete: ")).lower()
   92
                   cursor.execute("SELECT * FROM books WHERE title = '{}'".format(book_name))
   93
                   data = cursor.fetchone()
   94
'S Explorer
                   if data == None:
   95
                       print("No book like '{}' exists in database.".format(book_name)) # Checking if the book exists in database
```

```
1: Project
      books.py >
   97
                    else:
   98
                        try:
                            cursor.execute("DELETE FROM books where title = '{}'".format(book_name))
   99
                            mycon1.commit() # Deleting the book
  100
                            print("Successfully Deleted.")
  101
                        except:
  102
                             print("This book can't be deleted because Orders for this book exists.")
  103
                    count += 1
  104
                mycon1.close()
  105
                return
  106
  107
  108
           ''' This function will help administrator to view every detail of as many books admin wants.'''
  109
  110
           def View_details():
  111
                mycon1 = conn1.connect(host="localhost", user="root", password="Rinshu@03", database="book_shop")
  112
                cursor = mycon1.cursor(buffered=True)
  113
               no_books = int(input("Enter number of books of which you want details of?: "))
  114
                count = 0
  115
               while count < no_books:</pre>
Structure
  116
                    book_name = str(input("Enter the name of the book " + str(count + 1) + ': ')).lower()
  117
                    cursor.execute("SELECT * FROM books WHERE title = '{}'".format(book_name))
K)
  118
                    data = cursor.fetchone()
  119
                    if data == None:
  120
Favorites
                        choice = str(input(
  121
                            f"No book like '{book_name}' exists in database. Renter Again? (Y or N): ")) # Checking the Wrong Input
  122
  123
                        if choice == 'Y' or 'y':
                            View_details()
  124
                        else:
AWS Explorer
  125
  126
                            exit(0)
                    else:
  127
                                                              ", data[0])
  128
                        print("
                                    Book ID:
```

```
Project
      books.py >
  127
                   else:
                                                             ", data[0])
                       print("
                                   Book ID:
  128
                                                             ", string.capwords(data[1]))
                       print("
  129
                                   Book Name:
                                                             ", string.capwords(data[2]) + ' ' + string.capwords(data[3]))
  130
                        print("
                                   Author's Name:
                                                             ", data[4])
                       print("
                                   Release Year:
  131
                                                             ", string.capwords(data[5]))
                       print("
                                   Category:
  132
                       print("
                                                             ", string.capwords(data[6]))
                                   Publication:
  133
                                                             ", data[8])
                       print("
                                   Pages:
  134
                       print("
                                   Words Per Page (approx): ", data[9])
  135
                                                             ", data[7])
                       print("
                                   Stock Quantity:
  136
                                                             ", data[10])
                       print("
                                   Price:
  137
                   count += 1
  138
               mycon1.close()
  139
               return
  140
  141
  142
           '''This function will help administrator to view details of every book in the shop'''
  143
  144
           def View_all():
  145
               mycon1 = conn1.connect(host="localhost", user="root", password="Rinshu@03", database="book_shop")
  146
  147
               cursor = mycon1.cursor(buffered=True)
               cursor.execute("SELECT * FROM books")
  148
               data = cursor.fetchall()
  149
  150
               for row in data:
  151
                              # fetchall return tuple of tuples. Therefore, taking one tuple at a time in 'tp' through for loop.
                   tp = row
                   print("
                                                         ", tp[0])
                               Book ID:
  152
  153
                                                         ", string.capwords(tp[1]))
                               Book Name:
                   print("
                                                        ", string.capwords(tp[2]) + ' ' + string.capwords(tp[3]))
                   print("
                               Author's Name:
  154
                                                         ", tp[4])
  155
                   print("
                               Release Year:
AWS Explorer
                   print("
                               Category:
                                                         ", string.capwords(tp[5]))
  156
                                                        ", string.capwords(tp[6]))
                   print("
  157
                               Publication:
                                                         ", tp[8])
                   print("
                               Pages:
  158
```

