Exercises

Exercise

Create a class named Circle and intialize it with radius. Make two methods getArea (to return area of a circle) and getCircumference (to return the circumference of the circle)inside this class.

Exercise

You are tasked with developing a Python program for a small e-commerce platform. The program should include various functions to perform essential operations related to product management. Design and implement the following functions:

1.add_product: This function should take the following parameters:

- **product_name** (string): representing the name of the product.
- **price** (float): representing the price of the product.
- **quantity** (int): representing the initial quantity of the product in stock.

The function should create a dictionary object for the product with the given parameters and return it.

2.update_price: This function should take the following parameters:

- **product** (dictionary): representing the product dictionary object.
- **new_price** (float): representing the updated price of the product.

The function should update the price of the product in the dictionary and return the modified dictionary.

3.update_quantity: This function should take the following parameters:

- **product** (dictionary): representing the product dictionary object.
- **quantity_change** (int): representing the change in quantity of the product (positive for addition, negative for subtraction).

The function should update the quantity of the product in the dictionary and return the modified dictionary.

Exercise

Create a python class called BankAccount with attributes like account_number, balance, date_of_opening and customer_name, and methods like deposit, withdraw, check_balance and customer_details

i. The deposit method should return the amount deposited

ii. The withdraw method return insufficient balance if account balance is less than amount to be withdrawn else it should return the amount that has been withdrawn.

iii. The check_balance method should print the current balance.

iv. The customer_details method should print customer name, account number, date of account opening and balance

Create an instance of the BankAccount class to represent a bank account. Perform the following operations: make a deposit, withdraw funds, and display the account information for the created bank account instance.

Exercise

Consider a scenario where you are developing a software application for a library management system using Python. You have identified the need for creating classes to represent books and library members. Design and implement the necessary classes using object-oriented programming concepts.

1. Create a class named "Book" with the following attributes and methods:

Attributes:

1. title: a string representing the title of the book.
2. author: a string representing the author of the book.
3. publication_year: an integer representing the year of publication.
4. borrowed: a boolean indicating whether the book is currently borrowed or not.

Methods:

1. borrow_book(): a method that sets the "borrowed" attribute to True, indicating that the book has been borrowed.
2. return_book(): a method that sets the "borrowed" attribute to False, indicating that the book has been returned.
3. display_info(): a method that displays the title, author, publication year, and borrowed status of the book.

Exercise

1.Create a class named "LibraryMember" with the following attributes and methods:

Attributes:

1. member_id: an integer representing the unique identifier of the library member.
2. name: a string representing the name of the library member.
3. borrowed_books: a list of Book objects representing the books currently borrowed by the member.

Methods:

1. borrow_book(book): a method that takes a Book object as input and adds it to the "borrowed_books" list.
2. return_book(book): a method that takes a Book object as input and removes it from the "borrowed_books" list.
3. display_info(): a method that displays the member ID, name, and the list of borrowed books.

1.Implement appropriate inheritance between the classes to represent the relationship between books and library members.