9. ILLUSTRATION OF BINARY FILE PROGRAMING-III

A binary file named inventory.dat contain certain records of stock (product id, product name, quantity and price). Write a menu driven python program to do the following task:

1. Append a product record
2. Update a product based on the product id
3. Read and display all products

Source Code

```
import pickle

f = open("inventory.dat", "a") # Ensure that the file exists
f.close()

products = []
with open("inventory.dat", "rb") as f:
    while True:
        try: # Using a try block to catch errors
            product = pickle.load(f)
            products.append(product)
        except EOFError:
            break

def appendProduct():
    productID = int(input("Enter product ID: "))
    productName = input("Enter product name: ")
    productQnt = int(input("Enter quantity: "))
    productPrice = int(input("Enter product price: "))


    with open("inventory.dat", "ab") as f:
        pickle.dump([productID, productName, productQnt, productPrice], f)

def updateProduct():
    productID = int(input("Enter product ID: "))
    productName = input("Enter new product name: ")
    productQnt = int(input("Enter new quantity: "))
    productPrice = int(input("Enter new product price: "))
```

```
    products = []
    with open("inventory.dat", "rb") as f:
        while True:
            try: # Using a try block to catch errors
                product = pickle.load(f)
                products.append(product)
            except EOFError:
                break

    newProducts = []
    for product in products:
        if product[0] == productID:
            newProducts.append([productID, productName, productQnt, productPr
        else:
            newProducts.append(product)

    with open("inventory.dat", "ab") as f:
        for product in newProducts:
            pickle.dump(product, f)

    print("Updated product with ID", productID)
    print("--------------------")
    print("ID      :", productID)
    print("Name    :", productName)
    print("Quantity:", productQnt)
    print("Price   :", productPrice)
    print("--------------------")



def showProducts():
    products = []
    with open("inventory.dat", "rb") as f:
        while True:
            try: # Using a try block to catch errors
                product = pickle.load(f)
                products.append(product)
            except EOFError:
                break


    for product in products:
        print("--------------------")
        print("ID      :", product[0])
        print("Name    :", product[1])
        print("Quantity:", product[2])
        print("Price   :", product[3])
        print("--------------------")



while True:
```

```
    print("============================================")
    print("What would you like to do?")
    print("""
    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit
    """)

    ch = input("Enter your choice[1/2/3/4]: ")

    if ch == "1":
        appendProduct()

    elif ch == "2":
        updateProduct()

    elif ch == "3":
        showProducts()

    elif ch == "4":
        print("[ Exiting ]") # Break from the loop to exit
        break

    else:
        print("[ Invalid Choice ]") # In case user inputs a choice that was n
```

OUTPUT

```
============================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter product ID: 1
Enter product name: A
Enter quantity: 234
Enter product price: 234
============================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit
```

```
Enter your choice[1/2/3/4]: 1
Enter product ID: 2
Enter product name: B
Enter quantity: 234
Enter product price: 34
==========================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter product ID: 3
Enter product name: C
Enter quantity: 234
Enter product price: 45
==========================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit

Enter your choice[1/2/3/4]: 3
--------------------
ID      : 1
Name    : A
Quantity: 234
Price   : 234
--------------------
--------------------
ID      : 2
Name    : B
Quantity: 234
Price   : 34
--------------------
--------------------
ID      : 3
Name    : C
Quantity: 234
Price   : 45
--------------------
==========================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
```

```
    [4] Exit

Enter your choice[1/2/3/4]: 2
Enter product ID: 2
Enter new product name: Z
Enter new quantity: 11
Enter new product price: 111
Updated product with ID 2
--------------------
ID      : 2
Name    : Z
Quantity: 11
Price   : 111
--------------------
============================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit

Enter your choice[1/2/3/4]: 3
--------------------
ID      : 1
Name    : A
Quantity: 234
Price   : 234
--------------------
--------------------
ID      : 2
Name    : B
Quantity: 234
Price   : 34
--------------------
--------------------
ID      : 3
Name    : C
Quantity: 234
Price   : 45
--------------------
--------------------
ID      : 1
Name    : A
Quantity: 234
Price   : 234
--------------------
--------------------
ID      : 2
Name    : Z
Quantity: 11
Price   : 111
```

```
--------------------
--------------------
ID      : 3
Name    : C
Quantity: 234
Price   : 45
--------------------
============================================
What would you like to do?

    [1] Append a product
    [2] Update a product
    [3] Show all products
    [4] Exit

Enter your choice[1/2/3/4]: 4
[ Exiting ]
```