|  |  |
| --- | --- |
| EX:NO:4a | Implementing real-time/technical applications using Lists, Tuples. Items present in a library |
| Date: |

**Aim:**

To Implement real-time/technical applications using Lists, Tuples. Items present in a library.

**Algorithm:**

Step1:

Step2:

Step3:

Step4:

Step5:

Step6:

**Program:**

class Library:

def \_\_init\_\_(self):

self.items = []

def add\_item(self, title, author):

unique\_id = len(self.items) + 1

item = {

'id': unique\_id,

'title': title,

'author': author

}

self.items.append(item)

print(f"Item '{title}' by {author} added with ID {unique\_id}")

def remove\_item(self, item\_id):

for item in self.items:

if item['id'] == item\_id:

self.items.remove(item)

print(f"Item with ID {item\_id} removed.")

return

print(f"Item with ID {item\_id} not found.")

def display\_items(self):

if not self.items:

print("The library is empty.")

else:

print("Library Items:")

for item in self.items:

print(f"ID: {item['id']}, Title: {item['title']}, Author: {item['author']}")

def main():

library = Library()

while True:

print("\nLibrary Management System")

print("1. Add Item")

print("2. Remove Item")

print("3. Display Items")

print("4. Exit")

choice = input("Enter your choice: ")

if choice == '1':

title = input("Enter item title: ")

author = input("Enter author: ")

library.add\_item(title, author)

elif choice == '2':

item\_id = int(input("Enter item ID to remove: "))

library.remove\_item(item\_id)

elif choice == '3':

library.display\_items()

elif choice == '4':

print("Goodbye!")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Result:**