## **QUESTION 1**

Assignment -6  
Write a program that creates an integer array of 10 elements, stores the values into a file, and  
then retrieves them to display on the console.  
Create a list of strings, save it into a file, and then read the file to retrieve the list and display the  
strings on the console.  
Sort an array of objects (e.g., Student ) and store them to a file .  
Create a dictionary of <string, int > pairs representing names and ages. Save the dictionary to a file,  
and retrieve the data to display it in a readable format.  
Prompt the user to enter 5 integers, store them in an array, write the array to a file, and then  
retrieve and display the array from the file.  
Create a List<Product> class with properties name, price, and quantity . Save the list to a file and  
retrieve it to display the details of each product.  
Create an array of objects representing employees. Store the array to a file and retrieve it,  
Create an array of random integers, sort it using a sorting algorithm, and store the sorted array  
into a file. Late r, retrieve and display the sorted array.  
Create a list of book titles and authors, store it in a text file, and later retrieve and display the list  
in a formatted output.  
Create a 2D array of integers, write it to a file in a specific format, and retrieve it back to display  
as a matrix.  
Create a queue of integers, write the queue to a file, and then read and display the values in the  
queue after retrieving them.  
Create a list of tuples where each tuple contains a name, age, and city. Store the list into a file,  
and retrieve and display each tuple.  
Define a structure Student with id, name, and marks . Create an array of structures, save it to a file,  
and later retrieve the structure data.  
Implement a program that uses a stack of strings to store a sequence of user-entered words. Write  
the stack to a file and later read the file to display the stack contents.  
Create a list of DateTime objects, save the dates to a text file in a specific format, and retrieve  
them back from the file to display the dates in a read able format.

### **Code Solution**

import pickle  
import json  
import datetime  
import random  
  
class Student:  
 def \_\_init\_\_(self, id, name, marks):  
 self.id = id  
 self.name = name  
 self.marks = marks  
 def \_\_repr\_\_(self):  
 return f"Student(id={self.id}, name='{self.name}', marks={self.marks})"  
  
class Product:  
 def \_\_init\_\_(self, name, price, quantity):  
 self.name = name  
 self.price = price  
 self.quantity = quantity  
 def \_\_repr\_\_(self):  
 return f"Product(name='{self.name}', price={self.price}, quantity={self.quantity})"  
  
class Employee:  
 def \_\_init\_\_(self, name, emp\_id, department):  
 self.name = name  
 self.emp\_id = emp\_id  
 self.department = department  
 def \_\_repr\_\_(self):  
 return f"Employee(name='{self.name}', emp\_id={self.emp\_id}, department='{self.department}')"  
  
int\_array = [42, 78, 15, 96, 23, 57, 84, 31, 69, 3]  
with open('int\_array.txt', 'w') as f:  
 for num in int\_array:  
 f.write(f"{num}\n")  
with open('int\_array.txt', 'r') as f:  
 retrieved\_int\_array = [int(line.strip()) for line in f]  
print("Integer array:", retrieved\_int\_array)  
  
string\_list = ["apple", "banana", "cherry", "date", "elderberry"]  
with open('string\_list.txt', 'w') as f:  
 for s in string\_list:  
 f.write(f"{s}\n")  
with open('string\_list.txt', 'r') as f:  
 retrieved\_string\_list = [line.strip() for line in f]  
print("String list:", retrieved\_string\_list)  
  
students = [Student(1, "Alice", 85), Student(2, "Bob", 92), Student(3, "Charlie", 78)]  
students\_sorted = sorted(students, key=lambda x: x.marks, reverse=True)  
with open('students.pkl', 'wb') as f:  
 pickle.dump(students\_sorted, f)  
with open('students.pkl', 'rb') as f:  
 retrieved\_students = pickle.load(f)  
print("Sorted students:", retrieved\_students)  
  
age\_dict = {"Alice": 25, "Bob": 30, "Charlie": 35, "Diana": 28}  
with open('ages.json', 'w') as f:  
 json.dump(age\_dict, f)  
with open('ages.json', 'r') as f:  
 retrieved\_age\_dict = json.load(f)  
print("Age dictionary:", retrieved\_age\_dict)  
  
user\_ints = [10, 20, 30, 40, 50]  
with open('user\_ints.txt', 'w') as f:  
 for num in user\_ints:  
 f.write(f"{num}\n")  
with open('user\_ints.txt', 'r') as f:  
 retrieved\_user\_ints = [int(line.strip()) for line in f]  
print("User integers:", retrieved\_user\_ints)  
  
products = [Product("Laptop", 999.99, 5), Product("Mouse", 25.50, 20), Product("Keyboard", 75.00, 15)]  
with open('products.pkl', 'wb') as f:  
 pickle.dump(products, f)  
with open('products.pkl', 'rb') as f:  
 retrieved\_products = pickle.load(f)  
print("Products:", retrieved\_products)  
  
employees = [Employee("John", 101, "IT"), Employee("Jane", 102, "HR"), Employee("Mike", 103, "Finance")]  
with open('employees.pkl', 'wb') as f:  
 pickle.dump(employees, f)  
with open('employees.pkl', 'rb') as f:  
 retrieved\_employees = pickle.load(f)  
print("Employees:", retrieved\_employees)  
  
random\_ints = [random.randint(1, 100) for \_ in range(10)]  
random\_ints\_sorted = sorted(random\_ints)  
with open('sorted\_ints.txt', 'w') as f:  
 for num in random\_ints\_sorted:  
 f.write(f"{num}\n")  
with open('sorted\_ints.txt', 'r') as f:  
 retrieved\_sorted\_ints = [int(line.strip()) for line in f]  
print("Sorted random integers:", retrieved\_sorted\_ints)  
  
books = [("The Great Gatsby", "F. Scott Fitzgerald"), ("1984", "George Orwell"), ("To Kill a Mockingbird", "Har

### **FINAL Output**

