CODE-1

def merge\_sort(arr):

if len(arr) <= 1:

return arr

mid = len(arr) // 2

left = arr[:mid]

right = arr[mid:]

left = merge\_sort(left)

right = merge\_sort(right)

return merge(left, right)

def merge(left, right):

result = []

i = j = 0

while i < len(left) and j < len(right):

if left[i] < right[j]:

result.append(left[i])

i += 1

else:

result.append(right[j])

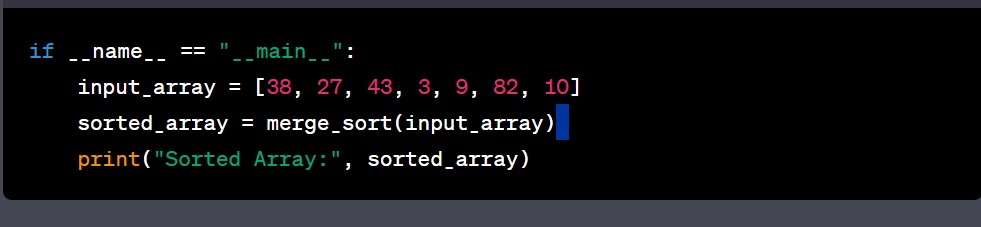
j += 1

result.extend(left[i:])

result.extend(right[j:])

return result

OUTPUT



CODE-2

class BankAccount:

def \_\_init\_\_(self, account\_number, account\_holder, balance=0.0):

self.account\_number = account\_number

self.account\_holder = account\_holder

self.balance = balance

def deposit(self, amount):

if amount > 0:

self.balance += amount

print(f"Deposited ${amount}. New balance: ${self.balance}")

else:

print("Invalid deposit amount. Please enter a positive value.")

def withdraw(self, amount):

if amount > 0:

if amount <= self.balance:

self.balance -= amount

print(f"Withdrew ${amount}. New balance: ${self.balance}")

else:

print("Insufficient funds for withdrawal.")

else:

print("Invalid withdrawal amount. Please enter a positive value.")

def get\_balance(self):

return self.balance

def \_\_str\_\_(self):

return f"Account Number: {self.account\_number}\nAccount Holder: {self.account\_holder}\nBalance: ${self.balance}"

# Create a dictionary to store bank accounts

accounts = {}

def create\_account():

account\_number = input("Enter account number: ")

account\_holder = input("Enter account holder's name: ")

if account\_number not in accounts:

accounts[account\_number] = BankAccount(account\_number, account\_holder)

print(f"Account created for {account\_holder}.")

else:

print("Account with this number already exists.")

def deposit():

account\_number = input("Enter account number: ")

if account\_number in accounts:

amount = float(input("Enter the deposit amount: $"))

accounts[account\_number].deposit(amount)

else:

print("Account not found.")

def withdraw():

account\_number = input("Enter account number: ")

if account\_number in accounts:

amount = float(input("Enter the withdrawal amount: $"))

accounts[account\_number].withdraw(amount)

else:

print("Account not found.")

def check\_balance():

account\_number = input("Enter account number: ")

if account\_number in accounts:

balance = accounts[account\_number].get\_balance()

print(f"Current balance for account {account\_number}: ${balance}")

else:

print("Account not found.")

def main():

print("Welcome to the Bank Account Management System!")

while True:

print("\nMenu:")

print("1. Create Account")

print("2. Deposit Funds")

print("3. Withdraw Funds")

print("4. Check Balance")

print("5. Exit")

choice = input("Enter your choice (1/2/3/4/5): ")

if choice == '1':

create\_account()

elif choice == '2':

deposit()

elif choice == '3':

withdraw()

elif choice == '4':

check\_balance()

elif choice == '5':

print("Thank you for using the Bank Account Management System!")

break

else:

print("Invalid choice. Please select a valid option.")

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

main()

OUTPUT

