Derived Data Types - Experiment 2

Aim : Write a menu-driven text application to maintain bank accounts of customers using Lists, Dictionary, Strings and Sets. This application handles error and exception using try block. The application allows following operations as python functions:

i) Adding and delete accounts of customers,

In [21]:

```
customers_list = ['alice','bob']

def add_customer():
    customer_name= input('Enter Customer Name You Want To Add :').lower()
    customers_list.append(customer_name)
    print('The customer list is: ',customers_list)
add_customer()
```

```
Enter Customer Name You Want To Add :Hamza
The customer list is:  ['alice', 'bob', 'hamza']
```

In [22]:

```
def del_customer():
    customer_name= input('Enter Customer Name You Want To Remove: ').lower()
    try:
        customers_list.remove(customer_name)
        print("After removing the customer name the current list is: ",customers_list)
    except:
          print("Customer Name not present in the list.Please try another name")


del_customer()
```

```
Enter Customer Name You Want To Remove: HAMZA
After removing the customer name the current list is:  ['alice', 'bob']
```

ii) Deposit money to/from accounts,

In [2]:

```
# Deposit
customer_list_balance = {'alice': 3000, 'bob': 4000, 'hamza': 5000}
try:
    customer_name = input('enter customer name you want to check balance: ').lower()
    customer_list_balance[customer_name]

except KeyError as error:
    print("This name customer not present")
    exit()
else:
    selected_acc_bal = str(customer_list_balance.get(customer_name))
    print("The current balance in " + customer_name + "'s account is: " + selected_acc_bal)
    amount_deposit = int(input('Enter the amount you want to deposit in ' + customer_name + "'s acc
ount "))
    customer_list_balance[customer_name] += amount_deposit
    print(customer_list_balance)

finally:
    print("Thankyou for banking with us :)")
```

```
enter customer name you want to check balance: hamza
The current balance in hamza's account is: 5000
Enter the amount you want to deposit in hamza's account 800
{'alice': 3000, 'bob': 4000, 'hamza': 5800}
Thankyou for banking with us :)
```

ii) withdraw money to/from accounts,

In [12]:

```
# Withdrawal
customer_list_balance = {'alice': 3000, 'bob': 4000, 'hamza': 5000}
try:
    customer_name = input('enter customer name you want to check balance: ').lower()
    customer_list_balance[customer_name]

except KeyError as error:
    print("This name customer not present")

else:
    selected_acc_bal = str(customer_list_balance.get(customer_name))
    print("The current balance in " + customer_name + "'s account is: " + selected_acc_bal)
    amount_deposit = int(input('Enter the amount you want to withdraw from ' + customer_name + "'s
account "))
    if customer_list_balance[customer_name] > amount_deposit:
        customer_list_balance[customer_name] -= amount_deposit
        print(customer_list_balance)
    else:
        print("You have insufficient balance")


finally:
    print("Thankyou for banking with us :)")
```

```
enter customer name you want to check balance: hamza
The current balance in hamza's account is: 5000
Enter the amount you want to withdraw from hamza's account 3000
{'alice': 3000, 'bob': 4000, 'hamza': 2000}
Thankyou for banking with us :)
```

iii) list last 3 transactions on a particular accounts

In [4]:

```
data = {'hamza': [1100, 1200,2000,4000], 'alice': [4500, 6000]}
try:
    cust_name=input('enter customer name you want to check last 3 transaction: ').lower()
    my_list = data[cust_name]
    print(my_list[-3:])
except KeyError as error:
    print("This name customer not present")
    exit()
finally:
    print("Thankyou for banking with us :)")
```

```
enter customer name you want to check last 3 transaction: hamza
[1200, 2000, 4000]
Thankyou for banking with us :)
```

iv) exit from the application.

In [3]:

```
import sys
sys.exit
```

Out[3]:

```
<function sys.exit>
```