

PiggyBank Project

In [3]:

```

import sqlite3

###Creating a table for the first time transaction
def dbase_create():
    db=sqlite3.connect("piggybank.db")
    db.execute("create table PiggyBank(Actnumber text,Amt int)")
    db.execute("insert into PiggyBank(Actnumber,Amt) values (101,0)")
    db.commit()
    db.close()
def piggy_bank():
    #Actid = '1111'
    while True:
        try:
            SE = str(input("Start or End:").upper())
            if (SE != "START" and SE != "END"):
                print("Please enter correect value")
            except ValueError:
                print("Please enter correect value")
                SE = str(input("Start or End:").upper())
                continue
        else:
            # connecting to DB and verifying the table is avaialable or not if n
            # balance amount in table with a fixed account number 101
            db=sqlite3.connect("piggybank.db")
            tbl=db.execute("SELECT count(*) FROM sqlite_master WHERE type='tabl
            e' AND name='PiggyBank'")
            for row in tbl:
                tblexist= row
            if tblexist[0]<=0:
                dbase_create()
            if SE=="START":
                print("Welcome to Piggy Bank: \n")
                op=input("Please Select D for Deposit, W for withdraw and C for
                balance verification :").upper()
                if op == 'D':
                    Deposit()
                elif op=='W':
                    Withdrawl()
                elif op=='C':
                    Check()
                else :
                    print("you entered an invalid transaction")
            elif SE=="END":
                print("Thanks for accessing Piggy bank and its closing now")
                break
            continue

def Deposit():
    Actid='101' # Making account id as hardcoed to 101 for standard usage we can
    make this reusable for multiple accounts if needed
    while True:
        try:
            dep1=int(input("Enter an Amount for deposit:"))
        except ValueError:
            print("Please enter numeric value for deposit")
            continue
        else:

```

```

db=sqlite3.connect("piggybank.db")
result=db.execute("select Amt from PiggyBank where Actnumber = '10
1'")

    for row in result:
        Amt=row
        AvailableAmt=Amt[0]
        # newly Deposited amount is adding to the existing balace amount
        AvailableAmt = AvailableAmt+depl
        db.execute("UPDATE PiggyBank SET Amt =? WHERE Actnumber =?",(Availab
leAmt,Actid))
        db.commit()
        db.close()
        print("after depositing the amount your available balance is {}".for
mat(AvailableAmt))
        return

def Withdrawl():
    Actid='101' ## Making account id as hardcoed to 101 for standard usage we ca
n make this reusable for multiple accounts if needed
    while True:
        try:
            wdamt1=int(input("Enter an Amount for Withdrawl:"))
        except ValueError:
            print("Looks like yo uare not entered numeric value withdrawl")
            continue
        else:
            db=sqlite3.connect("piggybank.db")
            result=db.execute("select Amt from PiggyBank where Actnumber = '10
1'")

            for row in result:
                Amt=row
                AvailableAmt=Amt[0]

            if AvailableAmt >=wdamt1 :
                # newly withdrawl amount is deducting from the existing balace a
mount
                AvailableAmt = AvailableAmt- wdamt1
                db.execute("UPDATE PiggyBank SET Amt =? WHERE Actnumber =?",(Ava
ilableAmt,Actid))
                db.commit()
                db.close()
                print("After withdrawl, Your available balance is {}".format(Ava
ilableAmt))
            else:
                print("you do not have sufiiceint amount to withdraw, the availa
ble balance is {}".format(AvailableAmt))
                return

def Check():
    db=sqlite3.connect("piggybank.db")
    result=db.execute("select Amt from PiggyBank where Actnumber = '101',")
    for row in result:
        Amt=row
        AvailableAmt=Amt[0]
        print("Your available balance is {}".format(AvailableAmt))
        db.commit()
        db.close()

piggy_bank()

```

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :D

Enter an Amount for deposit:10000

after depositing the amount your available balance is 10000

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :D

Enter an Amount for deposit:5000

after depositing the amount your available balance is 15000

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :W

Enter an Amount for Withdrawl:15000

After withdrawl, Your available balance is 0

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :W

Enter an Amount for Withdrawl:2000

you do not have sufiiceint amount to withdraw, the available balance is 0

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :D

Enter an Amount for deposit:3000

after depositing the amount your available balance is 3000

Start or End:start

Welcome to Piggy Bank:

Please Select D for Deposit, W for withdraw and C for balance verification :C

Your available balance is 3000

Start or End:End

Thanks for accessing Piggy bank and its closing now

In []: