L53 Final Exam Solution

Q1.

```
drivers=int(input('Number of drivers: '))
totAllBills=0.0
for i in range(1,drivers+1):
    orders=int(input('Number of orders: '))
    totDriverBills=0.0
    minDriverNo=0
    minBill=float('INF')
    for j in range(1,orders+1):
        bill=float(input('Bill amount: '))
        totDriverBills+=bill
        if bill<minBill:
            minBill=bill
            minDriverNo=i
    print(f'Total amount of all orders delivered by driver {i} =
{totDriverBills}')
    totAllBills+=totDriverBills
print(f'Total amount of all orders delivered by all drivers = {totAllBills}')
print(f'Driver with min order amount is driver No. {minDriverNo}')
```

Q2.

```
try:
    events=set()
    f=open('donations.txt','r')
    f.readline()
    a=0.0
    b=0
    c=list()
    for line in f:
        line=line.strip('\n')
        data=line.split(',')
        dateandtime=data[1].split(' ')
        date=dateandtime[0].split('-')
        time=dateandtime[1].split(':')
        events.add(data[0])
        if data[0]=='HDYYW00E013' and data[2]=='Unknown':
            a+=float(data[3].rstrip('$'))
        if data[2]=='Unknown' and date[1]=='10':
            b+=1
        if data[2]=='Unknown' and len(data)==5:
            t=int(time[0]+time[1])
            rating=int(data[4])
            if rating>6 and 1030<=t<=1730:
                c.append(data[0])
    print(a)
    print(b)
    print(c)
    out=open('events.txt','w')
    for event in events:
        out.write(f'{event}\n')
except FileNotFoundError:
    print('FileNotFoundError')
finally:
    f.close()
    out.close()
```

Q3.

```
def validateVoucher(code):
    if len(code)!=12: return 0
    if not (code[0].isalpha() and code[1].isalpha() and code[2].isalpha()):
return False
    if not (code[0].isupper() and code[1].islower() and code[2].isupper()):
return False
    l=u=s=d=0
    for i in range(3,10):
        if code[i].isupper():u+=1
        if code[i].islower():l+=1
        if code[i]=='_':s+=1
        if code[i].isdigit():d+=1
    if not (l==2 and u==1 and s==1 and d==3): return False
    if not (code[10].isdigit() and code[11].isdigit()): return False
    discount=int(code[-2:])
    return discount
amount=float(input('Bill amount: '))
code=input('Voucher code: ')
code='AbC1D2e_3f10'
discount = validateVoucher(code)
toPay = amount - amount*discount/100
print(f'Bill amount = {amount}, Discount = {discount}%, Total to pay = {toPay}')
```

Q.4

```
def highestRating(ratingList) :
    d=dict()
    for f in ratingList:
        k=f[0]
        v=(f[2],f[1])
        if k not in d.keys():
            d[k]=v
        elif v[0]>d[k][0]:
            d[k]=v
    return d
ratingList=[["USB Switch 8", "e-max", 6] , ["MS Surface 9", "virgin", 8] ,
["Lenovo 24", "Amazon" , 9], ["MS Surface 9", "Alibaba" , 7], ["Lenovo 24",
"Alibaba", 8] ]
print(highestRating(ratingList))
```

Q.5

```
def magicSquare(intlist):
    n=len(intList)
    sr=[0]*n
    sc=[0]*n
    sd1=sd2=0
    for i in range(n):
        sd1+=intList[i][i]
        sd2+=intList[i][n-1-i]
        for j in range(n):
            sr[i]+=intList[i][j]
            sc[i]+=intList[j][i]
    if sd1!=sd2: return False
    for i in range(n):
        if sr[i]!=sd1: return False
        if sc[i]!=sd1: return False
    return True

intList=[[2,16,13,3],[11,5,8,10],[7,9,12,6],[14,4,1,15]]
print(magicSquare(intList))
```

Q.6

```
class Student:
    def __init__(self,name, quid, major, grades):
        self.name=name
        self.quid=quid
        self.major=major
        self.grades=grades
    def __str__(self):
        return f'name={self.name}, quid={self.quid}, mjor={self.major},
grades={self.grades}'

def f(students):
    d=dict()
    for s in students:
        d[s.quid]=sum(s.grades)
    return d

students=list()
s1=Student('Huda',200206364,'CS',[90,87,79,95,68])
s2=Student('Iman',200104812,'EE',[93,91,88,97])
s3=Student('Sarah',200200130,'CE',[75,73,68,81,83,77])
students.append(s1)
students.append(s2)
students.append(s3)
for s in students:
    print(s)
print(f(students))
```