In [ ]:
```
Name:P.Asha Belcilda
Rollno:225229104
Labsheet-6
```

Question1. Write a program for Password Management System

File creation: Ask user to enter N user names and their passwords. Store usernames and passwords into a file named “loginfile.txt”. Store each user and password in one line. File Processing: Write a program that opens your “security.txt” file and reads usernames and passwords from it. Store user names in one list and passwords in another lists. Querying: ask user to enter user name and password for verification. If they match the values stored in the lists, print a message “Login Successful”. Otherwise print a message “Login Failed, try again”.

In [3]:
```
#1
with open("loginfile.txt","w") as log:
   a=int(input("No of users:"))
   for i in range (a):
        b=input("Enter User Name:")
        c=input("Enter Password:")
        log.write(b)
        log.write(c)
log.close()
```

```
No of users:4
Enter User Name:Asha
Enter Password:1403
Enter User Name:Nisha
Enter Password:1710
Enter User Name:Belcilda
Enter Password:2905
Enter User Name:Rosilda
Enter Password:0612
```

In [4]:
```
logf=open("loginfile.txt","r")
secf=open("security.txt","w")
w=logf.read()
secf.write(w)
secf.close()
```

In [5]:
```
secf=open("security.txt","r")
print(secf.read())
secf.close()
```

```
Asha1403Nisha1710Belcilda2905Rosilda0612
```

In [6]:
```
sec=open("security.txt","r")
r=sec.read()
a=input("User Name:")
b=input("Password:")
if a+b in r:
    print("log-in Successful")
else:
    print("log_in Failed")
sec.close()
```

```
User Name:Asha
Password:1403
log-in Successful
```

Question2. Write a program for Student Performance Analysis

Create a text file, 'marks.txt', with N marks as floating point numbers. Open the file, read marks from it and compute and print the highest mark. If the user runs the program more than once you should not overwrite the previous text file – simply append the marks to the end of the file. Modify the above program so that it also prints Top-3 highest marks (Note: you may need to use list concept) Modify the above program so that it also prints the Lowest-3 marks.

In [7]:
```
a=open("marks.txt","a")
b=int(input("No of Values:"))
c=[]
for i in range (b):
    li=float(input("Enter value:"))
    c.append(li)
a.write(str(c))
a.close()
```

```
No of Values:5
Enter value:67
Enter value:99
Enter value:87
Enter value:59
Enter value:87
```

In [10]:
```
a=open("marks.txt","r")
a.read()
print("Top Marks:",max(c))
c.sort(reverse=True)
print("Top-3 Highest Marks:",(c[:3]))
c.sort()
print("Lowest-3 Marks:",(c[:3]))
print("Low Mark:",min(c))
a.close()
```

```
Top Marks: 99.0
Top-3 Highest Marks: [99.0, 87.0, 87.0]
Lowest-3 Marks: [59.0, 67.0, 87.0]
Low Mark: 59.0
```

Question3. Write a program for Stock Price Analysis

File Creation: Continually prompt a user for stock name, followed by price values for 5 days. Each row indicates stock name and daily prices of one stock. Store these values in a text file called “stock-prices.txt”. Open the file in Append Mode. Prompt message “Do you want to continue? “ and stop reading values accordingly. Then, you can close your file. File Processing: Now, open your file for processing. Print stock name, minimum price, maximum price and average price values. You can also print which day stock price was lowest in the week and which day stock price was highest. So, modify your print statement to print stock name, minimum price & day of minimum price, maximum price & day of maximum price and average price values. (Hint: Use enumerate to get index values)

In [35]:
```
while True:
    a=input("Stock name:")
    f=open("stock-prices.txt","w")
    f.write(a)
    f.write('\t')
    for i in range(4):
        b=input("Price:")
        f.write(b)
        f.write('\t')
    f.write('\n')
    l=input("Want to Continue (s for stop):")
    if l=="s":
        break
f.close()
```

```
Stock name:Honda
Price:1000
Price:987
Price:789
Price:999
Want to Continue (s for stop):s
```

In [43]:
```
for m in open("stock-prices.txt","r").readlines():
    low=[]
    avg=0
    ab=m.split()
    for i in range(1,4):
        low.append(int(ab[i]))
        mx=max(low)
        imx=low.index(mx)
        mn=min(low)
        imn=low.index(mn)
        avg=sum(low)/4
    print("Maximum Price",mx,"at day",imx+1)
    print("Minimum Price",mn,"at day",imn+1)
    print("Average Price:",avg)
```

```
Maximum Price 1000 at day 1
Minimum Price 789 at day 3
Average Price: 694.0
```

In [ ]:
```
```