



Project File
Computer Science
Session: 2022-2023

Submitted to-

Mrs. Khushboo Jain

Submitted by-

Aditya Goel
Roll no.-



Abstract:

The ultimate motto of this project is to help workplaces better organize employee's data and is built with the primary goal of keeping ease of access in mind and making it as accessible as possible.

Table of Contents

Abstract:	1
Certificate.....	3
Acknowledgement	4
Python-.....	5
Requirements:	6
Introduction.....	7
Code:	8
Part one containing basic functions used by the code:	8
Part two containing the UI related functions and code.	14
Output Screenshots:	20
Main Screen:	20
Inserting data with some errors caught:	21
Updating Values of previously entered record to fix misplaced values:	21
Deleting Data:	22
Searching Data:	22
Sorting Data:	26
Viewing reports of company:.....	31
Displaying the entire file at once:	33
Advantages:.....	34
Limitations:.....	35
Bibliography:	36

Certificate

This is to certify that **Aditya Goel**, a student of class **XII-A** has successfully completed the project titled “**Employee Management**” under the guidance of **Ms. Khushboo Jain** (subject teacher) during the academic year **2022-23** for partial fulfillment of computer science practical examination conducted by central board of secondary education.

Signature
(External Examiner)

Signature
(Internal Examiner)

Acknowledgement

*I would like to express my special thanks of gratitude to my teacher **Mrs Khushboo Jain** as well as our principal **Mrs Komal** who gave me the golden opportunity to do this wonderful project on the topic **Employee Management**, which also helped me in doing a lot of Research and i came to know about so many new things I am really thankful to them.*

Secondly i would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

Python-

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.

Advantages:

- 1.** Easy to use due to simple syntax rule.
- 2.** Interpreted language : code execution & interpretation line by line.
- 3.** Cross-Platform language : it can run on windows, Linux, Macintosh etc. equally.
- 4.** Expressive language : less code to be written as it itself express the purpose of the code.
- 5.** Completeness : support wide range of library.
- 6.** Free & Open source : can be downloaded freely and source code can be Modify for improvement.

Disadvantages:

- 1.** Lesser libraries as compared to other programming languages like c++, java,.net etc.
- 2.** Slow language : as it is interpreted languages, it executes the program slowly.
- 3.** Weak on type-binding : it does not pinpoint on use of a single variable for different datatype.

Requirements:

Hardware:

- Computer for coding and typing the required documents of the project.
- Printer to print the required documents of the project.
- Processor: Pentium Quad core or above
- Ram: 512 mb or higher
- Storage: 18 mb spaced required on disk

Software:

- Operating system: Windows 7 or above/ MacOS/ Linux
- Python 3.10 for executing the program
Suggested VScode/ pycharm/ Anaconda for better output screen and console.
- .csv module for editing and reading the .csv files.
- Suggested: A .csv editor like excel for ease of access.

Introduction

The object of this project on “**Employee Management**” is to attempt to provide a solution to store and access employee information in a digital and efficient way in easy-to-use manner. The code is written keeping in mind a wide range of possibilities and it will work with wide range of datasets with little tweaks to initial columns variable. It provides with all the functionalities like editing, inserting new records, generating smart reports etc while accounting for many errors that a user might make and fixing them automatically if possible or pointing them out. The project is written on commonly known language python(3.10) using .csv files which supports interpreters like excel which is commonly used.

I have seen lots of companies working with excel but some employee made silly errors like wrong date format etc which go unnoticed but however my system can catch some of those errors. It is not just limited to some one type of data management; it has inbuilt functions isolating the code in easy to read and understandable sections that allows a programmer to import parts of code easily that can be used in other projects.

I do acknowledge that there are already well-made softwares more suitable for such sort of data management, but this our best shot at it, and I think that it stands out on being more immune to errors.

Code:

Part one containing basic functions used by the code:

```
from statistics import mode
from csv import reader, writer

csvfile = "MOCK_DATA.csv"

_Sno_ = 0
_Name_ = 2
_DOB_ = 3
_Salary_ = 8
_Address_ = 6
_EmpID_ = 1
_Sector_ = 9
_Sex_ = 4
_email_ = 5
_DOJ_ = 10

# Used to print formatted records and automatically account for increased records.
def printer(i: list):
    try:
        print('{:<7}{:<10}{:<20}{:<12}{:<5}{:<35}{:<30}{:<15}{:<10}{:<25}{:<15}'.format(
            *i))
    except IndexError:
        print('{:<15}' * len(i)).format(*i)

# Used to get the file as a list
def filelist(file) -> list:
    with open(file) as temp:
        return list(reader(temp, delimiter=','))

# Writes a list to the file as a new record
def writelist_to_file(cont: list, rd: str = 'w'):
    with open(csvfile, rd, newline='') as temp:
        writer(temp, delimiter='|').writerows(cont)

def display(file): # Used to display the file
    for i in filelist(file):
        printer(i)

def displaylist(temp: list): # Display list in file format.
    for i in temp:
        printer(i)

# Converts date to yyyy-mm-dd from string
def managedate(date: str) -> list:
    date = list(int(j) for j in date.split('/', 2))
    return [date[2], date[0], date[1]] # YYYY-MM-DD

# Correct serial number of the file in list format if needed
def modifySnoList(var: list) -> list:
```

```

cnt = 0
for i in var:
    cnt += 1
    i[_Sno_] = cnt
return var

# Inserts a record at desired position, and at last if none mentioned
def insert(file, cont: list, ind=''):
    tempstor = filelist(file)
    if ind == '':
        index = int(tempstor[-1][_Sno_]) + 1
        writelist_to_file([[index, *cont]], 'a')
    else:
        try:
            tempstor.insert(int(ind), [ind] + cont)
            header = tempstor[0]
            tempstor = modifySNOlist(tempstor[1::])
            writelist_to_file([header, *tempstor])
        except ValueError:
            print('Please enter a serial no. as a integer')

# Updates the desired record
def updateSNO(file, cont: list, sno: int):
    tempstor = filelist(file)
    record = tempstor[sno][1::]
    for i in range(len(cont)):
        if str(cont[i]) == '0':
            cont[i] = record[i]

    tempstor[sno] = [sno] + cont
    writelist_to_file(tempstor)

# Deletes a record
def deletion(file, ind=None):
    tempstor = filelist(file)
    header = tempstor[0]
    if ind is None:
        print("Successfully deleted record: ")
        printer(header)
        printer(tempstor.pop())
        tempstor.remove(header)
    else:
        flag = True
        for i in tempstor:
            if i[_Sno_] == str(ind):
                print("Successfully deleted record: ")
                printer(header)
                printer(i)
                tempstor.remove(i)
                tempstor = modifySNOlist(tempstor[1::])
                flag = False
                break
        if flag:
            print(f'Can't find a record with mentioned serial number:{ind}')

    writelist_to_file([header, *tempstor])

# Searches a record with sno
def search_sno(file, sno: int, opr='1') -> int | None:
    tempstor = filelist(file)
    header = tempstor[0]
    flag = True
    if opr == "1":

```

```

for i in tempstor[1::]:
    if int(i[_Sno_]) == sno:
        printer(header)
        printer(i)
        return int(i[int(_Sno_)])
    return None

elif opr == "2":
    for i in tempstor[1::]:
        if int(i[_Sno_]) > sno:
            if flag:
                printer(header)
                flag = False
            printer(i)

elif opr == "3":
    for i in tempstor[1::]:
        if int(i[_Sno_]) < sno:
            if flag:
                printer(header)
                flag = False
            printer(i)
else:
    print('Please enter a reasonable operator with respect to ur input.')

# Searches the record smartly with any last name or first name automatically
def better_srch_name(file, name: str) -> int:
    tempstor = filelist(file)
    names = []
    for i in tempstor:
        names.append(i[_Name_].lower().split())
    flag = False
    flag1 = True
    for i in names:
        if name.lower() in i or name.lower() == ' '.join(i):
            indx: int = names.index(i)
            if flag1:
                printer(tempstor[0])
                flag1 = False
            printer(tempstor[indx])
            flag = True
    if flag:
        return indx
    else:
        print('Check your spelling.')

# Searches with emp id
def search_empid(file, parameter_empid: int) -> int | None:
    tempstor = filelist(file)
    for i in tempstor[1::]:
        if int(i[_EmpID_]) == parameter_empid:
            printer(tempstor[0])
            printer(i)
    return int(i[_Sno_])
print("\n Can't find a employee with mentioned employee id.")
return None

# automatically detects input from name, Sno., empid and search.
def auto_search(file, inp: int | str) -> int:
    try:
        inp = int(inp)
        if len(str(inp)) <= 3: # Sno. checker
            ind = search_sno(file, inp)
            if ind is None:

```

```

        print('Cant find someone with mentioned Sno.')
    return ind
else: # Emp ID assumer
    ind = search.empid(file, inp)
    if ind is None:
        print('Cant find someone with mentioned Employee ID.')
    return ind
except ValueError: # Name
    ind = better_srch_name(file, inp)
    if ind is None:
        print('Cant find someone with mentioned Name.')
    return ind

# Searches with given SEX of employee
def search_sex(file, sex: str):
    tempstor = filelist(file)
    flag = True
    for i in tempstor[1::]:
        if sex == i[_Sex_]:
            printer(i)
            flag = False
        elif sex == i[_Sex_]:
            printer(i)
            flag = False
        elif sex == i[_Sex_]:
            printer(i)
            flag = False
    if flag:
        print('Please enter a valid sex(M/F/O)')

# Searches with mentioned salary
def search_salary(file, parameter_salary: int, operator: str = '1'):
    tempstor = filelist(file)
    header = tempstor[0]
    flag = True
    flag1 = True
    for i in tempstor[1::]:
        if operator == "1":
            if int(i[_Salary_]) == parameter_salary:
                if flag1:
                    printer(header)
                    flag1 = False
                printer(i)
                flag = False
        elif operator == "2":
            if int(i[_Salary_]) > parameter_salary:
                if flag1:
                    printer(header)
                    flag1 = False
                printer(i)
                flag = False
        elif operator == "3":
            if int(i[_Salary_]) < parameter_salary:
                if flag1:
                    printer(header)
                    flag1 = False
                printer(i)
                flag = False
    if flag:
        print('Please enter a reasonable operator.')

# Search with dates
def srch_date(file, inp_date: str, opr="1"):
```

```

tempstor = filelist(file)
header = tempstor[0]
date = managedate(inp_date)
flag = True
flag1 = False
for i in tempstor[1::]:
    srtdate = managedate(i[_DOJ_])
    if str(opr) == '1':
        if srtdate == date:
            if flag1:
                printer(header)
                flag1 = False
            printer(i)
            flag = False
    elif str(opr) == '2':
        if srtdate > date:
            if flag1:
                printer(header)
                flag1 = False
            printer(i)
            flag = False
    elif str(opr) == '3':
        if srtdate < date:
            if flag1:
                printer(header)
                flag1 = False
            printer(i)
            flag = False
    if flag:
        print('Enter a reasonable date and ensure its in correct format.')
# Search by division
def search_divion(file, parameter_division: str):
    tempstor = filelist(file)
    flag = True
    for row in tempstor:
        if row[_Sector_].lower() == parameter_division.lower().rstrip(' '):
            if flag:
                printer(tempstor[0])
                flag = False
            printer(row)

# Sorts alphabetically
def sortalpha(file, desc=False):
    tempstor = filelist(file)

    def function(x: list) -> str:
        return x[_Name_]

    header = tempstor[0]
    tempstor.remove(header)
    tempstor = modifysnolist(sorted(tempstor, key=function, reverse=desc))
    sorteddata = [header, *tempstor]
    displaylist(sorteddata)
    return sorteddata

# Sorts by date
def sort_date(file, desc=False) -> list:
    tempstor = filelist(file)

    def function(x: list) -> list:
        return managedate(x[_DOJ_])

    header = tempstor[0]

```

```

tempstor.remove(header)
tempstor = modifysnolist(sorted(tempstor, key=function, reverse=desc))
sorteddata=[header, *tempstor]
displaylist(sorteddata)
return sorteddata

# Sorts by salary
def sort_salary(file, desc=False):
    tempstor = filelist(file)

    header = tempstor[0]
    tempstor.remove(header)

    def function(x: list) -> int:
        return int(x[_Salary_])

    tempstor = modifysnolist(sorted(tempstor, key=function, reverse=desc))
    sorteddata=[header, *tempstor]
    displaylist(sorteddata)
    return sorteddata

# Sorts by employee ID
def sort_empid(file, desc=False):
    tempstor = filelist(file)

    header = tempstor[0]
    tempstor.remove(header)

    def function(x: list) -> int:
        return int(x[_EmpID_])

    tempstor = sorted(tempstor, key=function, reverse=desc)

    tempstor = modifysnolist(tempstor)
    sorteddata=[header, *tempstor]
    displaylist(sorteddata)
    return sorteddata

# Makes a salary report
def better_salary_report(file):
    tempstor = filelist(file)
    sal = []
    for i in tempstor[1::]:
        sal.append(int(i[_Salary_]))
    print("SALARY REPORT"
          "\n-----\n"
          f"Total Paid Salary: {sum(sal)}\n"
          f"\n <Sal> (<Name>: <emp_id>)\n"
          f"Average Salary Paid:{(sum(sal) // len(sal))}\n"
          f"Highest Salary:{tempstor[sal.index(max(sal))][_Name_]} : "
          f"{tempstor[sal.index(max(sal))][_EmpID_]} \n"
          f"Lowest Salary:{tempstor[sal.index(min(sal))][_Name_]} : "
          f"{tempstor[sal.index(min(sal))][_EmpID_]}")

# Makes a report that displays divisions and employees that work in it
def better_rep_div_cnt(file):
    print("WORK DIVISON REPORT")
    print("-----\n")

    tempstor = filelist(file)
    tempstor.remove(tempstor[0])
    print(f"Total Employee Count: {int(tempstor[-1][_Sno_])} \n")
    for i in tempstor:

```

```

        i[_Sector_] = i[_Sector_].rstrip(' ')
div = []
for i in tempstor:
    div.append(i[_Sector_])
l = div.count(mode(div))
div = sorted(list(set(div)))
print('{:<25}' * len(div)).format(*list(i.upper() for i in div)))
data = list([i[_Name_], i[_Sector_]] for i in tempstor)

def function(x: list) -> str:
    return x[1]

data = sorted(data, key=function)
test = []
for j in range(l):
    TEST = [""] * len(div)
    keye = []
    for i in range(len(data)):
        if data[i][1] not in keye and data[i][0] not in test:
            keye.append(data[i][1])
            TEST[div.index(data[i][1])] = data[i][0]
            test.append(data[i][0])
    print('{:<25}' * len(div)).format(*TEST))

```

Part two containing the UI related functions and code.

```

from csv_file_func import *

print("You are now viewing the private database which keeps track of all the employees")
print("This program is going to help you navigate and work with this database")
print("x-----x\n")

# Used to get int input and filter out everything else
def getint(txt: str) -> int:
    while True:
        inte = ''.join(i for i in input(txt) if i.isdigit())
        if inte == '':
            print('Please enter valid integers i.e. 1 2 3 etc.')
        else:
            return int(inte)

# Used to get date in correct format and correct the mistakes of wrong format.
def getdate(DOB=True):
    global dob
    while True:
        if DOB:
            print("Enter employee's date of birth (MM/DD/YYYY): ", end=' ')
        else:
            print("Enter employee's date of joining (MM/DD/YYYY): ", end=' ')
        truedate = ''.join(i for i in input(':\\t') if i == '/' or i.isdigit())
        if truedate == '0':
            return truedate
        try:
            date = managedate(truedate)
        except:
            print("Please ensure to use '/' between dates.")
            continue
        if len(truedate) != 10:
            print('Please ensure to add a 0 before a single digit entry, for eg: 11/03/2005')
        elif date[2] > 31:

```

```

        print('Please enter the date correctly, input the DOB again.')
    elif date[1] > 12:
        print('Please enter the month correctly, input the DOB again.')
    elif DOB:
        if 2022 - date[0] < 18:
            print('Please ensure employee is 18+')
        elif 2022 - date[0] > 60:
            print("Please ensure employee is less than 60 years old due to
                  company's policy")
        else:
            dob = date
    return truedate
else:
    try:
        age = date[0] - dob[0]
    except:
        age = date[0] - 1975
    if 2022 - date[0] < 0:
        print('Please enter a date in past.')
    elif 61 < age or age < 17:
        print('Please ensure employee is in between 18-60 years of age.')
    else:
        return truedate

# Used to get sex inputted and account for many possible answers.
def getsex():
    while True:
        sex = input("Enter employee's sex (M/F/O):    ").upper()
        i: str
        sex = ''.join(i for i in sex if not i.isspace() and i.isalpha() or i == '0')
        if sex == '0':
            return sex
        if sex in ['MALE', 'M', 'MAN', 'BOY']:
            return 'M'
        elif sex in ['FEMALE', 'F', 'WOMAN', 'GIRL']:
            return 'F'
        elif sex in ['O', 'OTHER', 'OTHERS']:
            return 'O'
        else:
            print('Please enter a valid sex and check your spellings.')

# Used to get the employee id in the correct format
def getemp():
    while True:
        emp = getint("Enter employee id:\t")
        if len(str(emp)) < 4 and emp != 0:
            print("Please enter employee id of atleast 4 numeric digits")
        else:
            return emp

# Used to get a record inputted from the user.
def getinput() -> list:
    while True:
        inp = [gettemp(), # Employee ID
               input("Enter employee's full Name:    "), # Employee Name
               getdate(), # DOB
               getsex(), # Sex
               input("Enter employee's email id:    "), # Email id
               input("Enter employee's address:    "), # Address
               getint("Enter employee's phone number:    "), # Phone Number
               getint("Enter employee's salary:    "), # Employee Salary
               input("Enter employee's designation:    "), # Designation
               getdate(False)] # Date of joining
    print("Following entered values will be passed into database.")

```

```

    "\n(Don't worry 0 will be replaced with original value)")
printer(inp)
print("Are you satisfied with entered data?")
choice = getchoiceYorN()
if choice:
    return inp
else:
    continue

# Used to get simple yes/no inputs from user
def getchoiceYorN():
    while True:
        Choice = input(":      ")

        if Choice.lower() in ('y', 'yes', 'yup', 'oui', 'si'):
            return True
        elif Choice.lower() in ('n', 'no', 'nope', 'non'):
            return False
        else:
            print("\nEnter a valid response.\n")

# Used to sort the file in multiple ways
def choose_sort() -> bool | None:
    while True:
        print("How will you sort the data?"
              "\nType 1 for Sorting in ascending order or just leave blank."
              "\nType 2 for Sorting in descending order"
              "\nType 0 to go back to previous menu")
        choice = ''.join(i for i in input(":      ") if i.isdigit())
        if choice == '0':
            print("Going back.....\n")
            return None
        elif choice == '1' or choice == '':
            return False
        elif choice == '2':
            return True
        else:
            print("\nEnter a valid response.\n")

# Used to get a choice from the user.
def choose_opr() -> str:
    while True:
        opr = input("Choose the method of operation.\n"
                   "Type 0 to exit this menu\n"
                   "Type 1 for finding equal entries or just leave blank.\n"
                   "Type 2 for finding greater entries.\n"
                   "Type 3 for finding lesser entries.\n"
                   ":      ")
        opr = ''.join(i for i in opr if i.isdigit())
        if opr in ['0', '1', '2', '3']:
            return opr
        elif opr == '':
            return '1'
        else:
            print('Please enter a valid no. (1-3).')

# Used to get the index input from user withing range
def getind(txt: str, blank=False):
    file = filelist(csvfile)
    maxsno = file[-1][0]
    minsno = file[1][0]
    while True:
        ind = ''.join(i for i in input(txt) if i.isdigit())
        if ind == '' and blank:

```

```

        return ''
    elif ind == '' and not blank:
        return int(ind)
    elif int(minsno) <= int(ind) <= int(maxsno):
        return int(ind)
    else:
        print(f'Please enter a integer between {minsno}-{maxsno} ')

# Used to update the records
def updateUI():
    print("Enter Serial No. or Employee ID or Name to update record off.")
    update_meth = input(":      ")

    print("Following entry is going to be updated.")
    ind: int = auto_search(csvfile, update_meth)
    if ind is None:
        return None
    print('\nAre you sure? (Y/N)')
    choice = getchoiceYorN()
    if choice:
        print("Enter 0 if you don't want to update certain field.")

        updateSNO(csvfile, getinput(), ind)
        print('\n')
        search_sno(csvfile, ind)
        print('\n')
    else:
        print('OK updation canceled.')

```



```

# Used to delete the records
def delUI():
    print("Enter Serial No. or Employee ID or Name to delete record off, leave blank to
delete last record.")
    InpSno = input(":      ")
    if InpSno.strip(' ') == '':
        deletion(csvfile)
        return None
    print("Following entry/s are going to be deleted: ")
    ind = auto_search(csvfile, InpSno)
    if ind is None:
        return None
    print("\n Are you sure? (Y/N)")
    choice = getchoiceYorN()
    if choice:
        deletion(csvfile, ind)
        print("Entry/s has/have been successfully deleted.")
    else:
        print('OK deletion cancelled.')

```



```

# While loop for the menu
while True:
    print("\nWhat should I do for you?"
          "\nType 1 for Inserting data"
          "\nType 2 for Updating values of the data"
          "\nType 3 for Delete entries of the data"
          "\nType 4 for Searching the data"
          "\nType 5 for Viewing sorted data"
          "\nType 6 for Returning a report of the Company"
          "\nType 7 for Displaying the entire file at once."
          "\nType 0 for Exiting the program")
    MainInp = getint(":      ")
    if MainInp == 0:
        print("Exiting the program. Goodbye!")
        break

```

```

if MainInp == 1: # insertion
    print("Start entering the details of the employee one by one.")
    insert(csvfile,
           getinput(),
           getind('Enter the index no. u want to add the record to, ' # Index
                  'leave blank if you wish to add at the end: ', True))
    print('\nInsertion Complete\n')

elif MainInp == 2: # update
    updateUI()

elif MainInp == 3: # delete
    delUI()

elif MainInp == 4: # search
    while True:
        print("How would you like to search the data?"
              "\nType 1 for Searching by Serial Number"
              "\nType 2 for Searching by Employee's Name"
              "\nType 3 for Searching by Employee ID"
              "\nType 4 for Searching by Salary"
              "\nType 5 for Searching by Designation"
              "\nType 6 for Searching by Date of Joining"
              "\nType 7 for Searching by Sex"
              "\nType 8 to automatically detect Sno., Employee ID or Name."
              "\nType 0 to go back to previous menu")
        SecInp = getint(":      ")

        if SecInp == 0:
            print("Going back.....\n")
            break

        elif SecInp == 1:
            opr = choose_opr()
            if opr == '0':
                print("Going back.....\n")
                break
            search_sno(csvfile,
                       getind("Enter a serial number or leave blank to check last
entry:   "), opr)
            break

        #####-----Name-----
        elif SecInp == 2:
            better_srch_name(csvfile, input("Enter a first/last/full name:   "))
            break
        #####-----EmpId-----
        elif SecInp == 3:
            search.empid(csvfile, getemp())
            break
        #####-----Salary-----
        elif SecInp == 4:
            opr = choose_opr()
            if opr == '0':
                print("Going back.....\n")
                break
            search_salary(csvfile,
                          getint("Enter the Amount:   "), opr)
            break
        #####-----Designation-----
        elif SecInp == 5:
            search_divion(csvfile, input('Enter the division:   '))
            break
        #####-----DOJ40-----
        elif SecInp == 6:

```

```

opr = choose_opr()
if opr == '0':
    print("Going back.....\n")
    break
srch_date(csvfile,
           getdate(False), opr)
break

#####-----Sex-----
elif SecInp == 7:
    search_sex(csvfile, getsex())
    break
elif SecInp == 8:
    auto_search(csvfile, input("Enter any Sno., Employee ID or Name\n"
                           "      :   "))
#####-----Valid Response-----
else:
    print("\nEnter a valid response.\n")

##--Sorted function Start, updates the file only if users wants to.
elif MainInp == 5:
    while True:
        print("Choose your method of sorting\n"
              "Type 1 for Sorting by employee id\n"
              "Type 2 for Sorting by name\n"
              "Type 3 for Sorting by salary\n"
              "Type 4 for Sorting by date of joining\n"

              "Type 0 to go back to previous menu")
        SecInp = getint(":   ")

        #####-----Going Back-----
        if SecInp == 0:
            print("Going back.....\n")
            break
        #####-----EmpID-----
        if SecInp in [1, 2, 3, 4]:
            choice = choose_sort()
            if choice is None:
                break
        else:
            print("\nEnter a valid response.\n")
            continue

        #####-----Emp_ID-----
        if SecInp == 1:
            data = sort.empid(csvfile, choice)

        #####-----Name-----
        elif SecInp == 2:
            data = sortalpha(csvfile, choice)

        #####-----Salary-----
        elif SecInp == 3:
            data = sort_salary(csvfile, choice)

        #####-----Date-----
        elif SecInp == 4:
            data = sort_date(csvfile, choice)
        print("Would you like to update the file with the sorted data?")
        if getchoiceYorN():
            writelist_to_file(data)
            break
        else:
            break

```

```

# --Report function Start
elif MainInp == 6:
    print("\nWhich report should I present to you?\n"
        "Type 1 for getting a report on company's expenditure on Salary\n"
        "Type 2 for getting a report on company's work division\n"
        "Type 0 to go back to previous menu")
while True:
    SecInp = getint(":      ")
    #####-----Going Back-----
    if SecInp == 0:
        print("Going back.....\n")
        break

    elif SecInp == 1:
        better_salary_report(csvfile)
        break
    elif SecInp == 2:
        better_rep_div_cnt(csvfile)
        break
    else:
        print('Please enter a valid response')
elif MainInp == 7:
    display(csvfile)

else:
    print("\nEnter a valid response.\n")

```

Output Screenshots:

Main Screen:

You are now viewing the private database which keeps track of all the employees
This program is going to help you navigate and work with this database
X-----X

What should I do for you?
Type 1 for Inserting data
Type 2 for Updating values of the data
Type 3 for Delete entries of the data
Type 4 for Searching the data
Type 5 for Viewing sorted data
Type 6 for Returning a report of the Company
Type 7 for Displaying the entire file at once.
Type 0 for Exiting the program
: |

Inserting data with some errors caught:

```
: 1
Start entering the details of the employee one by one.
Enter employee id: 342
Please enter employee id of atleast 4 numeric digits
Enter employee id: 435235
Enter employee's full Name: Sam Willington
Enter employee's date of birth (MM/DD/YYYY): : 11/04/2020
Please ensure employee is 18+
Enter employee's date of birth (MM/DD/YYYY): : 11/04/1950
Please ensure employee is less than 60 years old due to company's policy
Enter employee's date of birth (MM/DD/YYYY): : 13/11/1996
Please enter the month correctly, input the DOB again.
Enter employee's date of birth (MM/DD/YYYY): : 11/13/1996
Enter employee's sex (M/F/O): male
Enter employee's email id: Sam Willington
Enter employee's address: sam45@gmail.com
Enter employee's phone number: 4532523123
Enter employee's salary: 45000
Enter employee's designation: Communication
Enter employee's date of joining (MM/DD/YYYY): : 11/10/2010
Please ensure employee is in between 18-60 years of age.
Enter employee's date of joining (MM/DD/YYYY): : 11/11/2020
Following entered values will be passed into database.
(Don't worry @ will be replaced with original value)
435235      Sam Willington 11/13/1996      M      Sam Willington sam45@gmail.com4532523123      45000      Communication 11/11/2020
Are you satisfied with entered data?
: yup
Enter the index no. u want to add the record to, leave blank if you wish to add at the end:

Insertion Complete
```

Updating Values of previously entered record to fix misplaced values:

```
Type @ for Exiting the program
: 2
Enter Serial No. or Employee ID or Name to update record off.
: SAM
Following entry is going to be updated.
S no. Emp_ID     Name          DOB        Sex   Email           Address          Phone No.    Pay(in rs)Designation      Date of joining
101  435235     Sam Willington  11/13/1996  M   Sam Willington  sam45@gmail.com  4532523123  45000  Communication  11/11/2020

Are you sure? (Y/N)
: Y
Enter @ if you don't want to update certain field.
Enter employee id: @
Enter employee's full Name: @
Enter employee's date of birth (MM/DD/YYYY): : @
Enter employee's sex (M/F/O): @
Enter employee's email id: sam45@gmail.com
Enter employee's address: 45 julio street
Enter employee's phone number: @
Enter employee's salary: @
Enter employee's designation: @
Enter employee's date of joining (MM/DD/YYYY): : @
Following entered values will be passed into database.
(Don't worry @ will be replaced with original value)
@      @      @      @      sam45@gmail.com45 julio street@      @      @
Are you satisfied with entered data?
: yes
```

Deleting Data:

```
: 3
Enter Serial No. or Employee ID or Name to delete record off, leave blank to delete last record.
: willington
Following entry/s are going to be deleted:
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
101 435235 Sam Willington 11/13/1996 M sam45@gmail.com 45 jilio street 4532523123 45000 Communication 11/11/2020

Are you sure? (Y/N)
: y
Successfully deleted record:
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
101 435235 Sam Willington 11/13/1996 M sam45@gmail.com 45 jilio street 4532523123 45000 Communication 11/11/2020
Entry/s has/have been successfully deleted.
```

Searching Data:

What should I do for you?

Type 1 for Inserting data

Type 2 for Updating values of the data

Type 3 for Delete entries of the data

Type 4 for Searching the data

Type 5 for Viewing sorted data

Type 6 for Returning a report of the Company

Type 7 for Displaying the entire file at once.

Type 0 for Exiting the program

```
: 4
```

How would you like to search the data?

Type 1 for Searching by Serial Number

Type 2 for Searching by Employee's Name

Type 3 for Searching by Employee ID

Type 4 for Searching by Salary

Type 5 for Searching by Designation

Type 6 for Searching by Date of Joining

Type 7 for Searching by Sex

Type 8 to automatically detect Sno., Employee ID or Name.

Type 0 to go back to previous menu

```
: 1
```

1. Searching by serial number:

```
Type 0 to go back to previous menu
: 1
Choose the method of operation.
Type 0 to exit this menu
Type 1 for finding equal entries or just leave blank.
Type 2 for finding greater entries.
Type 3 for finding lesser entries.
: 102
Please enter a valid no. (1-3).
Choose the method of operation.
Type 0 to exit this menu
Type 1 for finding equal entries or just leave blank.
Type 2 for finding greater entries.
Type 3 for finding lesser entries.
:
Enter a serial number or leave blank to check last entry: 102
Please enter a integer between 1-100
Enter a serial number or leave blank to check last entry: 69
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
69 82963 Octavia Drake 11/19/1983 M odrake3@tripod.com 903 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004
```

2. Searching by Employee's name:

```
Enter the employee id: 82963
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
69 82963 Octavia Drake 11/19/1983 M odrake3@tripod.com 903 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004
```

3. Searching by Employee ID:

```
: 2
Enter a first/last/full name: drake
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
69 82963 Octavia Drake 11/19/1983 M odrake3@tripod.com 903 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004
```

4. Searching by Salary:

```
: 4
Choose the method of operation.
Type 0 to exit this menu
Type 1 for finding equal entries or just leave blank.
Type 2 for finding greater entries.
Type 3 for finding lesser entries.
: 2
Enter the Amount: 168000
S no. Emp_ID Name DOB Sex Email Address Phone No. Pay(in rs)Designation Date of joining
2 41859 Adelaide Steljes 5/29/2000 M asteljes29@soundcloud.com 76856 Namekagon Alley 1302480774 176000 Consumer Durables 10/31/2005
3 62046 Aguste Keam 11/6/1986 M akeam1d@xing.com 78 Sherman Junction 0480161763 180000 Finance 5/19/2009
8 49093 Arline Baelde 6/9/1986 F abae1de2i@nbcnews.com 97719 Ryan Circle 7774794301 167000 Basic Industries 10/26/2013
13 93235 Byrle Baack 9/5/1996 M bbaack1a@reverbnation.com 1585 Sachtjen Alley 9472580343 167000 Health Care 4/11/2003
26 76484 Cyrus Petti 1/7/1996 M cpetti12@telegraph.co.uk 591 Vidon Point 7478276009 178000 Consumer Services 9/6/2006
45 85216 Josy Jackling 8/28/1998 F jjacklingd@auda.org.au 8954 Dakota Pass 1845340493 182000 Engineers 10/24/2006
46 91836 Kale Brahams 7/8/1995 F kbrahams1j@hhs.gov 5152 Merrick Point 6687383841 184000 Technology 4/5/2016
49 64795 Kenn Childe 1/14/1992 M kchilde2p@stumbleupon.com 51121 Kim Alley 3756039579 171000 Consumer Durables 1/29/2016
50 60515 Kennie Masterson 2/10/1990 F kmasterson1l@dedecms.com 8262 Debs Terrace 2893762131 208000 Technology 8/14/2003
56 68750 Lonnard Tointon 10/6/1987 F ltointon1i@nature.com 9 Eagle Crest Avenue 4560553157 197000 Capital Goods 1/6/2011
57 75036 Mackenzie Rawlinson 3/1/1987 M mrawlinson1w@bravesites.com 4446 Forest Run Street 1519607733 186000 Finance 5/6/2003
58 94433 Marietta O'Connolly 3/23/1995 F moconnollys@ft.com 8 Sunfield Pass 6795142914 165000 Engineers 8/22/2012
60 86503 Mendel Wainwright 7/15/1992 M mwainwrightp@storify.com 5 Del Mar Place 7186076602 181000 Engineers 10/4/2011
61 76446 Merna Schlagh 6/7/1994 M mschlaghy@arstechnica.com 82200 Transport Avenue 2500564312 167000 Engineers 3/22/2009
65 38993 Nedda Plackstone 6/29/1989 M nplackstoneo@hostgator.com 635 Eastwood Terrace 4037512459 183000 Technology 4/30/2019
67 55931 Nial Goodlet 8/26/1982 M ngoodlet2a@flavors.me 9103 Sunbrook Alley 7330210692 177000 Engineers 5/19/2012
```

5. Searching by division

Enter the division: finance										
S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)	Designation	Date of joining
1	65175	Addia Smyth	10/28/1985	F	asmythh@hubpages.com	947 Talisman Terrace	4131740184	87000	Finance	10/10/2005
3	62046	Aguste Keam	11/6/1986	M	akeam1d@xing.com	78 Sherman Junction	0480161763	180000	Finance	5/19/2009
5	46811	Alphonso Broxholme	1/19/1984	F	abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001
12	71441	Boycey Thorwarth	8/29/2000	F	bthorwarth1m@lulu.com	05 Graedel Road	9790634870	84000	Finance	12/7/2004
20	67020	Chuck Stollwerck	1/27/2000	F	cstollwerck2h@harvard.edu	0464 Park Meadow Drive	7457160396	68000	Finance	3/25/2009
30	71535	Fianna Giannassi	12/20/1987	F	fgiannassik@mediafire.com	72 Russell Junction	6831712150	114000	Finance	5/3/2002
37	85512	Gonzales Pinching	1/7/1988	F	gpinching0@java.com	85704 Mandrake Terrace	7016711056	74000	Finance	4/22/2006
40	46838	Jenni Padkin	5/25/1997	F	jpadkin1g@amazon.de	3 Hoepker Court	6554757449	111000	Finance	11/10/2008
53	35945	Kyle Quarton	11/20/1998	F	kquarton1x@narod.ru	464 Cordelia Terrace	3993345878	129000	Finance	12/28/2003
57	75036	Mackenzie Rawlinson	3/1/1987	M	mrawlinson1w@bravesites.com	4446 Forest Run Street	1519607753	186000	Finance	5/6/2003
62	93357	Micky Brosi	9/5/1983	M	mbrosiw@ca.gov	41193 David Way	2432786068	85000	Finance	9/15/2017
64	45024	Nadean Braunds	5/1/1987	F	nbraunds1t@cnet.com	6 Merrick Terrace	1319701809	134000	Finance	8/7/2009
82	37457	Samaria Dunster	12/12/1996	F	sdunster2q@eventbrite.com	98 Green Ridge Lane	6500173201	102000	Finance	2/22/2012
85	44208	Sari Bloxsome	3/13/1990	F	sbloxsome@csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001
89	68723	Sterling Abys	4/28/1988	F	sabys1r@amazonaws.com	13 Spohn Point	6067648288	37000	Finance	12/29/2004
94	46781	Tymon Burras	12/17/1991	M	tburras1u@wiley.com	982 Warbler Drive	1543451993	91000	Finance	12/7/2011
95	56302	Ursola Pedlingham	4/4/1994	F	upedlingham@goo.gl	6505 Gulseth Hill	2849059374	63000	Finance	9/16/2010

6. Searching by date of joining:

Choose the method of operation.									
Type 0 to exit this menu									
Type 1 for finding equal entries or just leave blank.									
Type 2 for finding greater entries.									
Type 3 for finding lesser entries.									
: 3									
Enter employee's date of joining (MM/DD/YYYY): : 01/01/2005									
5 46811 Alphonso Broxholme 1/19/1984 F abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001				
9 65852 Barbra Milroy 10/26/1989 M bmilroyq@issuu.com	49375 Menomonie Pass	2510480324	154000	Consumer Services	1/12/2003				
12 71441 Boycey Thorwarth 8/29/2000 F bthorwarth1m@lulu.com	05 Graedel Road	9790634870	84000	Finance	12/7/2004				
13 93235 Byrle Baack 9/5/1996 M bbaack1a@reverbnation.com	1585 Sachtjen Alley	9472588343	167000	Health Care	4/11/2003				
19 53518 Chucho Benning 3/26/1984 F cbenning1k@ibm.com	67505 Schurz Circle	9872713812	70000	Health Care	9/24/2001				
22 41158 Cordula O'Canovane 8/27/1990 M cocanovane9@github.com	07805 Birchwood Lane	1061499626	108000	Consumer Services	2/5/2004				
27 74760 Damiano Aldham 12/13/1999 M dalldham1f@netscape.com	3134 6th Hill	4855357235	123000	Energy	4/13/2003				
30 71535 Fianna Giannassi 12/20/1987 F fgiannassik@mediafire.com	72 Russell Junction	6831712150	114000	Finance	5/3/2002				
31 75576 Fielding Belden 4/26/1994 M fbelden2@mediafire.com	7 Redwing Hill	2300861011	49000	Technology	12/9/2001				

7. Searching by employee's sex:

:	7							
Enter employee's sex (M/F/O): girl								
1	65175	Addia Smyth	10/28/1985	F	asmynth@hubpages.com	947 Talisman Terrace	4131740184	87000 Finance 10/10/2005
5	46811	Alphonso Broxholme	1/19/1984	F	abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000 Finance 8/15/2001
7	61964	Angeline Goodere	8/14/1985	F	agoodere1v@creativecommons.org	1350 Goodland Crossing	9687618566	68000 Consumer Non-Durables 1/14/2014
8	49893	Arlene Baelde	6/9/1986	F	abaeldelz@nbcnews.com	97719 Ryan Circle	7774794301	167000 Basic Industries 10/26/2013
10	80455	Bernardo Tosh	9/1/1994	F	btosh21@arstechnica.com	827 Shopko Circle	9212211518	67000 Engineers 4/23/2013
12	71441	Boycey Thorwarth	8/29/2000	F	bthorwarth1m@lulu.com	05 Graedel Road	9790634870	84000 Finance 12/7/2004
15	71972	Cary Upchurch	10/25/1984	F	cupchurch2n@themeforest.net	3761 Drewry Park	1619602490	56000 Health Care 6/9/2009
17	81598	Cherie Lightbourne	3/6/2000	F	cightbourne@bigcartel.com	54 Vermont Plaza	8662140467	159000 Engineers 2/18/2005
19	53518	Chuchu Benning	3/26/1984	F	cbenning1k@ibm.com	67505 Schurz Circle	9872713812	54000 Health Care 9/24/2001
20	67020	Chuck Stollwerck	1/27/2000	F	cstollwerck2h@harvard.edu	0468 Park Meadow Drive	7457160396	68000 Finance 3/25/2009
24	83720	Cornell Sammars	11/10/1981	F	csammars7@rambler.ru	42695 Lotheville Circle	7803204392	95000 Engineers 1/23/2018
25	82773	Courtenay Larsen	8/9/1985	F	clarsen17@webnode.com	9356 Haas Street	8795651896	113000 Energy 12/12/2007
38	71535	Fianna Giannassi	12/20/1987	F	fgiannassik@mediavfire.com	72 Russell Junction	6831712150	114000 Finance 5/3/2002
32	61868	Filmore Cockarill	4/6/1988	F	fcockarillz@slate.com	9854 Dakota Drive	1361311487	54000 Engineers 3/21/2002
36	80854	Goldie Rockingham	12/13/1990	F	grockinham1q@chronoengine.com	84205 Mesta Street	8894517500	54000 Capital Goods 6/22/2006
37	85512	Gonzales Pinching	1/7/1988	F	gpincingh0@java.com	85704 Mandrake Terrace	7816711056	74000 Finance 4/22/2006
40	46838	Jenni Padkin	5/25/1997	F	jpadkin1g@amazon.de	3 Hoepker Court	6554757449	111000 Finance 11/10/2008
41	54186	Jesse Piercy	7/13/1996	F	jpierceyn@mozilla.org	2 Boyd Place	8164876887	136000 Consumer Services 5/29/2006
42	40134	Jillana Ubsdell	11/5/1997	F	jubsdel11c@princeton.edu	2 Hovda Way	4265193188	146000 Technology 8/11/2009
43	40764	Jojo Wickman	7/12/1981	F	jwickmant@bloglines.com	3669 Loftsgordon Lane	3614812979	45000 Health Care 9/23/2007
45	85216	Josy Jackling	8/28/1998	F	jjacklingd@auda.org.au	8954 Dakota Pass	1845340493	182000 Engineers 10/24/2006
46	91836	Kale Brahams	7/8/1995	F	kbrahams1j@hhs.gov	5152 Merrick Point	6687383841	184000 Technology 4/5/2016
58	60515	Kennie Masterson	2/10/1990	F	kmasterson1l@dedecms.com	8262 Debs Terrace	2893762131	200000 Technology 8/14/2003
51	94027	Konrad Americi	1/17/1982	F	kamericii@cyberchimps.com	1 Fairfield Plaza	6834466134	109000 Capital Goods 4/24/2009
53	35945	Kyle Quarton	11/20/1998	F	kquarton1x@narod.ru	464 Cordelia Terrace	3993345878	129000 Finance 12/28/2003
54	40278	Lazar Yegorkov	6/8/1986	F	lyegorkov2c@fc2.com	41 Gulseth Crossing	3130316094	84000 Engineers 3/13/2001
56	68750	Lonnard Tointon	10/6/1987	F	ltointon1i@nature.com	9 Eagle Crest Avenue	4560553157	197000 Capital Goods 1/6/2011
58	94433	Marietta O'Connolly	3/23/1995	F	moconnollys@ft.com	8 Sunfield Pass	6795142914	165000 Engineers 8/22/2012
59	63527	Melisandra Lilburne	5/30/1981	F	mlilburnel0@skyrack.com	743 Stephen Street	5900609170	84000 Health Care 4/26/2012

8. Automatic detection of Sno., name or emp_id:

Name:

:	8							
Enter any Sno., Employee ID or Name								
:	Drake	S no.	Emp_ID	Name	DOB	Sex	Email	Address Phone No. Pay(in rs)Designation Date of joining
69	82963	Octavia Drake			11/19/1983	M	odrake3@tripod.com	983 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004

Sno.:

Enter any Sno., Employee ID or Name								
:	69							
S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)Designation Date of joining
69	82963	Octavia Drake			11/19/1983	M	odrake3@tripod.com	983 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004

Emp ID:

Enter any Sno., Employee ID or Name								
:	82963							
S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)Designation Date of joining
69	82963	Octavia Drake			11/19/1983	M	odrake3@tripod.com	983 Tennessee Place 0581591011 170000 Capital Goods 1/3/2004

Sorting Data:

What should I do for you?

Type 1 for Inserting data

Type 2 for Updating values of the data

Type 3 for Delete entries of the data

Type 4 for Searching the data

Type 5 for Viewing sorted data

Type 6 for Returning a report of the Company

Type 7 for Displaying the entire file at once.

Type 0 for Exiting the program

: 5

Choose your method of sorting

Type 1 for Sorting by employee id

Type 2 for Sorting by name

Type 3 for Sorting by salary

Type 4 for Sorting by date of joining

Type 0 to go back to previous menu

1. Sorting by employee ID:

:	1									
How will you sort the data?										
Type 1 for Sorting in ascending order or just leave blank.										
Type 2 for Sorting in descending order										
Type 0 to go back to previous menu										
1	35164	Nettle Queyos	8/7/1997	F	nqueyosv@desdev.cn	26018 Melody Road	4725150134	148000	Health Care	1/9/2019
2	35945	Kyle Quarton	11/20/1998	F	kquarton1x@narod.ru	464 Cordelia Terrace	3993345878	129000	Finance	12/28/2003
3	36813	Carley Heineking	5/26/1990	M	cheineking2b@samsung.com	3 Golden Leaf Place	3427671550	100000	Health Care	8/10/2007
4	36839	Seana Rameau	1/31/1984	F	srameau6@multiply.com	035 Forest Junction	0642458065	171000	Basic Industries	2/23/2003
5	37457	Samarie Dunster	12/12/1996	F	sdunster2q@eventbrite.com	98 Green Ridge Lane	6500173201	102000	Finance	2/22/2012
6	38182	Kandace Aylen	6/22/1991	M	kaylen18@mtv.com	8859 Sommers Avenue	1395022658	53000	Basic Industries	11/27/2005
7	38572	Dulciana Willford	6/2/1992	M	dwillfordc@themeforest.net	650 Kipling Alley	0682617598	111000	Basic Industries	5/29/2019
8	38993	Nedda Plackstone	6/29/1989	M	nplackstoneo@hostgator.com	635 Eastwood Terrace	4037512459	183000	Technology	4/30/2019
9	40134	Jillana Ubsdell	11/5/1997	F	jubsdellic@princeton.edu	2 Hovde Way	4265193188	146000	Technology	8/11/2009
10	40278	Lazar Yegorkov	6/8/1986	F	lyegorov2c@fc2.com	41 Gulseth Crossing	3130316094	84000	Engineers	3/13/2001
11	40764	Jojo Wickman	7/12/1981	F	jwickmant@loglines.com	3669 Loftsgordon Lane	3614812979	45000	Health Care	9/23/2007
12	41158	Cordula O'Cannovane	8/27/1990	M	cocannovane9@github.com	07805 Birchwood Lane	1061499626	108000	Consumer Services	2/5/2004
13	41580	Otha Kee	3/7/1981	F	okee5@google.cn	92 Gulseth Street	2450040439	84000	Public Utilities	2/17/2016
14	41859	Adelaide Steljes	5/29/2000	M	aste1jes29@soundcloud.com	76856 Namekagon Alley	1302480774	176000	Consumer Durables	10/31/2005
15	42516	Gussie Page	5/18/1992	M	gpage2k@quantcast.com	125 Troy Parkway	4516615533	131000	Engineers	8/29/2006
16	43091	Nickolas Tabard	12/7/1985	F	ntabard1e@zimbio.com	4 Goodland Place	0188200517	193000	Health Care	11/9/2000
17	43691	Corilla MacDonough	12/1/1982	M	cmacdonough0@cam.ac.uk	5686 Nobel Circle	0352520361	39000	Capital Goods	3/1/2012
18	44025	Gaspar Hickinbottom	1/6/1993	M	ghickinbottom2g@clickbank.net	06277 Shoshone Circle	5419122162	35000	Consumer Services	2/29/2012
19	44208	Sari Bloxsom	3/13/1990	F	sbloxsbomb@csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001
20	45024	Nadean Braunds	5/1/1987	F	nbraundslt@net.com	6 Merrick Terrace	1319701889	134000	Finance	8/7/2009
21	46781	Tymon Burras	12/17/1991	M	tburraslu@wiley.com	982 Warbler Drive	1543451993	91000	Finance	12/7/2011
22	46811	Alphonso Broxholme	1/19/1984	F	abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001
23	46838	Jenni Padkin	5/25/1997	F	jpadkin1g@amazon.de	3 Hoepker Court	6554757449	111000	Finance	11/10/2008
24	47511	Rutter O'Day	4/13/1984	F	rodayr@senate.gov	4 Armistice Way	7113911897	168000	Engineers	6/4/2012

76	81768	Gherardo Switzer	6/23/1993	M	gswitzer2d@ameblo.jp	50180 Prentice Terrace	6076393416	143000	Consumer Non-Durables	4/28/2006
77	82773	Courtenay Larsen	8/9/1985	F	clarsen17@webnode.com	9356 Haas Street	8795651896	113000	Energy	12/12/2007
78	82963	Octavia Drake	11/19/1983	M	odrake3@tripod.com	903 Tennessee Place	0581591011	170000	Capital Goods	1/3/2004
79	83697	Sylvester Neissen	1/4/1990	M	sneissen1@nymag.com	714 Randy Lane	1227401957	182000	Basic Industries	1/18/2007
80	83720	Cornell Sammars	11/10/1981	F	csammars7@rambler.ru	42695 Lotheville Circle	7803204392	95000	Engineers	1/23/2018
81	84705	Willette Jayume	5/8/1986	M	wjayume28@dublogs.org	1398 Oriole Place	8182195284	145000	Engineers	10/12/2016
82	84905	Alistair Low	9/16/1989	M	alow1z@alibaba.com	4 Morning Avenue	0974406384	113000	Consumer Services	1/12/2007
83	85216	Josy Jackling	8/28/1998	F	jjacklingd@auda.org.au	8954 Dakota Pass	1845340493	182000	Engineers	10/24/2006
84	85512	Gonzales Pinching	1/7/1988	F	gpinchingu@java.com	85704 Mandrake Terrace	7016711056	74000	Finance	4/22/2006
85	86503	Mendel Wainwright	7/15/1992	M	mwainwrightp@storify.com	5 Del Mar Place	7186076602	181000	Engineers	10/4/2011
86	87465	Oriana De Meis	10/15/1991	F	ode1p@myspace.com	83768 Kensington Way	9437537224	111000	Health Care	3/7/2016
87	87823	Coraline Devanney	9/23/1998	M	cdevanney2l@prweb.com	793 Killdeer Point	4170999045	87000	Basic Industries	5/5/2017
88	88260	Robbi Kasbye	8/7/1988	F	rkasbye13@skuidoo.com	36022 Sunbrook Crossing	8890821759	181000	Public Utilities	9/7/2002
89	89021	Cate Heino	8/28/1990	M	cheino20@youtube.com	1 Judy Trail	1202418831	78000	Consumer Services	9/29/2015
90	90100	Jeana Alliband	1/19/1993	M	jalliband1b@sfgate.com	116 Commercial Trail	3426993333	103000	Health Care	7/18/2006
91	90179	Reinhold Delue	10/13/1987	F	rdelu23@unesco.org	62 Killdeer Avenue	7743045935	59000	Consumer Services	9/19/2016
92	90473	Germana Ellice	8/23/1991	M	gellice14@reverbnation.com	50762 La Follette Way	2307094091	78000	Miscellaneous	8/30/2012
93	91730	Tucky Lombardo	12/29/1987	M	tlombardo1n@plala.or.jp	88115 Bluestem Plaza	3831318948	185000	Engineers	4/8/2008
94	91836	Kale Brahams	7/8/1995	F	kbrahams1j@rhs.gov	5152 Merrick Point	6687383841	184000	Technology	4/5/2016
95	93235	Byrle Baack	9/5/1996	M	bbaack1a@reverbnation.com	1585 Sachtjen Alley	9472580343	167000	Health Care	4/11/2003
96	93357	Micky Brosi	9/5/1983	M	mbrosi@ca.gov	41193 David Way	2432786068	85000	Finance	9/15/2017
97	94027	Konrad Americi	1/17/1982	F	kamericii@cyberchimps.com	1 Fairfield Plaza	6834466134	109000	Capital Goods	4/24/2009
98	94433	Marietta O'Connolly	3/23/1995	F	moconnollys@ft.com	8 Sunfield Pass	6795142914	165000	Engineers	8/22/2012
99	94614	Emalee Passingham	4/29/1989	M	epassingham26@utexas.edu	04854 Union Way	0470075880	129000	Engineers	10/9/2008
100	99496	Pammie Ibanez	11/22/1999	F	pibanezm@weebly.com	3064 Cottonwood Alley	3766021648	49000	Consumer Services	9/5/2002

Would you like to update the file with the sorted data?

: nope

2. Sorting by Name:

How will you sort the data?										
Type 1 for Sorting in ascending order or just leave blank.										
Type 2 for Sorting in descending order										
Type 0 to go back to previous menu										
S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)	Designation	Date of joining
1	84705	Willette Jayume	5/8/1986	M	wjayume28@edublogs.org	1398 Oriole Place	8182195284	145000	Engineers	10/12/2016
2	81683	Willette Clementi	10/8/1982	F	wclementi16@alexa.com	60 Hermina Lane	6460742170	164000	Capital Goods	9/22/2005
3	77166	Wendell Ibanez	3/31/1989	F	wibanez20@iigo.com	122 Monterey Avenue	2609680554	74000	Consumer Services	6/21/2016
4	73202	Wash Setterfield	9/16/1991	M	wsetterfield10@indiegogo.com	7 Monument Trail	6745081059	51000	Engineers	4/29/2018
5	48668	Vyky Rolin	10/20/1984	M	vrolin2j@cornell.edu	27 Village Road	8719110041	195000	Engineers	8/14/2002
6	56302	Ursola Pedlingham	4/4/1994	F	upedlingham@goo.gl	6505 Gulseth Hill	2849059374	63000	Finance	9/16/2010
7	46781	Tymon Burras	12/17/1991	M	tburras1u@wiley.com	982 Warbler Drive	1543451993	91000	Finance	12/7/2011
8	91730	Tucky Lombardo	12/29/1987	M	tlombardo1n@plala.or.jp	88115 Bluestem Plaza	3831318948	185000	Engineers	4/8/2008
9	51949	Torrance MacTeague	10/29/1998	F	tmacteague27@fema.gov	80 Tennyson Terrace	8847498635	70000	Consumer Non-Durables	6/19/2008
10	60286	Talia Bartoszewicz	11/21/1995	M	tbartoszewicz15@businesswire.com	7 Spohn Place	8762283750	38000	Consumer Non-Durables	7/4/2002
11	83697	Sylvester Neissen	1/4/1990	M	sneissen1@nymag.com	714 Randy Lane	1227401957	182000	Basic Industries	1/18/2007
12	68723	Sterling Abys	4/28/1988	F	sabys1r@amazonaws.com	13 Spohn Point	6067648288	37000	Finance	12/29/2004
13	78842	Shaine Pilsworth	3/24/1982	O	spilsworthf@tripadvisor.com	865 Warner Avenue	3581395258	147000	Technology	4/28/2010
14	36839	Seana Rameau	1/31/1984	F	srameau6@multiply.com	035 Forest Junction	0642458065	171000	Basic Industries	2/23/2003
15	80982	Sayer Skinn	7/7/1983	M	sskinn2m@newyorker.com	0372 Division Lane	0914664980	102000	Engineers	1/28/2008
16	44208	Sari Bloxson	3/13/1998	F	sbloxsonb@csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001
17	70875	Sarah Mesant	1/17/1997	F	smesant24@cpanel.net	756 Ludington Terrace	3117347569	120000	Capital Goods	7/1/2013
18	49345	Sandi Son	1/10/1986	F	ssonl@simplemachines.org	25137 Melrose Place	3690264634	139000	Technology	4/13/2013
19	37457	Samaria Dunster	12/12/1996	F	sdunster2q@eventbrite.com	98 Green Ridge Lane	6500173201	102000	Finance	2/22/2012
20	47511	Rutter O'Day	4/13/1984	F	roday@senate.gov	4 Armistice Way	7113911897	168000	Engineers	6/4/2012
21	60745	Russ Samper	4/3/1994	M	rsampera@instagram.com	805 Union Lane	1104384205	165000	Consumer Services	9/16/2004
22	81452	Rosaline Ackwood	6/26/2000	F	rackwood4@jalbum.net	83261 Lawn Place	6115150027	115000	Engineers	4/23/2005
23	49782	Rora Battson	5/27/1982	M	rbattson8@psu.edu	5517 Grover Crossing	1365219631	139000	Engineers	2/15/2009
24	88260	Robbi Kasbye	8/7/1988	F	rkasbye13@squidoo.com	36022 Sunbrook Crossing	8899521759	181000	Public Utilities	9/7/2002
25	72757	Rey Balmforth	11/9/1986	F	rbalmforth19@marriott.com	629 Laurel Pass	8727621297	110000	Engineers	9/29/2015
26	90179	Reinhold Delue	10/13/1987	F	rdelue23@unesco.org	62 Killdeer Avenue	7743045935	59000	Consumer Services	9/19/2016

72	94614	Emalee Passingham	4/29/1989	M	epassingham26@utexas.edu	04854 Union Way	0470075880	129000	Engineers	10/9/2008
73	38572	Dulciana Willford	6/2/1992	M	dwillfordc@themeforest.net	650 Kipling Alley	0682617598	111000	Basic Industries	5/29/2019
74	74760	Damiano Aldham	12/13/1999	M	daldham1f@netscape.com	3134 6th Hill	4855357235	123000	Energy	4/13/2003
75	76484	Cyrillus Petti	1/7/1996	M	cpetti22@telegraph.co.uk	591 Vidon Point	7470276009	178000	Consumer Services	9/6/2006
76	82773	Courtenay Larsen	8/9/1985	F	clarsen17@webnode.com	9356 Haas Street	8795651896	113000	Energy	12/12/2007
77	83720	Cornell Sammars	11/10/1981	F	csammars1@rambler.ru	42695 Lotheville Circle	7803204392	95000	Engineers	1/23/2018
78	43691	Corilla McDonough	12/1/1982	M	cmcdonough0@cam.ac.uk	5686 Nobel Circle	0352520361	39000	Capital Goods	3/1/2012
79	41158	Cordula O'Cannovane	8/27/1990	M	cocannovane9@github.com	07805 Birchwood Lane	1061499626	108000	Consumer Services	2/5/2004
80	87823	Coraline Devanney	9/23/1998	M	cdevanney2l@prweb.com	793 Killdeer Point	4170999845	87000	Basic Industries	5/5/2017
81	67020	Chuck Stollwerck	1/27/2000	F	cstollwerck2h@harvard.edu	0468 Park Meadow Drive	7457160396	68000	Finance	3/25/2009
82	53518	Chuco Benning	3/26/1984	F	cbenning1k@ibm.com	67505 Schurz Circle	9872713812	70000	Health Care	9/24/2001
83	71840	Christa Clingoe	6/18/1999	M	cclingoe1h@psu.edu	7 Schmedeman Junction	4165640735	160000	Engineers	4/8/2016
84	81598	Cherie Lightbourne	3/6/2000	F	cightbourne@bigcartel.com	54 Vermont Plaza	8662140467	159000	Engineers	2/18/2005
85	89021	Cate Heino	8/28/1990	M	cheino20@youtube.com	1 Judy Trail	1202418031	78000	Consumer Services	9/29/2015
86	71972	Cary Upchurch	10/25/1984	F	cupchurch2n@themeforest.net	3761 Drewry Park	1619602490	56000	Health Care	6/9/2009
87	36813	Carley Heiniking	5/26/1990	M	cheiniking2b@samsung.com	3 Golden Leaf Place	3427671550	100000	Health Care	8/10/2007
88	93235	Byrle Baack	9/5/1996	M	bbaack1@reverbnation.com	1585 Sachtjen Alley	9472580343	167000	Health Care	4/11/2003
89	71441	Boycey Thorwarth	8/29/2000	F	bthorwarth1m@lulu.com	05 Graedel Road	9790634870	84000	Finance	12/7/2004
90	68669	Bo Lazonby	4/7/1994	M	blazonby2r@auda.org.au	8216 Hayes Hill	6977892552	102000	Energy	4/5/2010
91	80455	Bernardo Tosh	9/1/1994	F	btosh21@arstechnica.com	827 Shopko Circle	9212211518	67000	Engineers	4/23/2013
92	65852	Barbra Milroy	10/26/1989	M	bmilroyq@issuu.com	49375 Menomonie Pass	2510488324	154000	Consumer Services	1/12/2003
93	49093	Arlene Baele	6/9/1986	F	abaelde2i@nbcnews.com	97719 Ryan Circle	7774794301	167000	Basic Industries	10/26/2013
94	61964	Angelique Goodere	8/14/1985	F	agoodere1v@creativecommons.org	1350 Goodland Crossing	9687618566	68000	Consumer Non-Durables	1/14/2014
95	79780	Andie Meader	11/18/1996	M	ameader2e@hp.com	47702 Katie Way	3833425482	153000	Engineers	2/12/2011
96	46811	Alphonse Broxholme	1/19/1984	F	abroxholme@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001
97	84905	Alistair Low	9/16/1989	M	alow1z@alibaba.com	4 Morning Avenue	0974406384	113000	Consumer Services	1/12/2007
98	62046	Aguste Keam	11/6/1986	M	akeam1d@xing.com	78 Sherman Junction	0480161763	180000	Finance	5/19/2009
99	41859	Adelaide Steljes	5/29/2000	M	asteljes29@soundcloud.com	76856 Namekagon Alley	1302480774	176000	Consumer Durables	10/31/2005
100	65175	Addia Smyth	10/28/1985	F	asmithh@hubpages.com	947 Talisman Terrace	4131740184	87000	Finance	10/10/2005

Would you like to update the file with the sorted data?

:

3. Sorting by salary:

How will you sort the data?												
Type 1 for Sorting in ascending order or just leave blank.												
Type 2 for Sorting in descending order												
Type 0 to go back to previous menu												
S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)	Designation	Date of joining		
1	60515	Kennie Masterson	2/10/1990	F	kmasterson11@dedecms.com	8262 Debs Terrace	2893762131	200000	Technology	8/14/2003		
2	68750	Lonnard Tointon	10/6/1987	F	ltointon1i@nature.com	9 Eagle Crest Avenue	4560553157	197000	Capital Goods	1/6/2011		
3	48668	Vyky Rolin	10/28/1984	M	vrolin2j@cornell.edu	27 Village Road	0719110841	195000	Engineers	8/14/2002		
4	43091	Nickolas Tabard	12/7/1985	F	ntabard1e@zimbio.com	4 Goodland Place	0188200517	193000	Health Care	11/9/2000		
5	75036	Mackenzie Rawlinson	3/1/1987	M	mrawlinson1w@bravesites.com	4446 Forest Run Street	1519607733	186000	Finance	5/6/2003		
6	91730	Tucky Lombardo	12/29/1987	M	tlombardoin@lala.or.jp	8815 Bluestem Plaza	3831318948	185000	Engineers	4/8/2008		
7	91836	Kale Brahams	7/8/1995	F	kbrahams1j@hhs.gov	5152 Merrick Point	6687383841	184000	Technology	4/5/2016		
8	38993	Nedda Plackstone	6/29/1989	M	nplackstoneo@hostgator.com	635 Eastwood Terrace	4037512459	183000	Technology	4/30/2019		
9	85216	Josy Jackling	8/28/1998	F	jjackling1d@auda.org.au	8954 Dakota Pass	1845340493	182000	Engineers	10/24/2006		
10	83697	Sylvester Neissen	1/4/1990	M	sneissen11@nymag.com	714 Randy Lane	1227401957	182000	Basic Industries	1/18/2007		
11	86503	Mendel Wainwright	7/15/1992	M	mwainwrightp@stirify.com	5 Del Mar Place	7186076602	181000	Engineers	10/4/2011		
12	88260	Robbi Kasbye	8/7/1988	F	rkasbye13@squidoo.com	36022 Sunbrook Crossing	8890521759	181000	Public Utilities	9/7/2002		
13	62846	Agustie Keam	11/6/1986	M	akeam1d@xing.com	78 Sherman Junction	0480161763	180000	Finance	5/19/2009		
14	76484	Cyrillus Pettit	1/7/1996	M	cpettit12@telegraph.co.uk	591 Vidon Point	7470276009	178000	Consumer Services	9/6/2006		
15	55931	Nial Goodlet	8/26/1982	M	ngoodlet2a@flavors.me	9103 Sunbrook Alley	7330210692	177000	Engineers	5/19/2012		
16	41859	Adelaide Steljes	5/29/2000	M	aste1jes29@soundcloud.com	76856 Namekagon Alley	1302480774	176000	Consumer Durables	10/31/2005		
17	64795	Kenn Childes	1/14/1992	M	kchildes2p@stumbleupon.com	51121 Kim Alley	3756039579	171000	Consumer Durables	1/29/2016		
18	36839	Seana Rameau	1/31/1984	F	srameau6@multiply.com	035 Forest Junction	0642458065	171000	Basic Industries	2/23/2003		
19	82963	Octavia Drake	11/19/1983	M	odrake3@tripod.com	903 Tennessee Place	0581591011	170000	Capital Goods	1/3/2004		
20	47511	Rutter O'Day	4/13/1984	F	rodayr@senate.gov	4 Armistice Way	7113911897	168000	Engineers	6/4/2012		
21	49893	Arline Baelde	6/9/1986	F	abaelde2i@nbcnews.com	97719 Ryan Circle	7774794301	167000	Basic Industries	10/26/2013		
22	93235	Byrle Baack	9/5/1996	M	bbaack1a@reverbnation.com	1585 Sachjen Alley	9472580343	167000	Health Care	4/11/2003		
23	76446	Merna Schlagh	6/7/1994	M	mschlaghy@arstechnica.com	82200 Transport Avenue	2500564312	167000	Engineers	3/22/2009		
24	94433	Marietta O'Connolly	3/23/1995	F	moconnollys@ft.com	8 Sunfield Pass	6795142914	165000	Engineers	8/22/2012		
71	63527	Melisandra Lilburne	5/30/1981	F	mlilburne1o@skyrock.com	743 Stephen Street	5900609178	84000	Health Care	4/26/2012		
72	41580	Otha Kee	3/7/1981	F	okeef@google.cn	92 Gulseth Street	2450040439	84000	Public Utilities	2/17/2016		
73	89021	Cate Heino	8/28/1990	M	cheino20@youtube.com	1 Judy Trail	1202418031	78000	Consumer Services	9/29/2015		
74	90473	Germana Ellice	8/23/1991	M	geillice14@reverbnation.com	50762 La Follette Way	2307094091	78000	Miscellaneous	8/30/2012		
75	54190	Lise Newcom	11/21/1990	O	lnewcom12@nps.gov	400 East Plaza	4896026454	78000	Engineers	5/17/2008		
76	85512	Gonzales Pinching	1/7/1988	F	gpinchingu@java.com	85704 Mandrake Terrace	7016711056	74000	Finance	4/22/2006		
77	77166	Wendell Ibanez	3/31/1989	F	wibanez20@ddigo.com	122 Monterey Avenue	2609680554	74000	Consumer Services	6/21/2016		
78	68782	Kelwin Angelini	10/15/1994	M	kangelini2f@lycos.com	15795 Hanover Parkway	9986562058	73000	Health Care	8/16/2002		
79	53518	Chucho Benning	3/26/1984	F	cbenning1k@ibm.com	67505 Schurz Circle	9872713812	70000	Health Care	9/24/2001		
80	51949	Torrance MacLeague	10/29/1998	F	tmacteague27@fema.gov	00 Tennyson Terrace	8847498635	70000	Consumer Non-Durables	6/19/2008		
81	61964	Angelique Goodere	8/14/1985	F	agodere1v@creativecommons.org	1350 Goodland Crossing	9687618566	68000	Consumer Non-Durables	1/14/2014		
82	67028	Chuck Stollwerck	1/27/2000	F	cstollwerck2h@harvard.edu	0468 Park Meadow Drive	7457160396	68000	Finance	3/25/2009		
83	80455	Bernardo Tosh	9/1/1994	F	btosh21@arstechnica.com	827 Shopko Circle	9212211518	67000	Engineers	4/23/2013		
84	56302	Ursola Pedlington	4/4/1994	F	upedlingtonham@goo.gl	6505 Gulseth Hill	2849059374	63000	Finance	9/16/2010		
85	49693	Jori Jertz	7/19/1999	M	jjertz1s@histats.com	316 Grayhawk Crossing	9170999538	62000	Technology	10/29/2008		
86	90179	Reinhold Delue	10/13/1987	F	rdelue23@unesco.org	62 Killdeer Avenue	7743045935	59000	Consumer Services	9/19/2016		
87	71972	Cary Upchurch	10/25/1984	F	cupchurch2n@themeforest.net	3761 Drewry Park	1619602498	56000	Health Care	6/9/2009		
88	61868	Filmore Cockarill	4/6/1988	F	fcockarillz@slate.com	9854 Dakota Drive	1361311487	54000	Engineers	3/21/2002		
89	80854	Goldie Rockingham	12/13/1990	F	grockingham1q@chronoengine.com	84205 Mesta Street	8894517500	54000	Capital Goods	6/22/2006		
90	38182	Kandace Aylen	6/22/1991	M	keyalen18@mtv.com	8859 Sommers Avenue	1395022658	53000	Basic Industries	11/27/2005		
91	73202	Wash Setterfield	9/16/1991	M	wsetterfield10@indiegogo.com	7 Monument Trail	6745081059	51000	Engineers	4/29/2018		
92	75576	Fielding Belden	4/26/1994	M	fbelden2@mediafire.com	7 Redwing Hill	2300861011	49000	Technology	12/9/2001		
93	99496	Pammie Ibanez	11/22/1999	F	pibanez@weebly.com	3064 Cottonwood Alley	3766021648	49000	Consumer Services	9/5/2002		
94	40764	Jojo Wickman	7/12/1981	F	jwickmant@bloglines.com	3669 Loftsrgordon Lane	3614812979	45000	Health Care	9/23/2007		
95	44288	Sari Bloxson	3/13/1990	F	sbloxsom1csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001		
96	78746	Olenka Bagge	8/14/1984	F	obaggej@furl.net	99 8th Parkway	4214670671	40000	Technology	12/30/2012		
97	43691	Corilla MacDonough	12/1/1982	M	cmacdonough0@cam.ac.uk	5686 Nobel Circle	0352520361	39000	Capital Goods	3/1/2012		
98	60286	Talia Bartoszewicz	11/21/1995	M	tbartoszewicz15@businesswire.com	7 Spohn Place	0762283758	38000	Consumer Non-Durables	7/4/2002		
99	68723	Sterling Abys	4/28/1988	F	sabys1@amazonaws.com	13 Spohn Point	6067648288	37000	Finance	12/29/2004		
100	44025	Gaspar Hickinbottom	1/6/1993	M	ghickinbottom2g@clickbank.net	06277 Shoshone Circle	5419122162	35000	Consumer Services	2/29/2012		

Would you like to update the file with the sorted data?

:

no

4. Sorting by date of joining:

S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)	Designation	Date of joining
1	43091	Nickolas Tabard	12/7/1985	F	ntabard1e@zimbio.com	4 Goodland Place	0188200517	193000	Health Care	11/9/2000
2	40278	Lazar Yegorkov	6/8/1986	F	lyegorkov2c@fc2.com	41 Gulseth Crossing	3130316094	84000	Engineers	3/13/2001
3	46811	Alphonso Broxholme	1/19/1984	F	abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001
4	53518	Chucho Benning	3/26/1984	F	cbenning1k@ibm.com	67505 Schurz Circle	9872713812	70000	Health Care	9/24/2001
5	44208	Sari Bloxson	3/13/1990	F	sbloxsonb@csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001
6	75576	Fielding Belden	4/26/1994	M	fbelden2@mediafire.com	7 Redwing Hill	2300861011	49000	Technology	12/9/2001
7	61868	Filmore Cockarill	4/6/1988	F	fcockarill2@slate.com	9854 Dakota Drive	1361311487	54000	Engineers	3/21/2002
8	71535	Fianna Giannassi	12/20/1987	F	fgiannassisi@mediafire.com	72 Russell Junction	6831712150	114000	Finance	5/3/2002
9	60286	Talia Bartoszewicz	11/21/1995	M	tbartoszewicz15@businesswire.com	7 Spohn Place	0762283750	38000	Consumer Non-Durables	7/4/2002
10	48668	Vyky Rolin	10/20/1984	M	vrolin2j@cornell.edu	27 Village Road	07191108041	195000	Engineers	8/14/2002
11	68782	Kelwin Angelini	10/15/1994	M	kangelini2f@lycos.com	15795 Hanover Parkway	9986562058	73000	Health Care	8/16/2002
12	99496	Pammie Ibanez	11/22/1999	F	pibanezm@weebly.com	3064 Cottonwood Alley	3766021648	49000	Consumer Services	9/5/2002
13	88260	Robbi Kasbye	8/7/1988	F	rkasbye13@squidoo.com	36022 Sunbrook Crossing	8890521759	181000	Public Utilities	9/7/2002
14	65852	Barbra Milroy	10/26/1989	M	bmilroyq@issuu.com	49375 Monomone Pass	2510840324	154000	Consumer Services	1/12/2003
15	36839	Seana Rameau	1/31/1984	F	srameau6@multiply.com	835 Forest Junction	0642458065	171000	Basic Industries	2/23/2003
16	93235	Byrle Baack	9/5/1996	M	bbaack1a@reverbnation.com	1585 Sachtjen Alley	9472580343	167000	Health Care	4/11/2003
17	74760	Damiano Aldham	12/13/1999	M	daldham1f@netscape.com	3134 6th Hill	4855357235	123000	Energy	4/13/2003
18	75036	Mackenzie Rawlinson	3/1/1987	M	mrawlinson1w@bravesites.com	4446 Forest Run Street	1519607733	186000	Finance	5/6/2003
72	63527	Melisandra Lilburne	5/30/1981	F	mlilburne1o@skyrock.com	743 Stephen Street	5900609170	84000	Health Care	4/26/2012
73	55931	Nial Goodlet	8/26/1982	M	ngoodlet2a@flavors.me	9103 Sunbrook Alley	7330210692	177000	Engineers	5/19/2012
74	47511	Rutter O'Day	4/13/1984	F	rodayr@senate.gov	4 Armistice Way	7113911897	168000	Engineers	6/4/2012
75	94433	Marietta O'Connolly	3/23/1995	F	moconnollys@ft.com	8 Sunfield Pass	6795142914	165000	Engineers	8/22/2012
76	90473	Germana Ellice	8/23/1991	M	gellice14@reverbnation.com	50762 La Follette Way	2307094091	78000	Miscellaneous	8/30/2012
77	78746	Olenka Bagge	8/14/1984	F	obaggej@furl.net	99 8th Parkway	4214670671	40000	Technology	12/30/2012
78	49345	Sandi Son	1/10/1986	F	sson1@simplemachines.org	25137 Melrose Place	3690264634	139000	Technology	4/13/2013
79	80455	Bernardo Tosh	9/1/1994	F	btosh21@rstechnica.com	827 Shopko Circle	9212211518	67000	Engineers	4/23/2013
80	70875	Sarah Mesant	1/17/1997	F	smesant24@cpanel.net	756 Ludington Terrace	3117347569	120000	Capital Goods	7/1/2013
81	49093	ArLine Baelde	6/9/1986	F	abaelde2i@nbcnews.com	97719 Ryan Circle	7774794301	167000	Basic Industries	10/26/2013
82	61964	Angelique Goodere	8/14/1985	F	agoodere1v@creativecommons.org	1350 Goodland Crossing	9687618566	68000	Consumer Non-Durables	1/14/2014
83	89021	Cate Heino	8/28/1990	M	cheino20@youtube.com	1 Judy Trail	1202418831	78000	Consumer Services	9/29/2015
84	72757	Rey Balmforth	11/9/1986	F	rbalmforth19@marriott.com	629 Laurel Pass	6727621297	110000	Engineers	9/29/2015
85	64795	Kenn Childes	1/14/1992	M	kchildes2p@stumbleupon.com	51121 Kim Alley	3756039579	171000	Consumer Durables	1/29/2016
86	41580	Otha Kee	3/7/1981	F	okee5@google.cn	92 Gulseth Street	2450040439	84000	Public Utilities	2/17/2016
87	87465	Oriana De Meis	10/15/1991	F	ode1p@myspace.com	83768 Kensington Way	9437537224	111000	Health Care	3/7/2016
88	91836	Kale Brahams	7/8/1995	F	kbrahams1j@hhs.gov	5152 Merrick Point	6687383841	184000	Technology	4/5/2016
89	71840	Christa Clingoe	6/18/1999	M	cclingoelh@psu.edu	7 Schmedeman Junction	4165640735	160000	Engineers	4/8/2016
90	77166	Wendell Ibanez	3/31/1989	F	wibanez2o@diigo.com	122 Monterey Avenue	2609680554	74000	Consumer Services	6/21/2016
91	90179	Reinhold Delue	10/13/1987	F	rde1ue23@unesco.org	62 Killdeer Avenue	7743045935	59000	Consumer Services	9/19/2016
92	84705	Willette Jayume	5/8/1986	M	wjayume28@edublogs.org	1398 Oriole Place	8182195284	145000	Engineers	10/12/2016
93	73217	Reider Tockell	1/7/1993	M	rtockell11@prweb.com	8036 Artisan Road	5224422965	153000	Energy	12/9/2016
94	87823	Coraline Devanney	9/23/1998	M	cdevanney2l@prweb.com	793 Killdeer Point	4170999045	87000	Basic Industries	5/5/2017
95	93357	Micky Brosi	9/5/1983	M	mbrosi@ca.gov	41193 David Way	2432786668	85000	Finance	9/15/2017
96	83720	Cornell Sammars	11/10/1981	F	csammars7@rambler.ru	42695 Lotheville Circle	7803204392	95000	Engineers	1/23/2018
97	73202	Wash Setterfield	9/16/1991	M	wsetterfield10@indiegogo.com	7 Monument Trail	6745081059	51000	Engineers	4/29/2018
98	35164	Nettle Queyos	8/7/1997	F	nqueyos@desdev.cn	26018 Melody Road	4725150134	148000	Health Care	1/9/2019
99	38993	Nedda Plackstone	6/29/1989	M	nplackstoneo@hostgator.com	635 Eastwood Terrace	4037512459	183000	Technology	4/30/2019
100	38572	Dulciana Willford	6/2/1992	M	dwillfordc@theforest.net	650 Kipling Alley	6682617598	111000	Basic Industries	5/29/2019

Would you like to update the file with the sorted data?

: nope

Viewing reports of company:

1. Salary report:

```
: 6
```

Which report should I present to you?

Type 1 for getting a report on company's expenditure on Salary

Type 2 for getting a report on company's work division

Type 0 to go back to previous menu

```
: 1
```

SALARY REPORT

Total Paid Salary: 11763000

<Sal> (<Name>: <emp_id>)

Average Salary Paid: 117630

Highest Salary: 200000 (Kenn Childes :64795)

Lowest Salary: 35000(Filmore Cockarill:61868)

2. Work Division report:

WORK DIVISION REPORT							

Total Employee Count: 100							
BASIC INDUSTRIES	CAPITAL GOODS	CONSUMER DURABLES	CONSUMER NON-DURABLES	CONSUMER SERVICES	ENERGY	ENGINEERS	FINANCE
Arline Baelde	Corilla MacDonough	Adelaide Steljes	Angelique Goodere	Alistair Low	Bo Lazonby	Andie Meader	Addia Smyth
Coraline Devanney	Goldie Rockingham	Kenn Childes	Gherardo Switzer	Barbra Milroy	Courtenay Larsen	Bernardo Tosh	Aguste Keam
Dulciana Willford	Konrad Americi	Kurtis Carik	Talia Bartoszewicz	Cate Heino	Damiano Aldham	Cherie Lightbourne	Alphonso Broxholme
Kandace Aylén	Lonard Tointon		Torrance Macleague	Cordula O'Cannovane	Reider Tockell	Christa Clingoe	Boycey Thorwarth
Seana Rameau	Miller Corsor			Cyrillus Pettii		Cornell Sammars	Chuck Stollwerck
Sylvester Neissen	Octavia Drake	Sarah Mesant		Gaspar Hickinbottom		Emalee Passingham	Fianna Giannassi
		Willette Clementi		Jesse Piercy		Filmore Cockarill	Gonzales Pinching
				Pammie Ibanez		Gussie Page	Jenni Padkin
				Reinhold Delue		Josy Jackling	Kyle Quarton
				Russ Samper		Lazar Yegorkov	Mackenzie Rawlinson
				Wendell Ibanez		Lise Newcom	Micky Brosi
						Marietta O'Connolly	Nadean Braunds
						Mendel Wainwright	Samaria Dunster
						Merna Schlagh	Sari Bloxsom
						Nial Goodlet	Sterling Abys
						Rey Balmforth	Tymon Burras
						Rora Battson	Ursola Pedlingham
						Rosaline Ackwood	
						Rutter O'Day	
						Sayer Skinn	
						Tucky Lombardo	
						Vyky Rolin	
						Wash Setterfield	

FINANCE	HEALTH CARE	MISCELLANEOUS	PUBLIC UTILITIES	TECHNOLOGY
Addia Smyth	Byrle Baack	Germana Ellice	Otha Kee	Fielding Belden
Aguste Keam	Carley Heineking		Robbi Kasbye	Jillana Ubsdell
Alphonso Broxholme	Cary Upchurch			Jori Jertz
Boycey Thorwarth	Chucho Benning			Kale Brahams
Chuck Stollwerck	Jeana Alliband			Kennie Masterson
Fianna Giannassi	Jojo Wickman			Nedda Plackstone
Gonzales Pinching	Kelwin Angelini			Olenka Bagge
Jenni Padkin	Melisandra Lilburne			Sandi Son
Kyle Quarton	Nettle Queyos			Shaine Pilsworth
Mackenzie Rawlinson	Nickolas Tabard			
Micky Brosi	Oriana De Meis			
Nadean Braunds				
Samaria Dunster				
Sari Bloxsom				
Sterling Abys				
Tymon Burras				
Ursola Pedlingham				

Displaying the entire file at once:

Type 7 for Displaying the entire file at once.

Type 0 for Exiting the program

S no.	Emp_ID	Name	DOB	Sex	Email	Address	Phone No.	Pay(in rs)	Designation	Date of joining
1	65175	Addia Smyth	10/28/1985	F	asmynth@hubpages.com	947 Talisman Terrace	4131740184	87000	Finance	10/10/2005
2	41859	Adelaide Steljes	5/29/2000	M	asteljes29@soundcloud.com	76856 Namekagon Alley	1302480774	176000	Consumer Durables	10/31/2005
3	62046	Aguste Keam	11/6/1986	M	akeam1d@xing.com	78 Sherman Junction	0480161763	180000	Finance	5/19/2009
4	84905	Alistair Low	9/16/1989	M	alow1z@alibaba.com	4 Morning Avenue	0974406384	113000	Consumer Services	1/12/2007
5	46811	Alphonso Broxholme	1/19/1984	F	abroxholmex@hud.gov	6692 Erie Trail	6934869583	160000	Finance	8/15/2001
6	79780	Andie Meader	11/18/1996	M	ameader2e@hp.com	47782 Katie Way	3833425482	153000	Engineers	2/12/2011
7	61964	Angeline Goodere	8/14/1985	F	agoedere1v@creativecommons.org	1350 Goodland Crossing	9687618566	68000	Consumer Non-Durables	1/14/2014
8	49893	Arline Baelde	6/9/1986	F	abaeld2i@cbcnews.com	97719 Ryan Circle	7774794301	167000	Basic Industries	10/26/2013
9	65852	Barbra Milroy	10/26/1989	M	bmilroyq@issuu.com	49375 Menomonie Pass	2510480324	154000	Consumer Services	1/12/2003
10	80455	Bernardo Tosh	9/1/1994	F	btosh21@arstechnica.com	827 Shopko Circle	9212211518	67000	Engineers	4/23/2013
11	68669	Bo Lazonby	4/7/1994	M	blazonby2r@auda.org.au	8216 Hayes Hill	6977892552	102000	Energy	4/5/2010
12	71441	Boycey Thorwarth	8/29/2000	F	bthorwarth1m@lulu.com	85 Graedel Road	9790634870	84000	Finance	12/7/2004
13	93235	Byrle Baack	9/5/1996	M	bbaack1@reverbnation.com	1585 Sachtjen Alley	9472580343	167000	Health Care	4/11/2003
14	36813	Carley Heineking	5/26/1990	M	cheineking2b@samsung.com	3 Golden Leaf Place	3427671550	100000	Health Care	8/10/2007
15	71972	Cary Upchurch	10/25/1984	F	cupchurch2n@themeforest.net	3761 Drewry Park	1619602490	56000	Health Care	6/9/2009
16	89821	Cate Heino	8/28/1990	M	cheino20@youtube.com	1 Judy Trail	1202418031	78000	Consumer Services	9/29/2015
17	81598	Cherie Lighthourne	3/6/2000	F	cilightbournee@bigcartel.com	54 Vermont Plaza	8662140447	159000	Engineers	2/18/2005
18	71840	Christa Clingoe	6/18/1999	M	cclingoe1h@psu.edu	7 Schmedeman Junction	4165640735	160000	Engineers	4/8/2016
19	53518	Chuco Benning	3/26/1984	F	cbenning1k@ibm.com	67505 Schurz Circle	9872713812	70000	Health Care	9/24/2001
20	67020	Chuck Stollwerck	1/27/2000	F	cstollwerck2h@harvard.edu	80468 Park Meadow Drive	7457160396	68000	Finance	3/25/2009
21	87823	Coraline Devanney	9/23/1998	M	cdevanney2l@prweb.com	793 Killdeer Point	4170999045	87000	Basic Industries	5/5/2017
22	41158	Cordula O'Cannovane	8/27/1990	M	cocannovane9@github.com	87885 Birchwood Lane	1061499626	108000	Consumer Services	2/5/2004
23	43691	Corilla MacDonough	12/1/1982	M	cmacdonough0@cam.ac.uk	5686 Nobel Circle	0352520361	39000	Capital Goods	3/1/2012
24	83720	Cornell Sammars	11/10/1981	F	csammars7@rambler.ru	42695 Lotheville Circle	7803204392	95000	Engineers	1/23/2018
25	82773	Courtenay Larsen	8/9/1985	F	clarsen17@webnode.com	9356 Haas Street	8795651896	113000	Energy	12/12/2007
26	76484	Cyrillus Petti	1/7/1996	M	cpettii2@telegraph.co.uk	591 Vidon Point	7470276009	178000	Consumer Services	9/6/2006
27	74760	Damiano Aldham	12/13/1999	M	daldham1f@netscape.com	3134 6th Hill	4855357235	123000	Energy	4/13/2003

71	87465	Oriana De Meis	10/15/1991	F	odelp@myspace.com	83768 Kensington Way	9437537224	111000	Health Care	3/7/2016
72	41580	Otha Kee	3/7/1981	F	okee5@google.cn	92 Gulseth Street	2450040439	84000	Public Utilities	2/17/2016
73	99496	Pammie Ibanez	11/22/1999	F	pibanezm@weebly.com	3064 Cottonwood Alley	3766021648	49000	Consumer Services	9/5/2002
74	73217	Reider Tockell	1/7/1993	M	rtockell11@prweb.com	8036 Artisan Road	5224422965	153000	Energy	12/9/2016
75	90179	Reinhold Delue	10/13/1987	F	rdelue23@unesco.org	62 Killdeer Avenue	7743045935	59000	Consumer Services	9/19/2016
76	72757	Rey Balmforth	11/9/1986	F	rbalmforth19@marriott.com	629 Laurel Pass	0727621297	110000	Engineers	9/29/2015
77	88260	Robbi Kasbye	8/7/1998	F	rkasbye13@quidoo.com	36022 Sunbrook Crossing	8890521759	181000	Public Utilities	9/7/2002
78	49782	Rora Battson	5/27/1982	M	rbattson8@psu.edu	5517 Grover Crossing	1365219631	139000	Engineers	2/15/2009
79	81452	Rosaline Ackwood	6/26/2000	F	rackwood@jalbum.net	83261 Lawn Place	6115150027	115000	Engineers	4/23/2005
80	60745	Russ Samper	4/3/1994	M	rsampera@instagram.com	805 Union Lane	1104384205	165000	Consumer Services	9/16/2004
81	47511	Rutter O'Day	4/13/1984	F	rodayr@senate.gov	4 Armistice Way	7113911897	168000	Engineers	6/4/2012
82	37457	Samaria Dunster	12/12/1996	F	sdunster2q@eventbrite.com	98 Green Ridge Lane	6500173201	102000	Finance	2/22/2012
83	49345	Sandi Son	1/10/1986	F	sson1@simplemachines.org	25137 Melrose Place	3690264634	139000	Technology	4/13/2013
84	70875	Sarah Mesant	1/17/1997	F	smesant24@cpanel.net	756 Ludington Terrace	3117347569	120000	Capital Goods	7/1/2013
85	44208	Sari Bloxsom	3/13/1990	F	sbloxsome@csmonitor.com	95 1st Park	6025229457	45000	Finance	10/22/2001
86	80982	Sayer Skinn	7/7/1983	M	sskin2m@newyorker.com	0372 Division Lane	0914664980	102000	Engineers	1/28/2008
87	36839	Seana Rameau	1/31/1984	F	srameau6@multiply.com	035 Forest Junction	0642458065	171000	Basic Industries	2/23/2003
88	78042	Shaine Pilsworth	3/24/1982	O	spilsworthf@tripadvisor.com	865 Warner Avenue	3581395258	147000	Technology	4/28/2010
89	68723	Sterling Abys	4/28/1988	F	sabys1r@amazonaws.com	13 Spohn Point	6067648288	37000	Finance	12/29/2004
90	83697	Sylvester Neissen	1/4/1990	M	sneissen1@nymag.com	714 Randy Lane	1227401957	182000	Basic Industries	1/18/2007
91	60286	Talia Bartoszewicz	11/21/1995	M	tbaratoszewicz15@businesswire.com	7 Spohn Place	0762283750	38000	Consumer Non-Durables	7/4/2002
92	51949	Torrance MacTeague	10/29/1998	F	tmactague27@fema.gov	00 Tennyson Terrace	8847498635	70000	Consumer Non-Durables	6/19/2008
93	91730	Tucky Lombardo	12/29/1987	M	tlombardo1n@plala.or.jp	88115 Bluestem Plaza	3831318948	185000	Engineers	4/8/2008
94	46781	Tymon Burras	12/17/1991	M	tburras1u@wiley.com	982 Warbler Drive	1543451993	91000	Finance	12/7/2011
95	56302	Ursola Pedlingham	4/4/1994	F	upedlingham@oocql	6505 Gulseth Hill	2849059374	63000	Finance	9/16/2010
96	48668	Vyky Rolin	10/20/1984	M	vrolin2j@cornell.edu	27 Village Road	0719110041	195000	Engineers	8/14/2002
97	73202	Wash Setterfield	9/16/1991	M	wsetterfield10@indiegogo.com	7 Monument Trail	6745081059	51000	Engineers	4/29/2018
98	77166	Wendell Ibanez	3/31/1989	F	wibanez20@diigo.com	122 Monterey Avenue	2609680554	74000	Consumer Services	6/21/2016
99	81683	Willette Clementi	10/8/1982	F	wclementi16@lexa.com	60 Hermina Lane	6460742170	164000	Capital Goods	9/22/2005
100	84705	Willette Jayume	5/8/1986	M	wjayume20@edublogs.org	1398 Oriole Place	8182195284	145000	Engineers	10/12/2016

Advantages:

- The system is immune to any sort of errors, in fact in our testing, we haven't encountered a single error.
- Our system provides with lots of auto-correcting features like-
 - All the things except integers are ignored in all int inputs to avoid breaking the system.
 - Date inputs are checked for the correct format, delimiter, and their validity.
 - Other inputs like employee ID, phone number etc are all checked to avoid non-sensible inputs.
- Smart searching features which automatically detect what the user is entering from Serial No. of records, Employee ID to Names and display the appropriate records accordingly
- Smart name search features allow employee to enter, just the first name, last name or full name and have the system automatically detect the type and show records accordingly.
- Employee sex, basic choice inputs, etc all support various synonyms for the same word to increase user's comfort.
- Does not use data systems like MySQL; instead, opts for more common and suitable and user-friendly data types like .csv which are used commonly by popular industries.
- Generates smart Division and Salary reports in properly formatted and organized manner.
- The code written is very versatile and can be used to display and format other files as well that may not share the same column names or degree. It can be easily optimized by changing the initial variables.

Limitations:

- The system is still in its initial stages.
- Sometimes, random insertions of 0 may break the system.
However, if the user follows the instructions, it is unlikely.
- Lacks GUI.
- Has limited scope.

Bibliography:

- <https://discord.gg/python>
- <https://www.geeksforgeeks.org/python-programming-language/>
- <https://www.w3schools.com/python/>