

**Name:Sakshi Shinde**

### **UNIX File System & Permissions**

**1: Give the execute permission for the user for a file chap1.**

Ans: `chmod u+x chap1`

**2: Give execute permission for user, group and others for a file add.c**

Ans: `chmod a+x add.c`

**3: Remove the execute permission from user, give read permission to group and others for a file aa.c**

Ans: `chmod u-x,g+r,o+r aa.c`

**4: Give execute permission for users for a.c, kk.c, nato and myfile using single command.**

Ans: `chmod u+x a.c kk.c nato myfile`

**5: Change the directory to root directory. Check the system directories, like bin, etc, usr etc.**

1. Ans: Change to the root directory: `cd /`
2. List the contents of the root directory to see the system directories: `ls`

### **Using Pipes and Filters**

**1: Redirect the content of the help document ls, into a file called as lsdoc.**

Ans: `man ls > lsdoc`

**2: Display the content of the lsdoc page wise.**

Ans: `less lsdoc`

**3: Display only the first 4 lines of the lsdoc file.**

Ans: `head -n 4 lsdoc`

**4: Display only the last 7 lines of the file lsdoc.**

**Ans: tail -n 7 lsdoc**

**5: Remove the file lsdoc.**

**Ans: rm lsdoc**

**6: There will be B'day celebration from the friends file, find how many B'day parties will be held. If two of the friends have the B'date on the same day, then we will be having one party on that day.**

**Ans:1. Basic = sort friends | uniq | wc -l**

**2.If specific dates are given= cut -d ' ' -f 2 friends | sort | uniq | wc -l**

**7: Display the lines starting with Ma, in the file friends.**

**Ans: grep '^Ma' friends**

**8: Display the lines starting with Ma, ending with i or ending with id, in the file friends.**

**Ans: grep '^Ma.\*\ (i\ |id\ )\$' friends**

**9: Print all the files and the directory files from the current directory across all the sub directories, along with its path**

**Ans: find . -print**

**10: Print only the Directory files.**

**Ans: find . -type d -print or ls -R | grep ':\$'**

**11: Display the files starting with chap, along with its path.**

**Ans: find . -type f -name 'chap\*' -print**

**12: Sort the file friends in ascending order of names.**

**Ans: sort friends -o friends\_sorted**

**13: Display the contents of the file friends in uppercase letters.**

**Ans: cat friends | tr '[:lower:]' '[:upper:]'**

**14: Store the contents of your home directory in a file called dir.**

**Ans: ls ~ > dir**

**15: From the above file dir, display the file permissions and the name of the file only.**

1. **Ans: List the contents of your home directory with detailed information and store it in dir: ls -l ~ > dir**
2. **Extract the file permissions and file names from the dir file: awk '{print \$1, \$9}' dir**

**16: From the same dir file, store only the file names in a file called files.**

**Ans: cat dir | awk '{print \$9}' | grep -v '^\$' > files**

**17: From the same dir file, store only the permissions of files in a file called perms.**

**Ans: cat dir | awk '{print \$1}' | grep -v '^\$' > perms**

**18: From the same dir file, store only the file sizes in a file called sizes.**

**Ans: cat dir | awk '{print \$5}' | grep -v '^\$' > sizes**

**19: Display the file names, sizes and permissions from your directory in that order.**

**Ans: ls -l | awk '{print \$9, \$5, \$1}'**

**20: Display the number of users working on the system.**

**Ans: who | wc -l**

**21: Find out the smallest file in your directory.**

**Ans:** `ls -lS | grep '^-' | tail -n 1`

**22: Display the total number of lines present in the file friends.**

**Ans:** `wc -l friends`

**23: Create the following fixed record format files (with “|” delimiter between fields) with the structure given below, and populate them with relevant data use these files to solve following questions**

**emp.lst:** Empid(4),Name(18),Designation(9),Dept(10),Date of Birth(8),Salary(5)

**dept.lst:** Dept.Code (2), Name (10), Head of Dept's id(4)

**desig.lst:** Designation Abbr.(2), Name (9)

**Ans:**

**Create emp.lst file:** `echo -e "1001|John Doe |Manager |Sales |19800101|50000\n1002|Jane Smith |Developer|IT |19850215|45000\n1003|Emily Johnson |Analyst |Finance |19900322|40000" > emp.lst`

**Create dept.lst file:** `echo -e "01|Sales |1001\n02|IT |1002\n03|Finance |1003" > dept.lst`

**Create design.lst file:** `echo -e "MG|Manager \nDV|Developer\nAN|Analyst " > desig.lst`

**1. Find the record lengths of each file.**

**Ans:** Find the record length of emp.lst: `head -n 1 emp.lst | wc -c`

Find the record length of dept.lst: `head -n 1 dept.lst | wc -c`

Find the record length of desig.lst: `head -n 1 desig.lst | wc -c`

**2. Display only the date of birth and salary of the last employee record.**

**Ans:** `tail -n 1 emp.lst | awk -F '|' '{print $5, $6}'`

3. Extract only employee names and designations. (Use column specifications). Save output as cfile1.

Ans: `cut -c 6-23,25-33 emp.lst > cfile1`

4. Extract Emp.id, dept, dob and salary. (Use field specifications). Save output as cfile2.

Ans: `cut -d '|' -f 1,4,5,6 emp.lst > cfile2`

5. Fix the files cfile1 and cfile2 laterally, along with the delimiter

Ans: `paste -d '|' ' cfile1 cfile2 > combined_file`

6. Sort the emp.lst file in reverse order of Emp. Names.

Ans: `sort -t '|' -k 2,2r emp.lst -o emp.lst`

7. Sort the emp.lst file on the salary field, and store the result in file srtf.

Ans: `sort -t '|' -k 6,6n emp.lst -o srtf`

8. Sort the emp.lst file on designation followed by name.

Ans: `sort -t '|' -k 3,3 -k 2,2 emp.lst -o emp_sorted.lst`

9. Sort the emp.lst file on the year of birth.

Ans: `sort -t '|' -k 5.1,5.4 emp.lst -o emp_sorted_by_dob.lst`

10. Find out the various designations in the employee file. Eliminate duplicate listing of designations.

Ans: `cut -d '|' -f 3 emp.lst | sort | uniq`

11. Find the non-repeated designation in the employee file.

Ans: `cut -d '|' -f 3 emp.lst | sort | uniq -u`

**12. Find the number of employees with various designations in the employee file.**

**Ans:** `cut -d '|' -f 3 emp.lst | sort | uniq -c`

**13. Create a listing of the years in which employees were born in, along with number of employees born in that year.**

**Ans:** `cut -d '|' -f 5 emp.lst | cut -c 1-4 | sort | uniq -c`

**14. Use nl command to create a code table for designations to include designation code (Start with dept. code 100, and subsequently 105, 110 ...).**

**Ans:** 1. Extract unique designations and save to a temporary file: `cut -d '|' -f 3 emp.lst | sort | uniq > designations.tmp`

2. Use the nl command to create the code table: `nl -v 100 -i 5 designations.tmp > designation_codes.lst`

**24: PCS has its offices at Pune, TTC and Mumbai. The employees' data is stored separately for each office. Create appropriate files (with same record structure as in previous assignment) and populate with relevant data.**

**Ans:**

1. **Create pune\_emp.lst file:** `echo -e "1001|John Doe |Manager |Sales  
|19800101|50000\n1002|Jane Smith |Developer|IT |19850215|45000\n1003|Emily Johnson  
|Analyst |Finance |19900322|40000" > pune_emp.lst`

2. **Create ttc\_emp.lst file:** `echo -e "2001|Alice Brown |Manager |HR  
|19791230|55000\n2002|Bob White |Developer|IT |19860410|47000\n2003|Charlie Black  
|Analyst |Finance |19910505|42000" > ttc_emp.lst`

3. **Create mumbai\_emp.lst file:** `echo -e "3001|David Green |Manager |Marketing  
|19820315|53000\n3002|Eve Blue |Developer|IT |19870725|46000\n3003|Frank Yellow  
|Analyst |Finance |19921212|41000" > mumbai_emp.lst`

**Check Files Using cat command.**

**1. List details about an employee 'Manu Sharma' in the Mumbai office.**

**Ans:** `grep 'Manu Sharma' mumbai_emp.lst`

**2. List only the Emp.Id. And Dept. of Manu Sharma.**

**Ans:** `grep 'Manu Sharma' mumbai_emp.lst | awk -F '|' '{print $1, $4}'`

**3. List details of all managers in all offices. (O/P should not contain file names.).**

**Ans:** `grep 'Manager' pune_emp.lst ttc_emp.lst mumbai_emp.lst`

To check file:

`cat pune_emp.lst ttc_emp.lst mumbai_emp.lst | grep 'Manager'`

**4. Find the number of S.E. in each office.**

**Ans:**

1. Count the number of S.E. in Pune office: `grep 'S.E.' pune_emp.lst | wc -l`
2. Count the number of S.E. in TTC office: `grep 'S.E.' ttc_emp.lst | wc -l`
3. Count the number of S.E. in Mumbai office: `grep 'S.E.' mumbai_emp.lst | wc -l`

**5. List only the Line Numbers and Employee names of employees in 'H/W' in Pune file.**

**Ans:** `grep -n 'H/W' pune_emp.lst | awk -F '|' '{print $1, $2}'`

**6. Obtain a listing of all employees other than those in 'HR' in the Mumbai file and save contents in a file 'nonhr'.**

**Ans:** `grep -v 'HR' mumbai_emp.lst > nonhr`

**7. Find the name and designation of the youngest person who is not a manager.**

**Ans:** `grep -v 'Manager' emp.lst | sort -t '|' -k 5,5r | head -n 1 | awk -F '|' '{print $2, $3}'`

**8. Display only the filename(s) in which details of employee by the name 'Seema Sharma' can be found.**

**Ans:** `grep -l 'Seema Sharma' pune_emp.lst ttc_emp.lst mumbai_emp.lst`

9. Locate the lines containing saxena and saksena in the Mumbai office.

Ans: `grep -i -E 'saxena|saksena' mumbai_emp.lst`

10. Find the number of managers who earn between 50000 and 99999 in the Pune office.

Ans: `grep 'Manager' pune_emp.lst | awk -F '|' '$6 >= 50000 && $6 <= 99999' | wc -l`

11. List names of employees whose id is in the range 2000 – 2999: in Pune Office; in all offices.

Ans: 1. For pune office : `grep -E '^200[0-9]|^2999' pune_emp.lst | awk -F '|' '{print $2}'`

2. For All offices: `grep -E '^200[0-9]|^2999' pune_emp.lst ttc_emp.lst mumbai_emp.lst | awk -F '|' '{print $2}'`

12. Locate people having same month of birth as current month in Pune office.

Ans: Get the current month:

`current_month=$(date +%m)`

Find employees with the same month of birth:

`grep "|....${current_month}" pune_emp.lst`

13. List details of all employees other than those of HR and Admin in file F1.

Ans: `grep -v -E 'HR|Admin' emp.lst > F1`

14. Locate for all Dwivedi, Trivedi, Chaturvedi in Pune file.

Ans: `grep -E 'Dwivedi|Trivedi|Chaturvedi' pune_emp.lst`

15. Obtain a list of people in HR, Admin and Recr. depts. sorted in reverse order of the dept.

Ans: `grep -E 'HR|Admin|Recr' emp.lst | sort -t '|' -k 4,4r`



**25: Write a command sequence that prints out date information in this order: time, day of week, day number, month, year :**

**13:44:42 IST Sun 16 Sept 1994**

**Ans: date +"%T %Z %a %d %b %Y"**

**26: Write a command sequence that prints the names of the files in the current directory in the descending order of number of links.**

**Ans: ls -l | sort -k2,2nr | awk '{print \$9}'**

**27: Write a command sequence that prints only names of files in current working directory in alphabetical order.**

**Ans: ls -1p | grep -v / | sort**

**28: Write a command sequence to print names and sizes of all the files in current working directory in order of size.**

**Ans: ls -lS | awk '{print \$5, \$9}'**

**29: Determine the latest file updated by the user.**

**Ans: ls -lt | head -n 2**