

# IGNOU MCA MCS-45 Solved Assignment 2011

Course Code	:	MCSL-045
Course Title	:	UNIX and DBMS Lab
Assignment Number	:	MCA(4)/045/Assign/2010-11
Maximum Marks	:	100
Weightage	:	25%
Last Date of Submission	:	31 <sup>st</sup> October, 2010 (for July, 2010 session) 30 <sup>th</sup> April, 2011 (for January, 2011 session)

The assignment has two parts A and B. Answer all the questions. Each part is for 20 marks. UNIX and DBMS lab record carries 40 Marks. Rest 20 marks are for viva voce. You may use illustrations and diagrams to enhance the explanations. Please go through the guidelines regarding assignments given in the MCA Programme Guide for the format of presentation. If any assumptions made, please state them.

## Question1.

(a) `$ find / -name 'program.c' 2>/dev/Null`  
`$find / -name`

(b) `Who : sort`

(c) `$ quep -c 'IGNOU' Ignou project.txt`

(d) `Cat > Assignment.txt`

This assignment is quite tough & Lengthy  
Ctr+d will save the file assignment

(e) `Chmod +R Myfile.txt`  
`Chmod -R Myfile.txt`  
`Chmod +w Myfile.txt`  
`Chmod -w Myfile.txt`

(f) `$ quep -n 'ab' assignment.txt`

(g) `Cmp file1 file2`

(h) Step 1 : `Cat / etc/ passed`  
Step 2 : `Cat / etc/ passed / qrep "/have"`  
Step 3 : Now we will get all the user account which have their have  
There have share in / home  
Step 4 : Now we will modify an  
`Cat / etc/ passed / qrep "/have" / cut - d : -f`



- (i) Cat > Ignou \_jalad
1. I Initially
  2. Was happy
  3. To take
  4. Admission
  5. In Ignou
  6. Now I
  7. See the
  8. Attitude
  9. Of My
  10. Teacher
  11. At study
  12. Center they
  13. Don't teach
  14. & I
  15. Am really
  16. Looted &
  17. Find cheated
  18. To take
  19. Admission in
  20. IGNOU.
- Ctrl + d
- \$ Split -b 10 Ignou\_jalad would  
Split the file Ignou\_jalad into  
Two pieces 'Ignou\_jalad 1, Ignou\_jalad 2.
- (j) Tr -Delete '=' & ( ) []'  
Delete Specified set of characters  
Defined in set 1 but do not translate

## Question 2

(a)      Filename = "ignou.txt"  
          File = open (filename , 'r')  
          Obj = file.read (filename)  
          Echo "obj.Account No."  
          Echo "obj.Date".  
          Echo "obj. Credit".  
          Echo "obj.Amount".

(b)

```
File = $1
Echo -n "enter a file name:"
Read file
If [! -f $ file]
Then
Echo "$ file not a file!"
Exit!
fi
Echo -n "enter a password:"
Read password.
# do encryphon any UNIX crypt Cammand
# this command will proper for a password
Crypt $ password <$file> $ file.cpy
Echo "$file.cpy created as encrypted file"
```

(c)      a= \$1  
          b=\$2  
          c=\$3  
          if [\$ # -lt 3]  
          then  
          echo "\$0 n1 n2 n3"  
          exit1  
fi  
if [\$ a -gt \$b-a \$a -gt \$c]  
then

```
echo "$ a is largest integer"
elif [ $b -gt $ a -a $b -gt $ c]
then
echo "$ b is largest integer"
elif [ $c -gt $ a -a $ c -gt $ b];
then
echo "$ c is largest interg"
else
echo "sorry cannot given number"
fi
```

**PART-II: MCS-043**

**Question No. (1)**

(a) Create database University

Use University

Create Table student

(Id int primary key,

Name char (25), Not Null,

Programme char (25), Not Null,

Total\_semester in not null

Semester\_completed int not null

Semester\_Registered int not null

)

Creat Table fees

(

Id int primary key,

Semester int Not Null,

Subject char(20) Not Null,

Total\_fee int Not Null,

Fee\_paid int not null,

{\_registered char (20) Not Null

)

Create Table Department

(Id int primary key ,

Name char (20) Not Null,

Programme char (20) Not Null

)

(b) (i)

Select student\_Id, student\_Name,Student\_Programme

From student inner join fees on

Student:Id = fees.Id where Fees\_Register='No'

(ii)

Select count (subject) from Department

Group by Department having  
Count (Subject) < 50

(iii)

Select count (semester) from student  
Group by semester having count (semester) > 4

(iv)

Select count (Semester) from student group by semester having  
Count (semester) =8

(c) Create view data entry

As

Insert into student values (1,'join', 'MCA', 2, 2, 3)

Create view show

As

Select \* from student where Id=@Id

(d) Create Procedure pro()

As

Select sum (fee paid) from fees

Group by Id having

Sum (fee paid) <8000

(e) Begin Transaction

Insert into student values

(2, 'Albert', 'BCA', 3.34)

Commit Transaction

(f) (i)

Create or Replace Trigger "CCAPADM"

TRG\_PRINT\_INDEX

AFTER REGISTRATION

On "CCAPADM"