**Exercise 1 - Extracting data using 'cut' command**

The filter command cut helps us extract selected characters or fields from a line of text.

1. Extracting characters.

The command below shows how to extract the first four characters.

1. 1
2. echo "database" | cut -c1-4

Copied!Executed!

You should get the string ‘data’ as output.

The command below shows how to extract 5th to 8th characters.

1. 1
2. echo "database" | cut -c5-8

Copied!Executed!

You should get the string ‘base’ as output.

Non-contiguous characters can be extracted using the comma.

The command below shows how to extract the 1st and 5th characters.

1. 1
2. echo "database" | cut -c1,5

Copied!Executed!

You get the output : ‘db’

1. Extracting fields/columns

We can extract a specific column/field from a delimited text file, by mentioning

* the delimiter using the -d option, or
* the field number using the -f option.

The /etc/passwd is a “:” delimited file.

The command below extracts usernames (the first field) from /etc/passwd.

1. 1
2. cut -d":" -f1 /etc/passwd

Copied!Executed!

The command below extracts multiple fields 1st, 3rd, and 6th (username, userid, and home directory) from /etc/passwd.

1. 1
2. cut -d":" -f1,3,6 /etc/passwd

Copied!Executed!

The command below extracts a range of fields 3rd to 6th (userid, groupid, user description and home directory) from /etc/passwd.

1. 1
2. cut -d":" -f3-6 /etc/passwd

**Exercise 2 - Transforming data using 'tr'**

tr is a filter command used to translate, squeeze, and/or delete characters.

1. Translate from one character set to another

The command below translates all lower case alphabets to upper case.

1. 1
2. echo "Shell Scripting" | tr "[a-z]" "[A-Z]"

Copied!Executed!

You could also use the pre-defined character sets also for this purpose:

1. 1
2. echo "Shell Scripting" | tr "[:lower:]" "[:upper:]"

Copied!Executed!

The command below translates all upper case alphabets to lower case.

1. 1
2. echo "Shell Scripting" | tr "[A-Z]" "[a-z]"

Copied!Executed!

1. Squeeze repeating occurrences of characters

The -s option replaces a sequence of a repeated characters with a single occurrence of that character.

The command below replaces repeat occurrences of ‘space’ in the output of ps command with one ‘space’.

1. 1
2. ps | tr -s " "

Copied!Executed!

In the above example, the space character within quotes can be replaced with the following : "[\:space\:]".

1. Delete characters

We can delete specified characters using the -d  option.

The command below deletes all digits.

1. 1
2. echo "My login pin is 5634" | tr -d "[:digit:]"

Copied!Executed!

The output will be : ‘My login pin is’

**Exercise 4 - Create a table**

In this exercise we will create a table called users in the PostgreSQL database using PostgresSQL CLI. This table will hold the user account information.

The table users will have the following columns:

1. uname
2. uid
3. home
4. You will connect to template1 database which is already available by default. To connect to this database, run the following command at the ‘postgres=#’ prompt.
5. 1
6. \c template1

Copied!

You will get the following message.

You are now connected to database "template1" as user "postgres".

Also, your prompt will change to ‘template1=#’.

1. Run the following statement at the ‘template1=#’ prompt to create the table.
2. 1
3. create table users(username varchar(50),userid int,homedirectory varchar(100));

Copied!

If the table is created successfully, you will get the message below.

CREATE TABLE