DBTechNet
DBTech VET

Lifelong Learning Programme

Quick Start to the DBTechLab (DebianDB) Database Laboratory

Disclaimers

This project has been funded with support from the European Commission. This publication [communication] reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein. Trademarks of products mentioned are trademarks of the product vendors.

DebianDB is a Debian Linux virtual machine providing a portable and free database laboratory to be used either on classroom courses or as self-study tool of students or individual professionals. It is based on an earlier version developed at University of Macedonia in Thessaloniki for DBTech EXT project of DBTechNet, and further developed by Mr. Taito Halonen at Haaga-Helia UAS in Helsinki for DBTechNet. We are grateful also to Debian.org (http://www.debian.org/index.en.html) on making the Linux platform and updates available and to the database vendors providing the free editions of their commercial products available for database studies, relevant in the working life.

The purpose of this document is to help you in installing the lab and to start using the database product(s) of your choice in this Linux based database laboratory of your own, even if you are not too familiar to Linux before.

Installing the Lab

The free virtual computer platforms which you can use for using the DebianDB laboratory include

- VMware Player available at http://www.vmware.com/products/player/
- Oracle VirtualBox available at https://www.virtualbox.org/

providing the corresponding installing instructions. For VirtualBox we recommend version 4.2 or later, and you need to install also the compatible Oracle VM VirtualBox Extension Pack.

The current version 6 of the DebianDB package is available at http://www.dbtechnet.org/download/DebianDBVM06.zip (4.9 GB) and includes the following files

DebianDBV06_n.ova (n depends on the available version)
ReadMeFirst.doc
ReleaseNotes.doc
installation documents

The OVA file contains the virtual computer image of our database lab, and it can be imported either to VMware or VirtualBox virtual machine platform. As the operating system we have Debian Linux with GNOME desktop environment, which is the same in both virtual platforms. In the following we will focus on VirtualBox platform.

The Virtual Machine Image User's Guide explains details of the setup, how to import the image (virtual appliance) to VirtualBox, how to configure the keyboard layout, which DBMS products have been installed in the Lab, and what other software have been installed there for experimenting. The free DBMS products in DebianDB include DB2 Express-C, Oracle XE, MySQL/InnoDB, PostgreSQL, and Pyrrho.

To import the OVA file into an appliance in VirtualBox you start the VirtualBox and select File -> Import Appliance.. You will get into "Import Virtual Appliance" form where you press the "Open appliance.." button which gets you to "Select an appliance to import" form by which you find fírst the proper OVA file and then press the "Open" button. With the selected OVA file you will return to the "Import Virtual Appliance" form and press there the "Next" button getting to "Appliance settings" where you press the "Import" button, and wait until the DebianDB appliance appears on the list of appliances. Then simply select the appliance and press the "Start" icon on the icon tools to start booting of DebianDB.

Users

The users currently defined in DebianDB and their passwords and Linux groups are explained in the document "Virtual Machine Image Administrators Guide". To keep the lab as an easy tool for database studies the typical password for most users has been changed just to "password".

As default user logs in as "student" user. The password of student is "password" which you occasionally need, for example on waking DebianDB if it has gone to sleep after waiting too long for your typing. User student can access some of the DBMS products in DebianDB, but not all. Therefore a generic user "dba" with administrator rights has been configured.

User login can be switched for example to "dba" by selecting

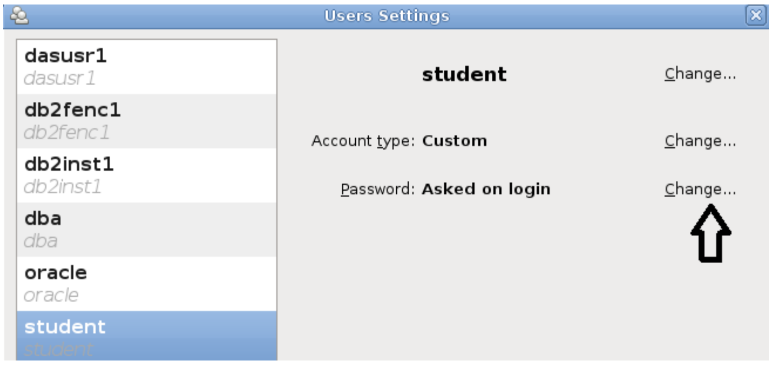
Applications -> System Tools -> New Login

and selecting "dba" from the pop-up menu. User "dba" can use the installed products, but for example DB2 instance has been created by user "db2inst1" (password 'password') and user "dba" needs to be granted connection and access privileges to database objects before the user can experiment with DB2.

Occasionally you need access as root user. The password for root is "P4ssw0rd" – so character "4" instead of "a" and character zero instead of "o".

If you prefer to change passwords, you can change the password of root simply as root user by the Linux command "passwd", and for other users selecting the proper form starting from menu bar as follows

System -> Administration -> Users and Groups -> User Settings

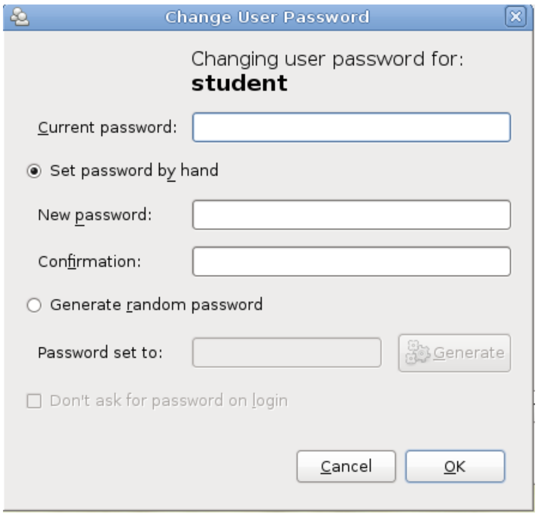


student

Account type: Custom

Password: Asked on login

Select the user and "Change…" for "Password: Asked on login", then filling in the form "Change User Password" the old and the new passwords, and finally pressing the OK button



Changing user password for:
student

Current password:

● Set password by hand

New password:

Confirmation:

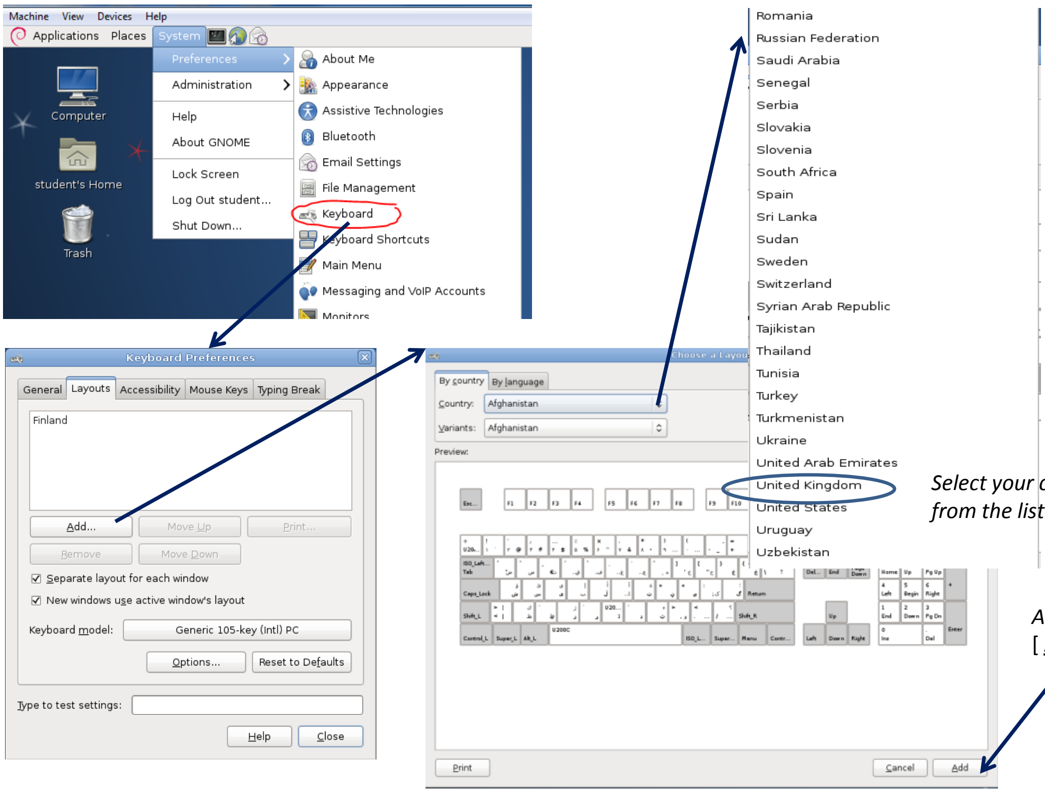
○ Generate random password

Password set to:

□ Don't ask for password on login

Keyboard Layout

The default layout is “Finland”, so you probably want to configure it into your native keyboard using the steps like in the following example of adding “United Kingdom” keyboard layout



☑ Separate layout for each window

☑ New windows use active window's layout

Keyboard model: Generic 105-key (Intl) PC

Type to test settings:

Country: Afghanistan

Variants: Afghanistan

Select your country from the list

And press [Add]

and removing “Finland”.

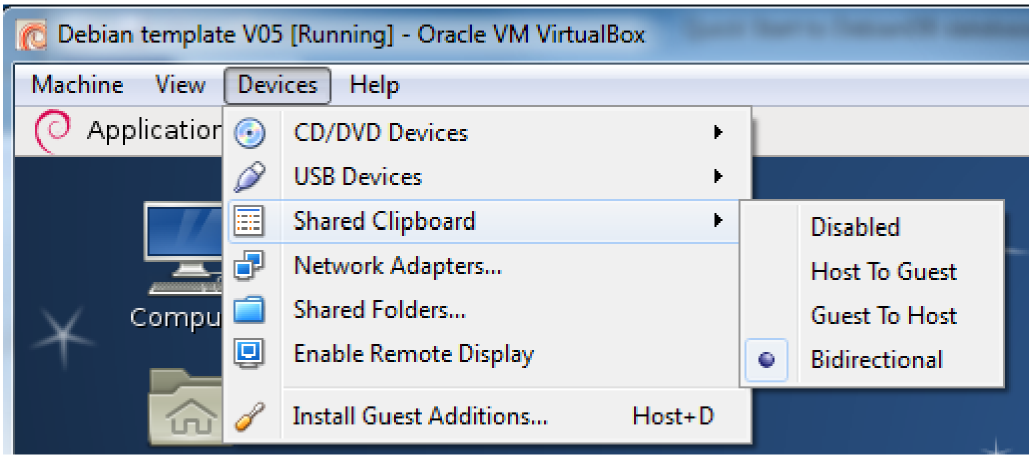
Interchange with the Host Computer

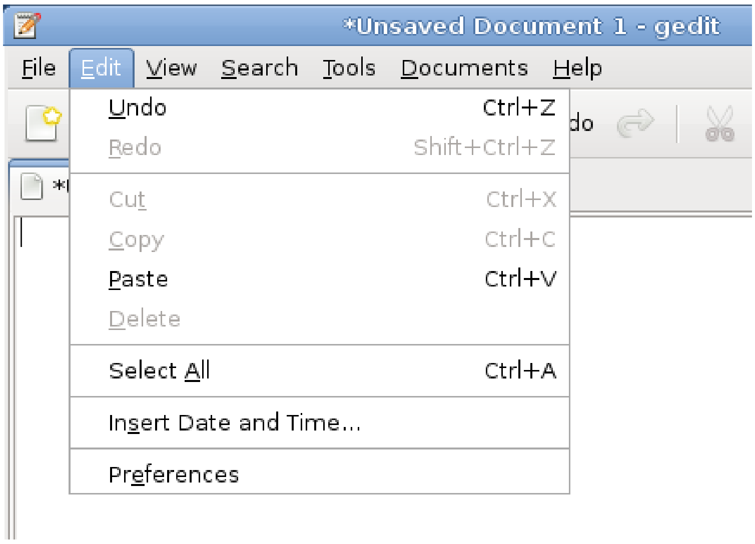
This can be used either by copy/paste of data using Shared Clipboard, or moving files via attached devices such as USB devices or via shared folders.

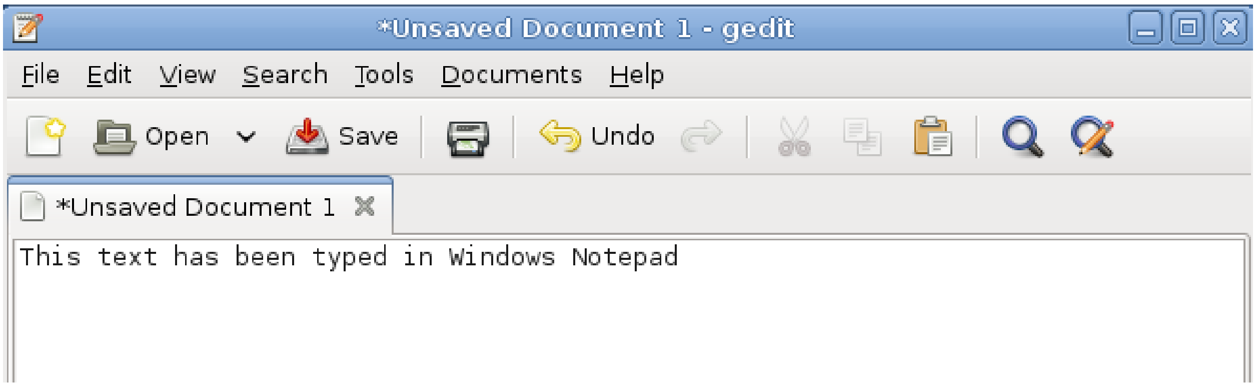
Shared Clipboard

Copy/paste can be configured by selecting

Devices -> Shared Clipboard -> Bidirectional



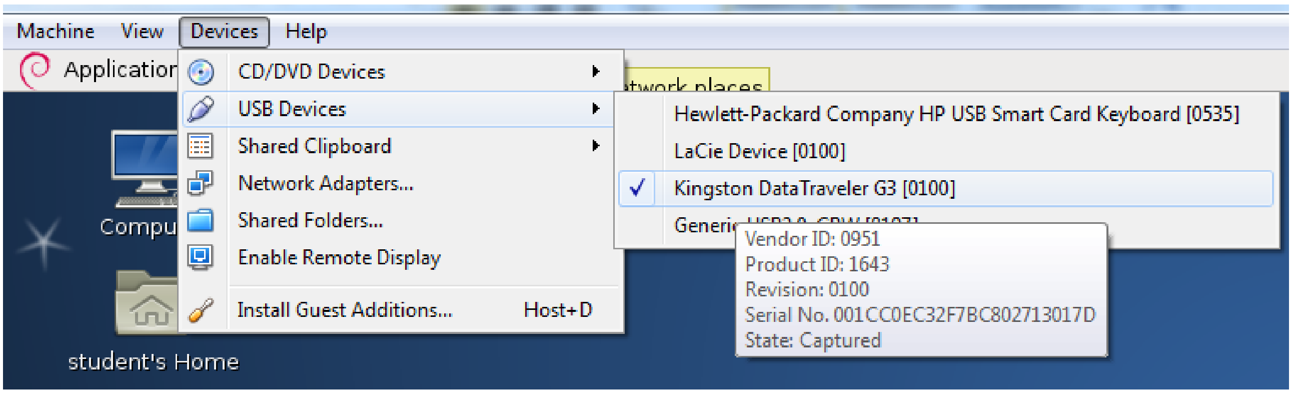



*Unsaved Document 1 - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
*Unsaved Document 1

This text has been typed in Windows Notepad

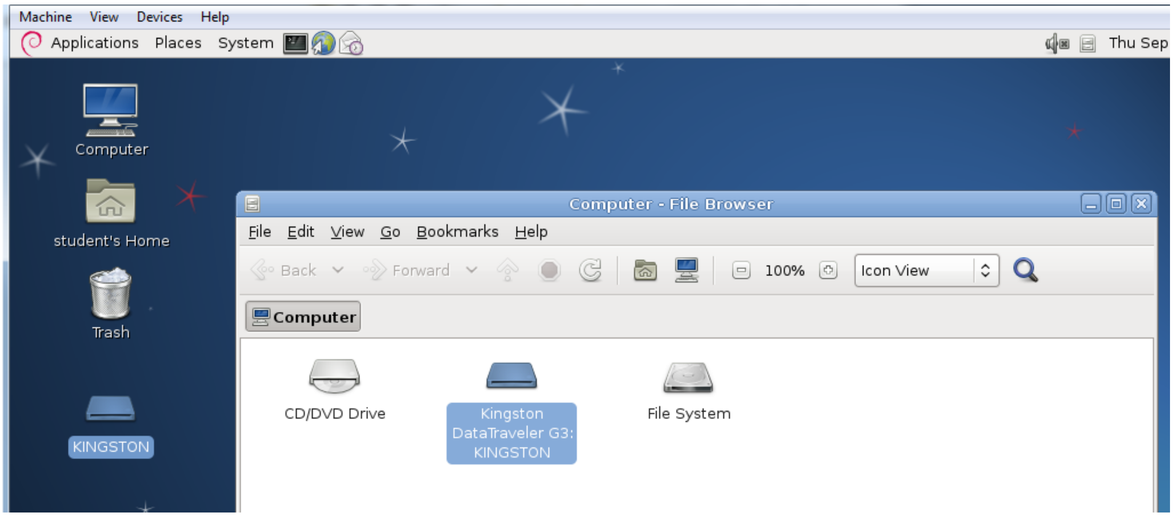
Shared USB Devices

The USB device to be used shall be attached into the host, but must be non-active. In the virtual machine menu bar we select Devices -> USB Devices -> <list of connected devices> from which we select the planned device.

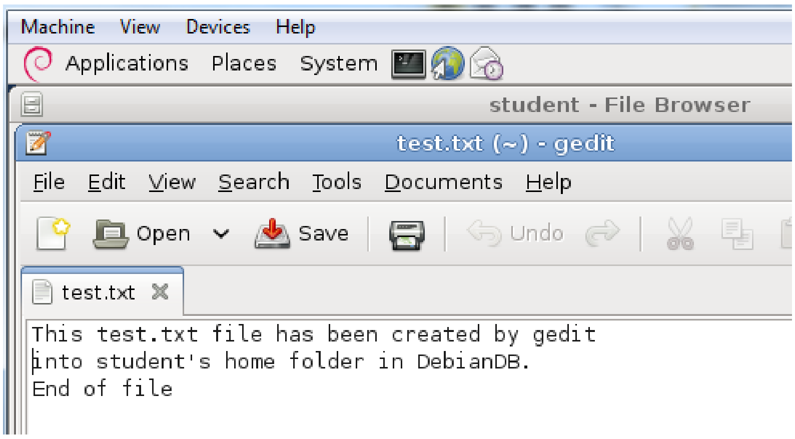


Vendor ID: 0951
Product ID: 1643
Revision: 0100
Serial No. 001CC0EC32F7BC802713017D
State: Captured

After this we need to wait for about half a minute for the device to appear available for the Computer as follows

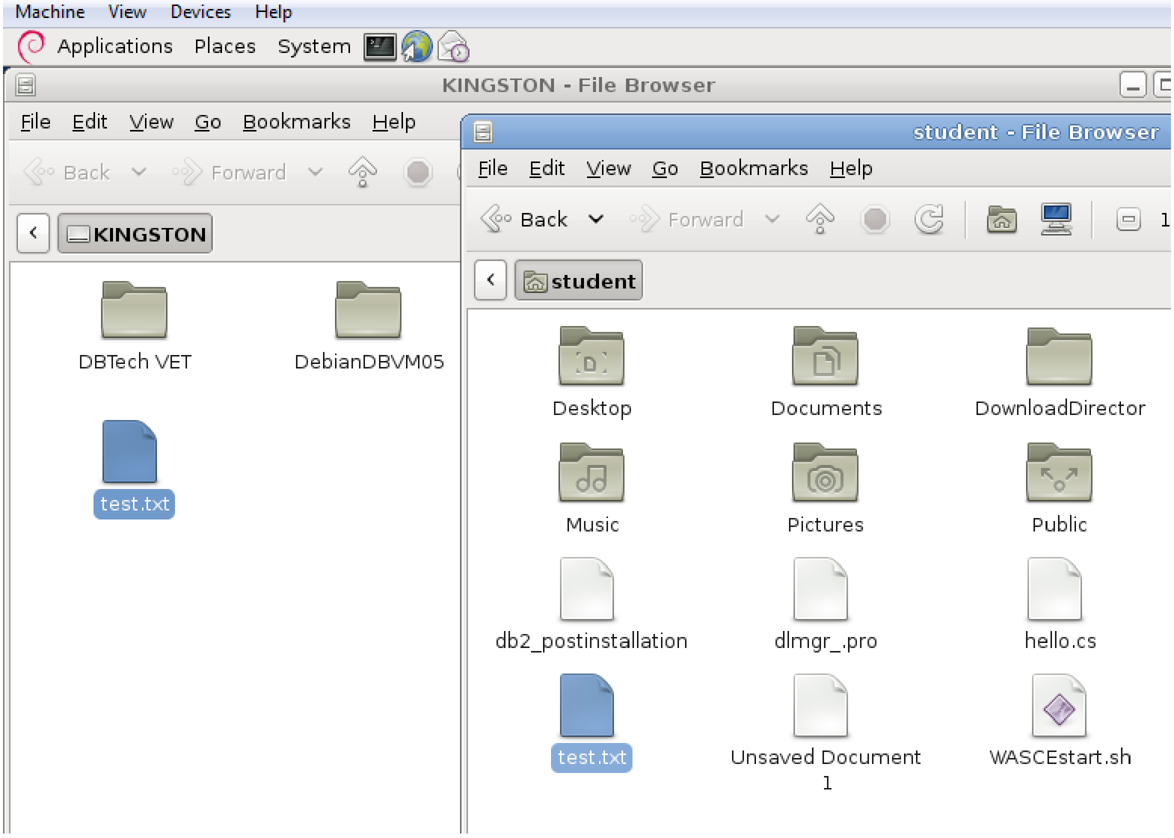


A problem in the file transfer between Windows and Unix/Linux is that Windows uses sequence of CR LF (\r \n) characters as linefeed markers whereas Unix and Linux us just single LF (\n) character. To test file transfer from Linux to Windows in the following, we first create file test.txt with the following contents:

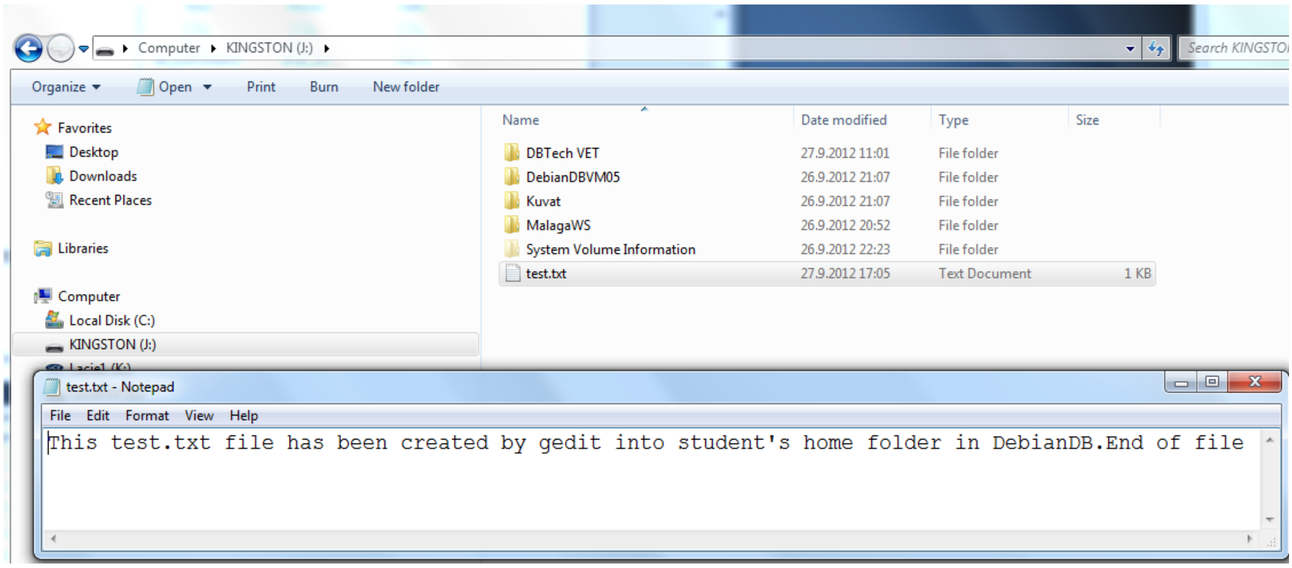

Machine View Devices Help
Applications Places System
student - File Browser
test.txt (~) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
test.txt

```
This test.txt file has been created by gedit
into student's home folder in DebianDB.
End of file
```

After saving the file to local folder in Linux, we drag and drop the file to the USB device



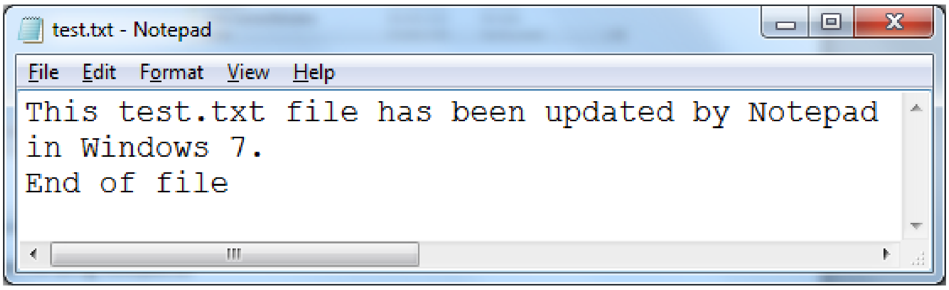
After unselecting the device and re-attaching the device into the host, we can access the test.txt file by the host as follows



Name	Date modified	Type	Size
DBTech VET	27.9.2012 11:01	File folder	
DebianDBVM05	26.9.2012 21:07	File folder	
Kuvat	26.9.2012 21:07	File folder	
MalagaWS	26.9.2012 20:52	File folder	
System Volume Information	26.9.2012 22:23	File folder	
test.txt	27.9.2012 17:05	Text Document	1 KB

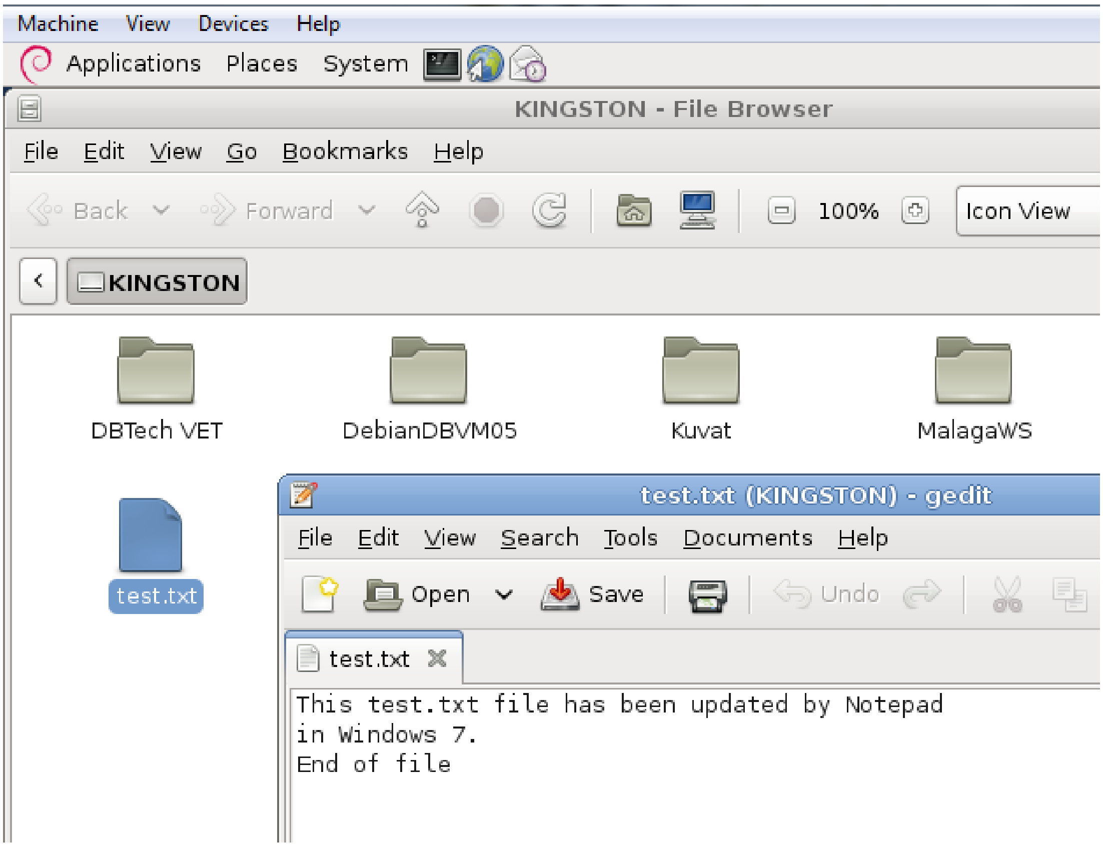
```
This test.txt file has been created by gedit into student's home folder in DebianDB.End of file
```

To test the reverse file transfer from Windows to Linux, we update the file contents using `Notepad` into the following



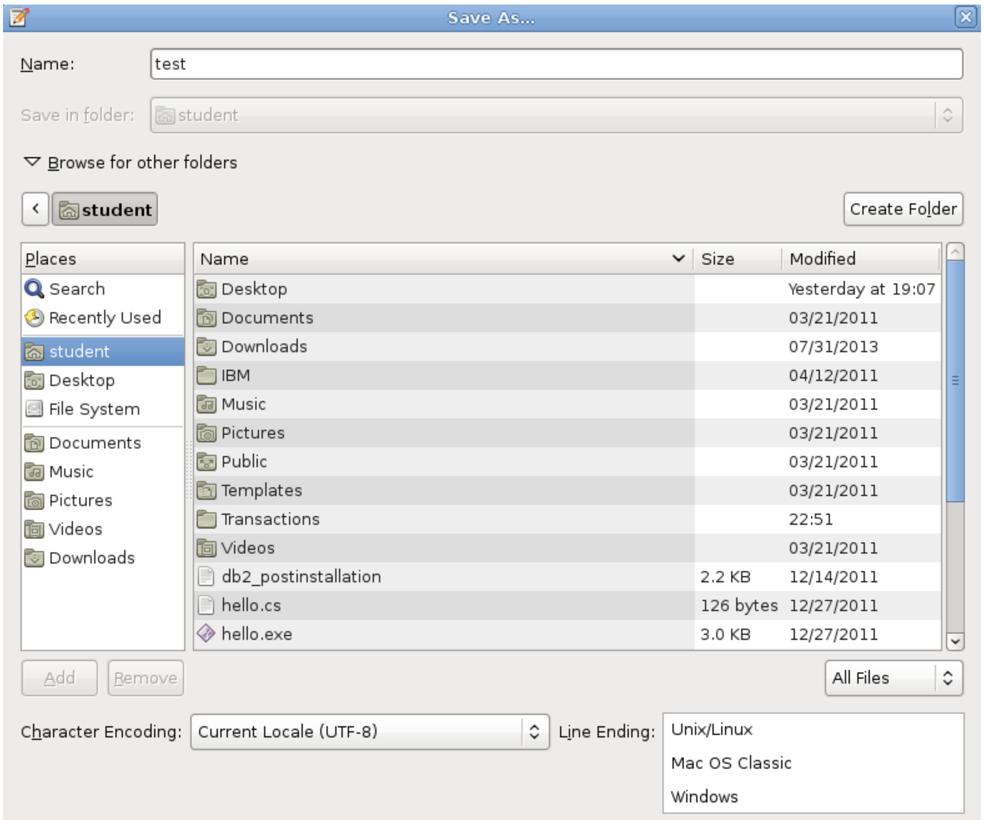
```
This test.txt file has been updated by Notepad
in Windows 7.
End of file
```

and reconnecting USB device to DebianDB, we can see that Linux `gedit` understands the Windows linefeeds


Machine View Devices Help
Applications Places System
KINGSTON - File Browser
File Edit View Go Bookmarks Help
Back Forward 100% Icon View
KINGSTON
DBTech VET
DebianDBVM05
Kuvat
MalagaWS
test.txt
test.txt (KINGSTON) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
test.txt

```
This test.txt file has been updated by Notepad
in Windows 7.
End of file
```

If the text file in gedit program should be copied later to Windows platform the linefeeds can be converted automatically in gedit's 'Save As ..' form selecting 'Windows' for the Line Ending value



Name: test

Save in folder: student

Name	Size	Modified
Desktop		Yesterday at 19:07
Documents		03/21/2011
Downloads		07/31/2013
IBM		04/12/2011
Music		03/21/2011
Pictures		03/21/2011
Public		03/21/2011
Templates		03/21/2011
Transactions		22:51
Videos		03/21/2011
db2_postinstallation	2.2 KB	12/14/2011
hello.cs	126 bytes	12/27/2011
hello.exe	3.0 KB	12/27/2011

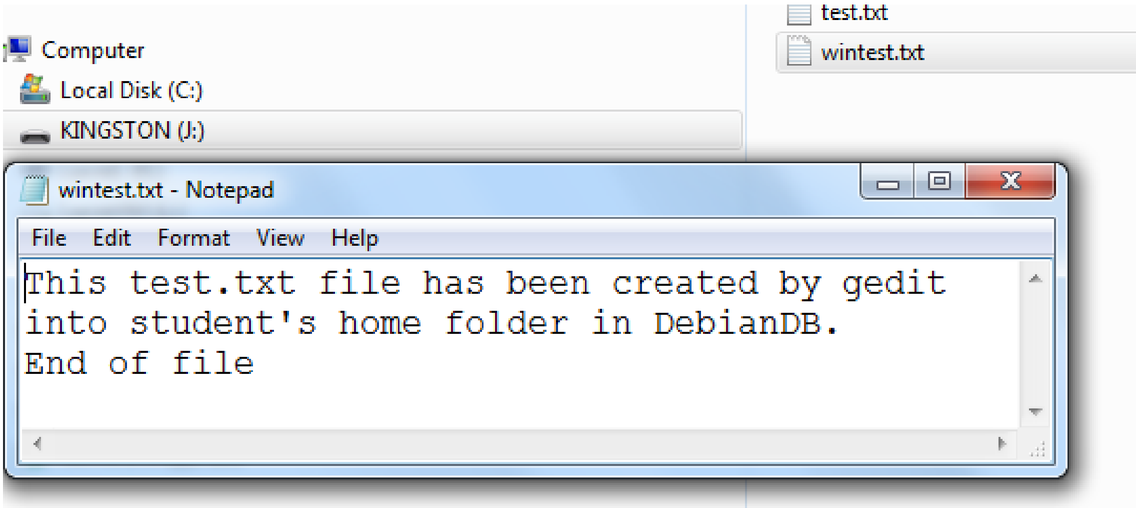
Character Encoding: Current Locale (UTF-8)

Line Ending:
Unix/Linux
Mac OS Classic
Windows

Converting Linux linefeeds to Windows linefeeds can also be done by Linux `awk` utility as follows

```
student@debianDB:~$ awk 'sub("$", "\r")' test.txt > wintest.txt
```

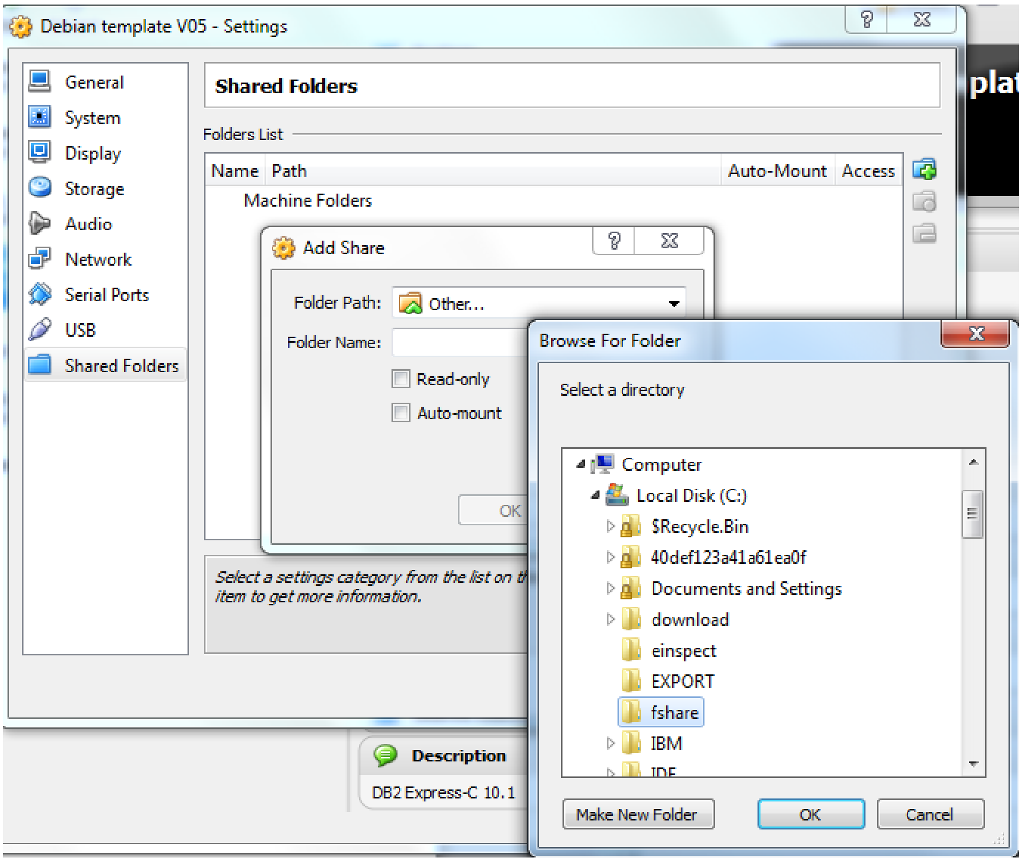
This creates the new file wintest.txt, which can be copied to the attached USB device, and after detaching the device we can see that the converted file is perfectly readable in Windows

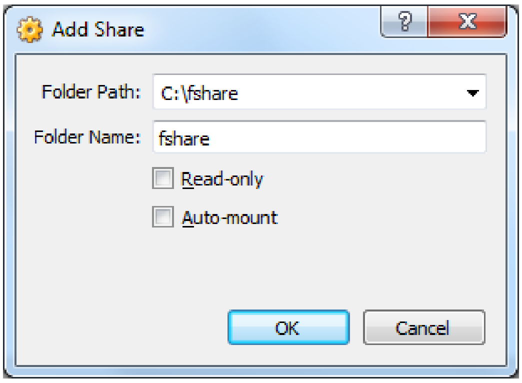


```
This test.txt file has been created by gedit
into student's home folder in DebianDB.
End of file
```

Shared Folders

1) prior to booting your virtual machine, from the virtual box interface you select the target VM and click on "Shared Folders" in the right-hand part of the screen. You direct the system to a certain folder of the host operating system and name it, say, fshare



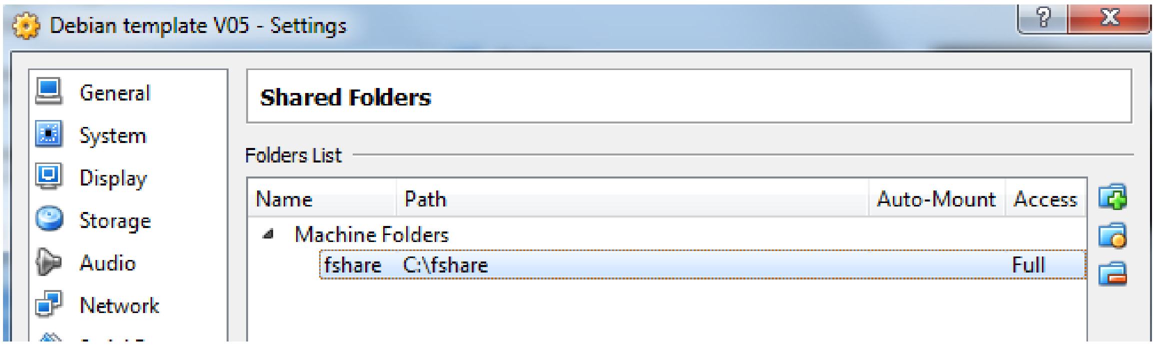


Folder Path: C:\fshare

Folder Name: fshare

□ Read-only

□ Auto-mount

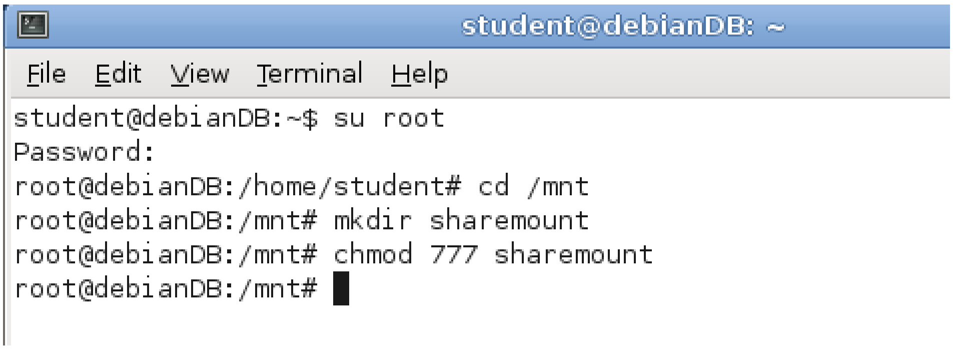


Name	Path	Auto-Mount	Access
Machine Folders			
fshare	C:\fshare		Full

2) you login to the Debian VM as 'student' and from within the student account you 'su root' (in the terminal window) in order to login as 'root'

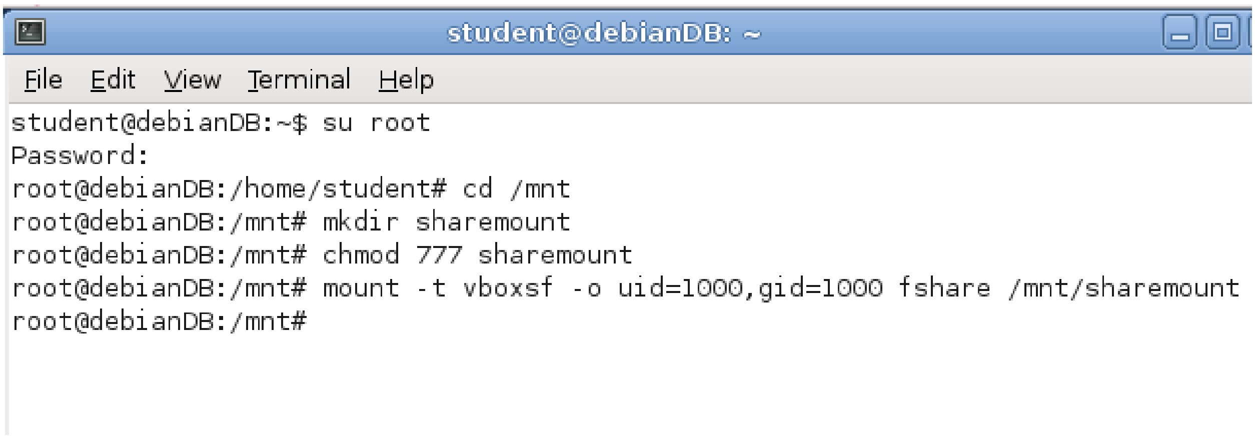
3) you create a new folder under the /mnt directory, say: /mnt/sharemount

4) you issue a 'chmod 777 /mnt/sharemount' command to give full access to all accounts on the 'sharemount' folder



```
student@debianDB:~$ su root
Password:
root@debianDB:/home/student# cd /mnt
root@debianDB:/mnt# mkdir sharemount
root@debianDB:/mnt# chmod 777 sharemount
root@debianDB:/mnt#
```

5) you issue a 'mount -t vboxsf -o uid=1000,gid=1000 fshare /mnt/sharemount' command to mount the 'fshare' (host) volume under the '/mnt/sharemount' folder

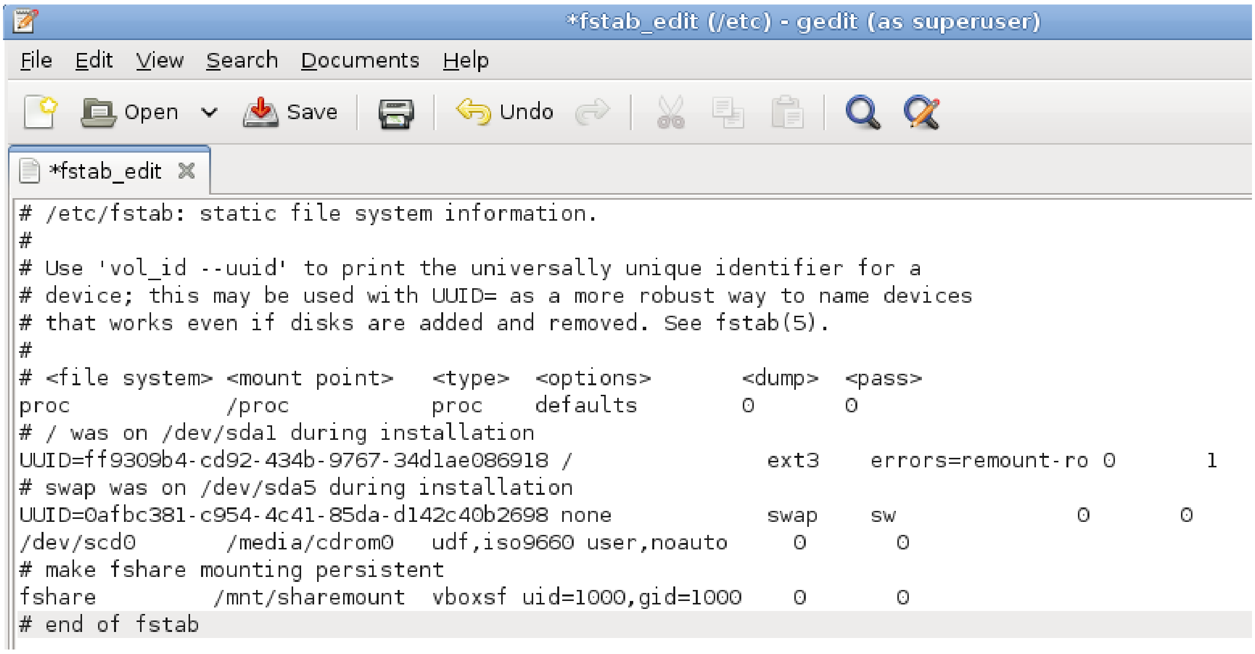


```
student@debianDB:~$ su root
Password:
root@debianDB:/home/student# cd /mnt
root@debianDB:/mnt# mkdir sharemount
root@debianDB:/mnt# chmod 777 sharemount
root@debianDB:/mnt# mount -t vboxsf -o uid=1000,gid=1000 fshare /mnt/sharemount
root@debianDB:/mnt#
```

6) to make the fshare volume mounting persistent (i.e. automatically be made at login time), you include the following line in the 'etc/fstab' file:
'fshare /mnt/sharemount vboxsf uid=1000,gid=1000 0 0'

Note: in the following experiment this was done by root using following commands

```
cp fstab fstab_bak
cp fstab fstab_edit
gedit fstab_edit
```


*fstab_edit (/etc) - gedit (as superuser)
File Edit View Search Documents Help
Open
Save
Undo
*fstab_edit

```
# /etc/fstab: static file system information.
#
# Use 'vol_id --uuid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
proc            /proc           proc    defaults        0       0
# / was on /dev/sda1 during installation
UUID=ff9309b4-cd92-434b-9767-34d1ae086918 /               ext3    errors=remount-ro  0       1
# swap was on /dev/sda5 during installation
UUID=0afbc381-c954-4c41-85da-d142c40b2698 none            swap    sw               0       0
/dev/scd0       /media/cdrom0   udf,iso9660 user,noauto      0       0
# make fshare mounting persistent
fshare         /mnt/sharemount  vboxsf uid=1000,gid=1000     0       0
# end of fstab
```

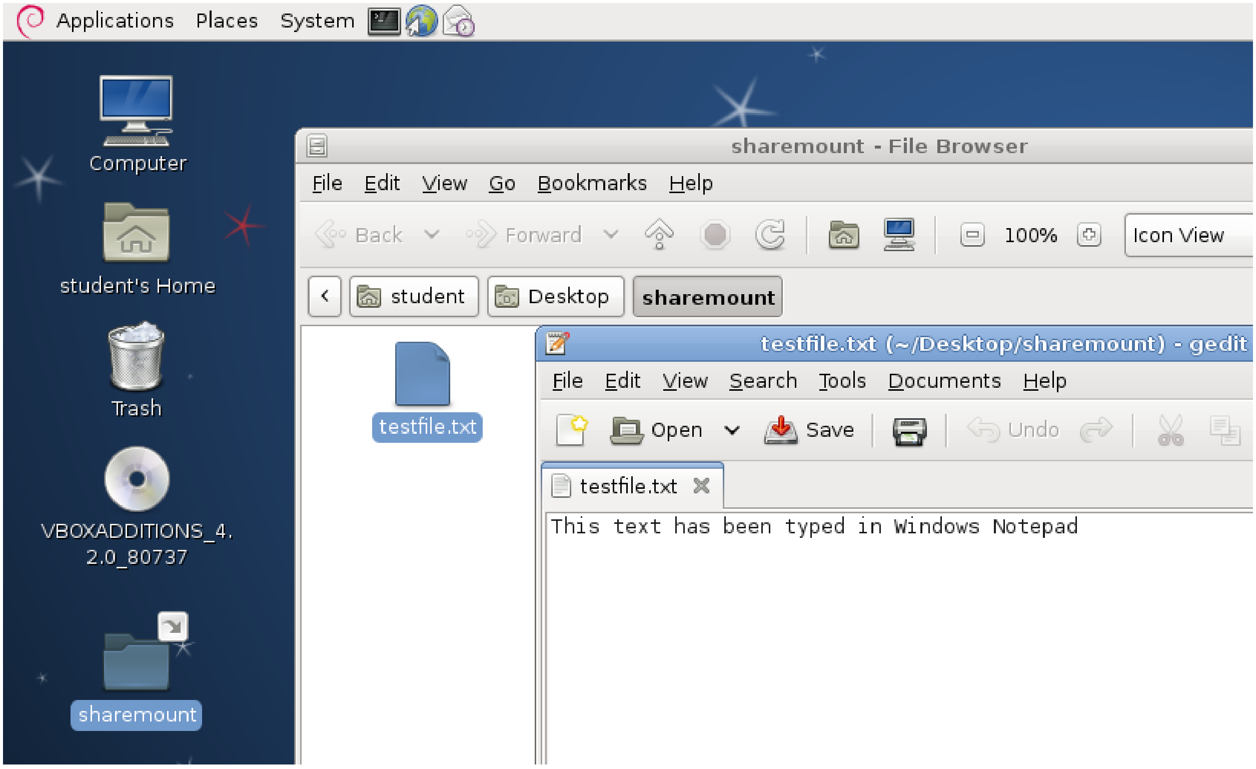
after saving the file it was copied over the original fstab

```
cp fstab_edit fstab
```

7) to have the sharemount folder appear on the desktop of the Debian VM you return to the student account and from within the terminal window you issue the command:

```
ln -s /mnt/sharemount $HOME/Desktop/sharemount
```

Now the sharemount folder appears on student’s desktop permanently and in the following test we see a testfile from the Windows C:\fshare directory


Applications Places System
Computer
student's Home
Trash
VBOXADDITIONS_4.2.0_80737
sharemount
sharemount - File Browser
File Edit View Go Bookmarks Help
Back Forward 100% Icon View
student Desktop sharemount
testfile.txt
testfile.txt (~/Desktop/sharemount) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
testfile.txt

```
This text has been typed in Windows Notepad
```

Accessing the DBMS Products

In the DebianDB virtual lab you can experiment with the free DBMS products: DB2 Express-C 9.7, Oracle 10g XE, MySQL 5, PostgreSQL, and Pyrrho DBMS. Even if most products have also client tools with graphical user interface, in this document we mainly present the traditional character mode tools, which we consider convenient for experimenting with concurrent SQL transactions.

DB2 Express-C 9.7

DB2 instance and the SAMPLE database in DebianDB has been created by user “db2inst1” (password “password”), so for the beginning, the easiest choice is to switch to user “db2inst1” selecting

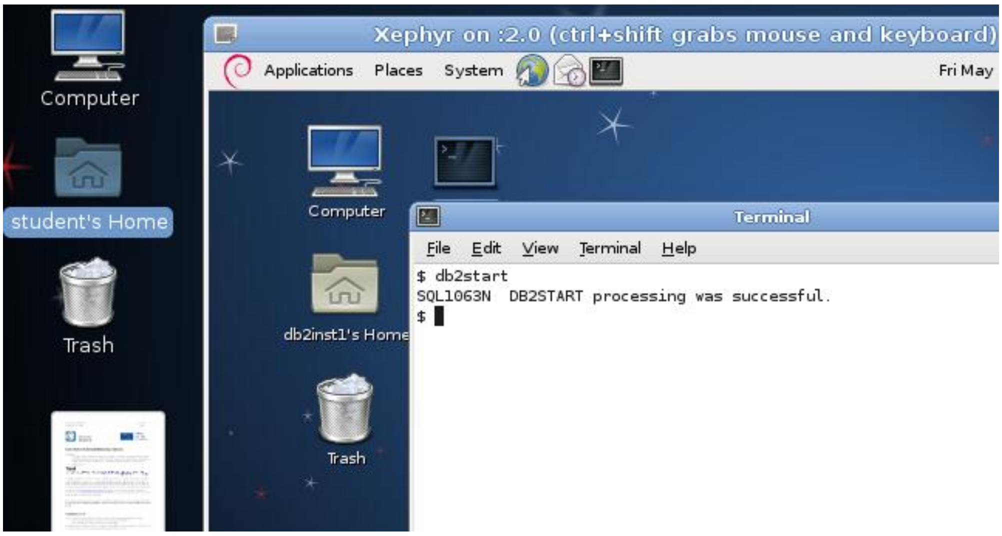
Applications -> System Tools -> -> New Login in a Window

and db2inst1 from the list in Xephyr window. Then start Terminal in the Xephyr window by selecting

Applications -> Accessories -> Terminal

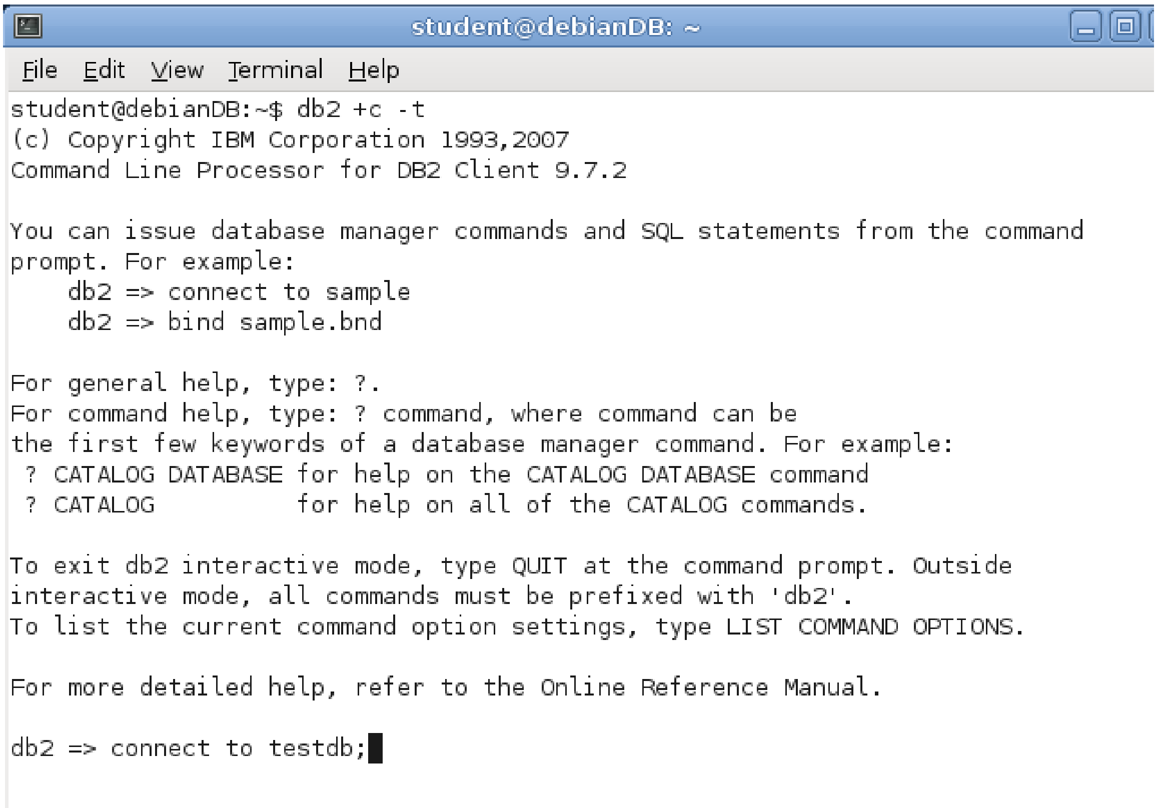
User db2inst1 can now start the DB2 instance entering the command

```
$ db2start
```


Computer
student's Home
Trash
Xephyr on :2.0 (ctrl+shift grabs mouse and keyboard)
Applications Places System
Fri May
Computer
db2inst1's Home
Trash
Terminal
File Edit View Terminal Help

```
$ db2start
SQL1063N  DB2START processing was successful.
$ 
```

After that db2inst1 or student user can start the Command Line Processor (CLP) of DB2 to be using SQL transactions by command line option "**+c**" and defining semicolon as the command terminator by command line option "**-t**" as follows, otherwise the default is autocommit mode and single line commands.



```
student@debianDB:~$ db2 +c -t
(c) Copyright IBM Corporation 1993,2007
Command Line Processor for DB2 Client 9.7.2

You can issue database manager commands and SQL statements from the command
prompt. For example:
    db2 => connect to sample
    db2 => bind sample.bnd

For general help, type: ?.
For command help, type: ? command, where command can be
the first few keywords of a database manager command. For example:
 ? CATALOG DATABASE for help on the CATALOG DATABASE command
 ? CATALOG          for help on all of the CATALOG commands.

To exit db2 interactive mode, type QUIT at the command prompt. Outside
interactive mode, all commands must be prefixed with 'db2'.
To list the current command option settings, type LIST COMMAND OPTIONS.

For more detailed help, refer to the Online Reference Manual.

db2 => connect to testdb;
```

DB2 session can be closed by a "quit" command. In addition to CLP, in version 9.7 we can still use the **Command Editor** (db2ce) and **Control Center** (db2cc) tools, but these will be dropped from DB2 version 10, and replaced by Eclipse based **Data Studio** of which in current DebianDB we have version 2.2.1.

For transaction concurrency tests we don't recommend use of the CLP from the Applications menu, since it may get blocked (to be sorted out later), but recommend the use from Terminal windows.

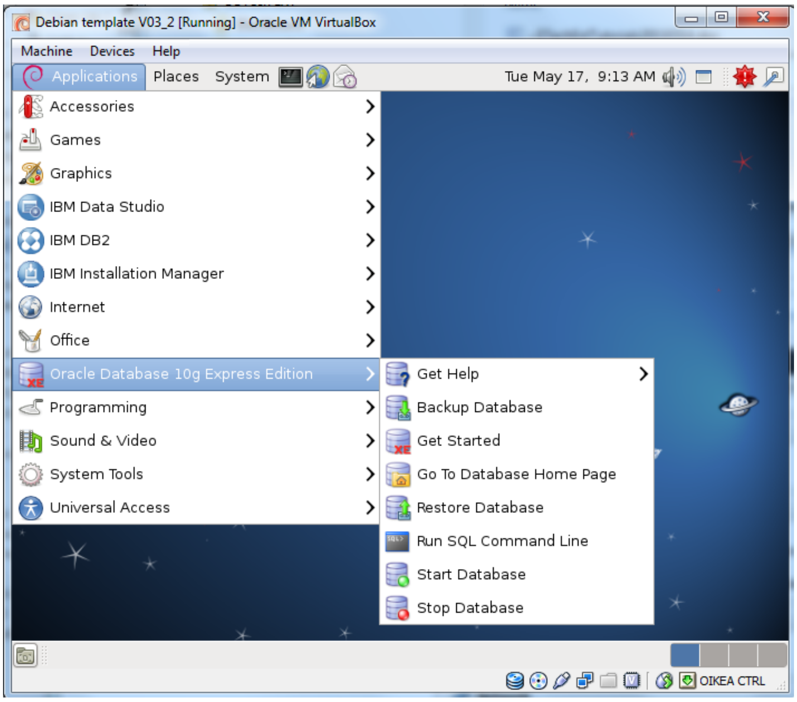
For allowing use of SAMPLE database to other usernames in DebianDB you need to grant them, as db2inst1 user, the access and other necessary database level and schema level object privileges, most conveniently using Control Center.

In DebianDB 6 we have already created database TESTDB, in which the student user can create own tables.

Oracle XE

You can access Oracle as student, dba, or oracle user. Howewver, the Oracle instance is not running all the time, but need to be started, for example by selections

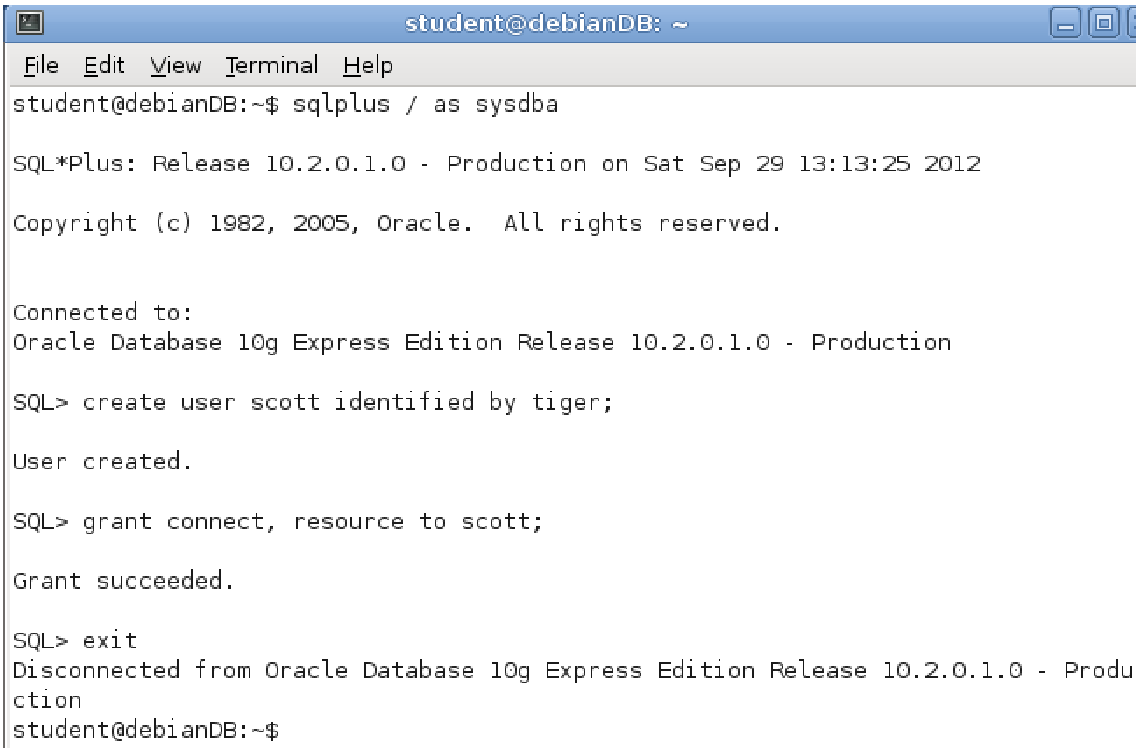
Applications – Oracle Database 10g Express Edition – Start Database



Since Oracle instance in DebianDB has been created by “student” user, the user can login using Oracle’s internal role sysdba and create some new database user, for example the typical test user “Scott” with the basic privileges CONNECT and RESOURCE for experimenting with own tables and SQL transactions.
The next figure shows how to start the traditional Oracle client program SQL*Plus as the creator of the database without entering the user name and password. SQL*Plus session will end by command “EXIT;”

The user Scott could start its SQL*Plus session by command

```
sqlplus scott/tiger
```



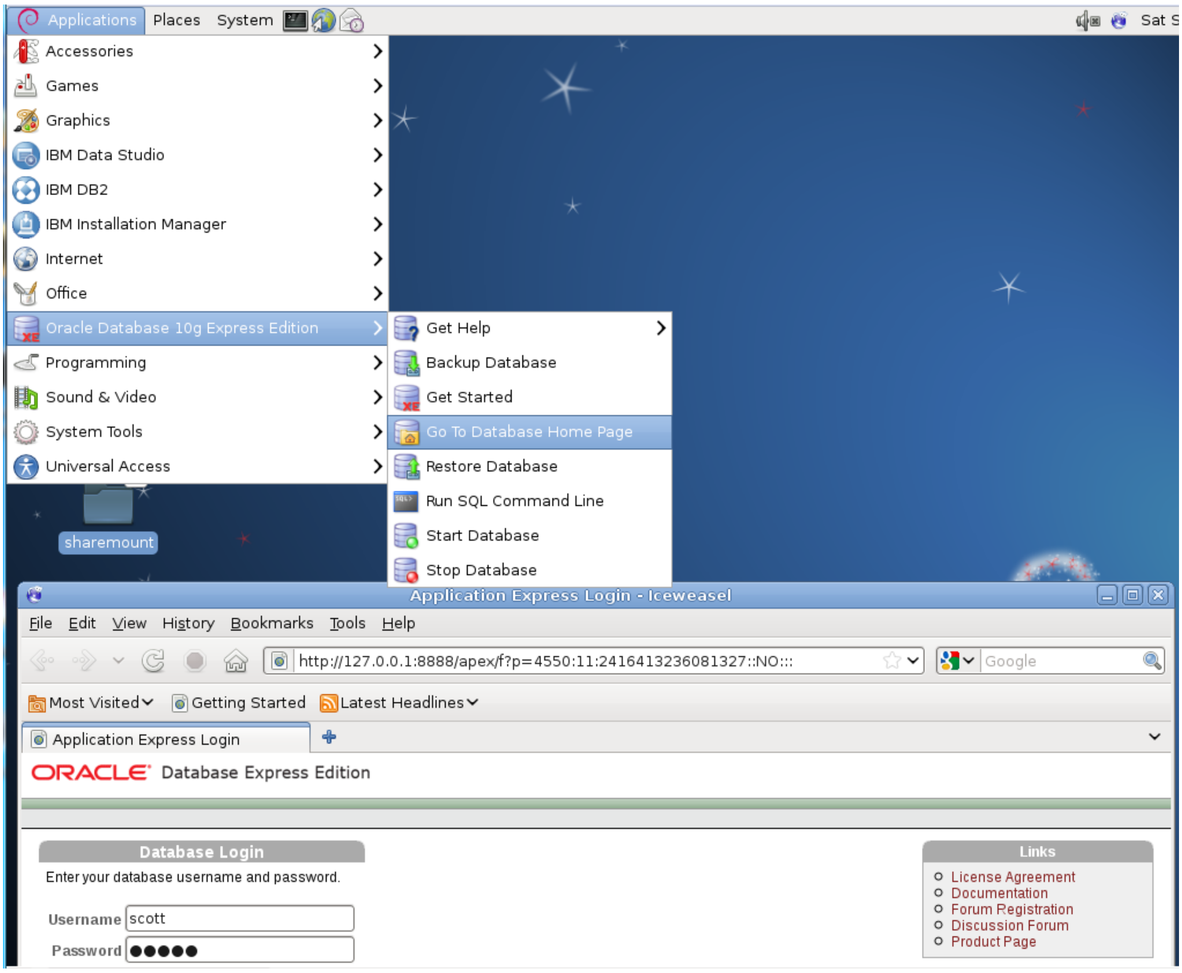
```
student@debianDB:~$ sqlplus / as sysdba

SQL*Plus: Release 10.2.0.1.0 - Production on Sat Sep 29 13:13:25 2012

Copyright (c) 1982, 2005, Oracle.  All rights reserved.


Connected to:
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

SQL> create user scott identified by tiger;

User created.

SQL> grant connect, resource to scott;

Grant succeeded.

SQL> exit
Disconnected from Oracle Database 10g Express Edition Release 10.2.0.1.0 - Produ
ction
student@debianDB:~$
```

SQL*Plus can also be started menu selections

Applications – Oracle Database 10g Express Edition – Run SQL Command Line

Another Oracle client interface is the browser based Application Express interface (see the snapshot below) client for Administration and SQL client started menu selections

Applications – Oracle Database 10g Express Edition – Go to Database Home Page



http://127.0.0.1:8888/apex/f?p=4550:11:2416413236081327::NO:::

Database Login

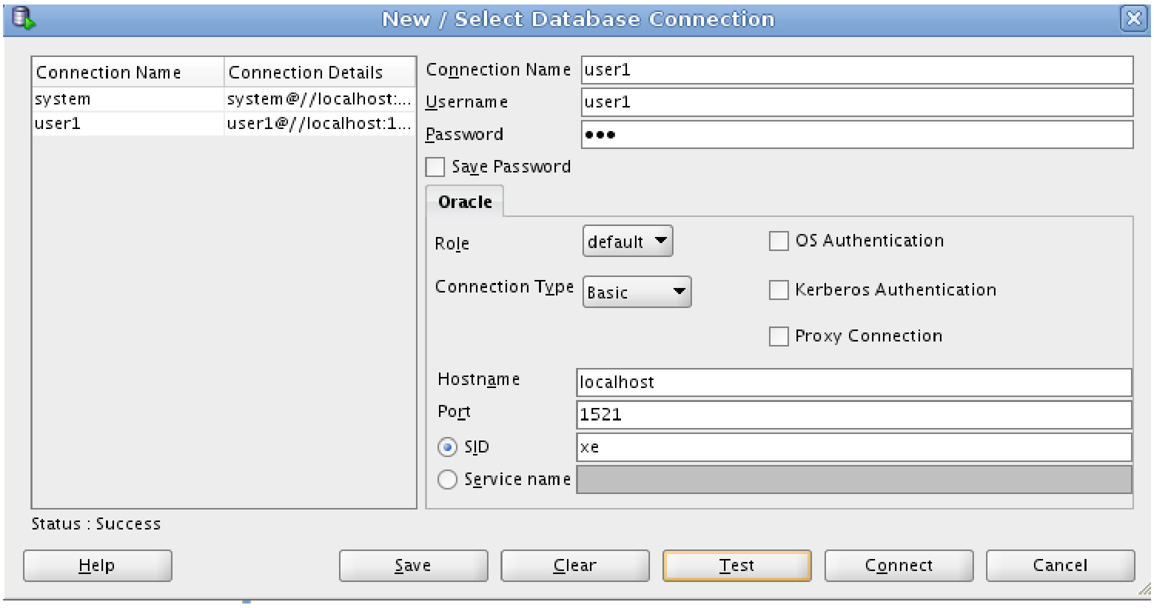
Enter your database username and password.

Username scott

Password ●●●●●

SQLDeveloper program is the third Oracle client interface available by menu selection
Applications – Programming – Oracle SQL Developer

in which connections for Oracle database access can be defined as follows



Connection Name	Connection Details
system	system@//localhost:...
user1	user1@//localhost:1...

Connection Name user1

Username user1

Password ●●●

□ Save Password

Role default

Connection Type Basic

□ OS Authentication

□ Kerberos Authentication

□ Proxy Connection

Hostname localhost

Port 1521

● SID xe

○ Service name

Status : Success

MySQL

MySQL server in DebianDB has been configured to be used by root only, but this can be fixed in terminal by starting root's session creating a database and granting other users access to that database, for example as follows:

```
student@debianDB:~$ su root
Password:
root@debianDB:/home/student# mysql -u root
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 36
Server version: 5.6.12 MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights
reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
Affiliates. Other names may be trademarks of their respective
Owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| TestDB             |
| mysql              |
+--------------------+
3 rows in set (0.01 sec)

mysql> CREATE DATABASE Test2;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL ON Test2.* TO 'student'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT ALL ON Test2.* TO 'dba'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> EXIT;
Bye
root@debianDB:/home/student# exit
exit
```

```
student@debianDB:~$ mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 37
...

mysql> USE Test2;
Database changed
mysql> SET AUTOCOMMIT=0;
Query OK, 0 rows affected (0.00 sec)

mysql> EXIT;
```

PostgreSQL

PostgreSQL instance in DebianDB has been created by root and root can grant access to other users like in the following dialogues:

```
student@debianDB:~$ su root
Password:
root@debianDB:/home/student# su postgres
postgres@debianDB:/home/student$ psql
psql (9.3.4, server 9.2.8)
Type "help" for help.

postgres=# CREATE USER student WITH login;
CREATE ROLE
postgres=# ALTER USER student WITH PASSWORD 'Student1';
ALTER ROLE
postgres=# CREATE DATABASE testdb;
CREATE DATABASE
postgres=# GRANT CONNECT ON DATABASE testdb TO student;
GRANT
postgres=# GRANT CREATE ON DATABASE testdb TO student;
GRANT
postgres=# \q
postgres@debianDB:/home/student$ exit
exit
root@debianDB:/home/student# exit
exit
There are stopped jobs.
root@debianDB:/home/student#
root@debianDB:/home/student# exit
exit
student@debianDB:~$
```

Now user student can login to PostgreSQL and enter following commands

```
psql testdb

BEGIN WORK;
CREATE TABLE T1 (id INT, c VARCHAR(10)) ;
INSERT INTO T1  (id, c) VALUES (1, 'foo1-1');
COMMIT;
\q
```

In DebianDB we also have graphical tools for PostgreSQL, but use of these will be documented in future.

Pyrrho DBMS

More information on this exciting DBMS product is available at http://pyrrhodb.uws.ac.uk/index.htm.
The manual is available at /opt/OSP/Pyrrho.docx.
Pyrrho DBMS is written in C# language and can be used in Linux on the ported .NET platform called Mono.
The following sample gives a short introduction on using Pyrrho in DebianDB:

```
# testing mono (~ .NET implementation on Linux) in Debian
student@debianDB:~$ gedit hello.cs
student@debianDB:~$ gmcs hello.cs
student@debianDB:~$ mono hello.exe
Hello Mono World

# testing Pyrrho
# Start the server in a terminal window
student@debianDB:~$ cd Pyrrho
student@debianDB:~$ cd Pyrrho
student@debianDB:~/Pyrrho$ ./server.sh
Enter to start upOpen Source Pyrrho (c) 2014 Malcolm Crowe and University
of the West of Scotland
5.1 (16 April 2014) www.pyrrhodb.com
MongoDB service started on port 27017
PyrrhoDBMS protocol on 127.0.0.1:5433
HTTP service started on port 8180
HTTPS service started on port 8133

#Start a client in another terminal window
student@debianDB:~$ cd Pyrrho
student@debianDB:~$Pyrrho# mono /opt/OSP/PyrrhoCmd.exe testdb

-- Test commands: --
```

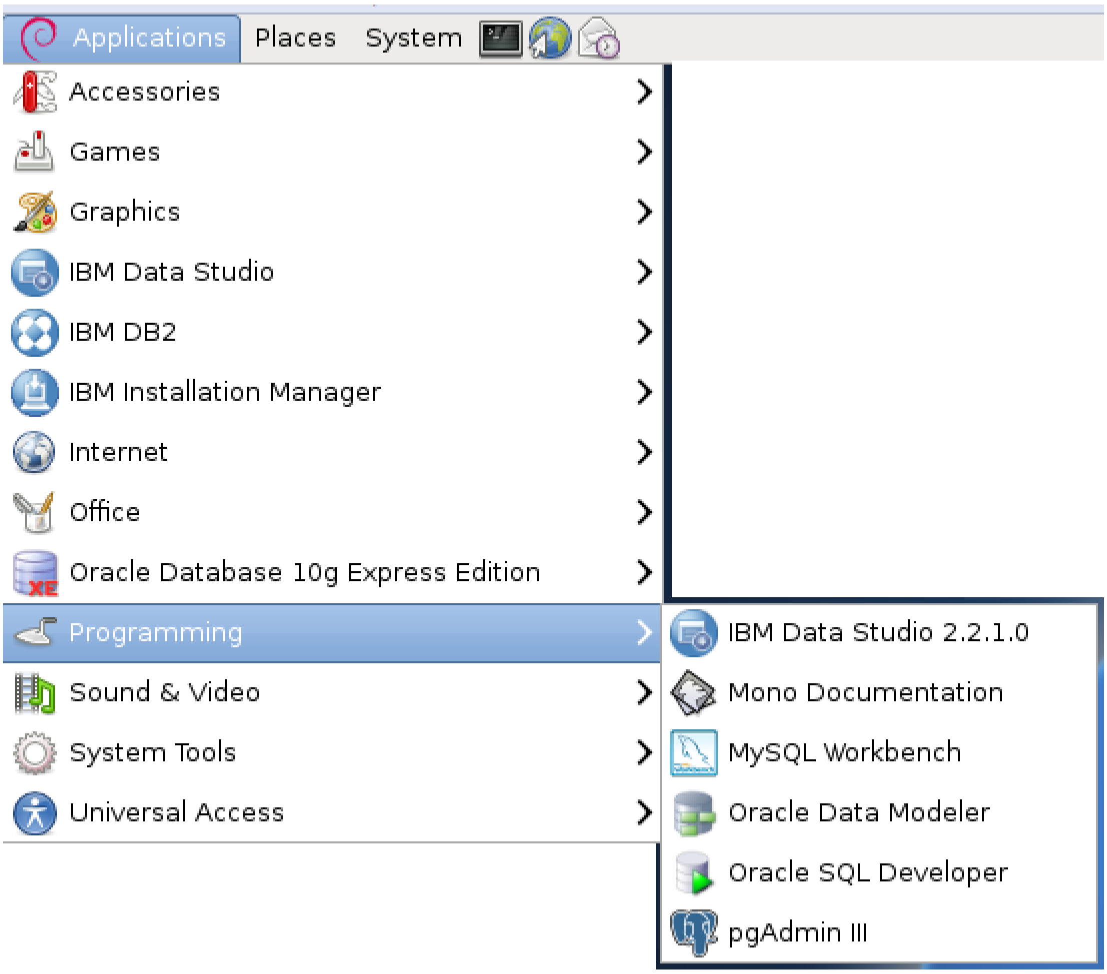
```
BEGIN TRANSACTION ;
CREATE TABLE T1 (id INT, c CHAR) ;
INSERT INTO T1 (id, c) VALUES (1, 'foo-1');
COMMIT;
QUIT;
```

Note: Multi-line commands need to be enclosed in brackets, for example

```
[CREATE TABLE T1 (
id INT,
c CHAR) ;]
```

Graphical tools in DebianDB

The user instructions on the following graphical workbench tools is out of scope of this tutorial, but can be found from Internet.

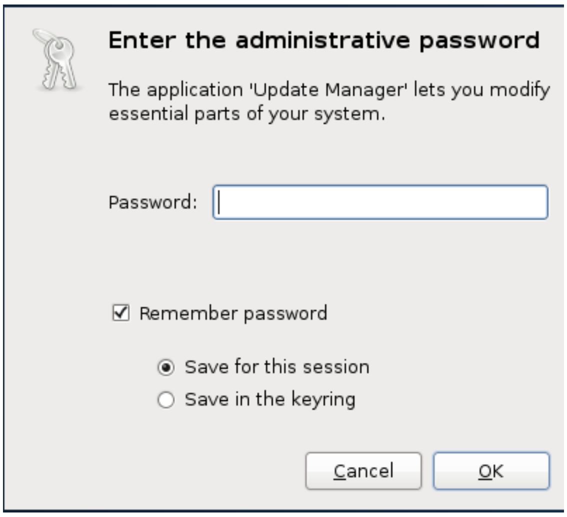


Keeping the Debian platform up-to-date

To keep your lab up-to-date with the Debian 6 software developments you need access to Internet.

From the menu bar select

System -> Administration -> Update Manager



Enter the administrative password

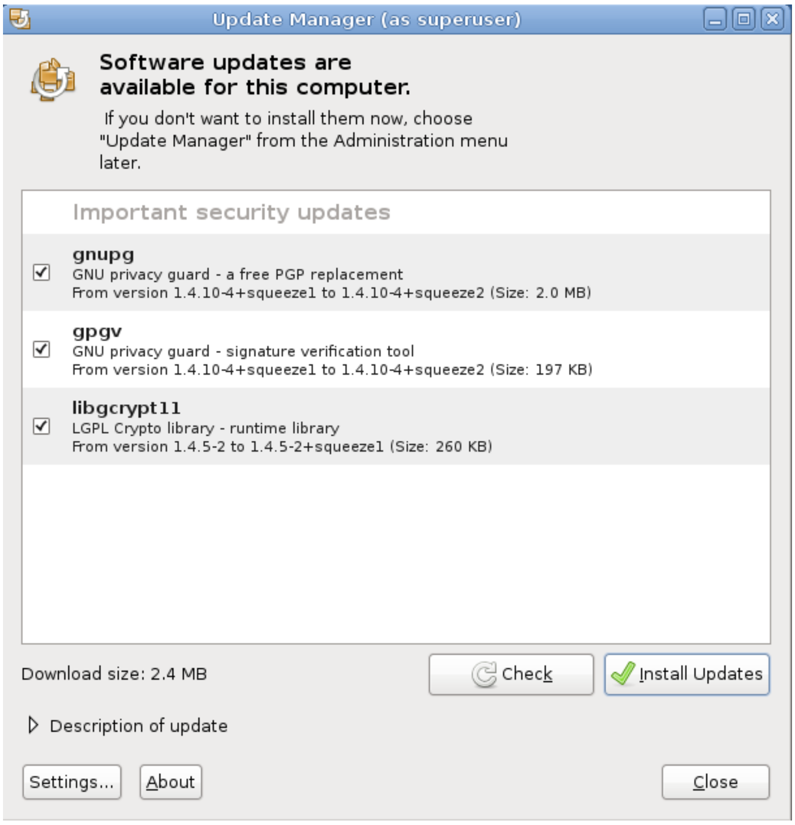
The application 'Update Manager' lets you modify essential parts of your system.

Password:

☑ Remember password

● Save for this session
○ Save in the keyring

Enter root's password and press OK. Next you will get into Update Manager form by buttons on it you can ask to Check for possible updates, then select the updates you prefer, and finally proceed to Install the Updates:



Update Manager (as superuser)

Software updates are available for this computer.

If you don't want to install them now, choose "Update Manager" from the Administration menu later.

Important security updates

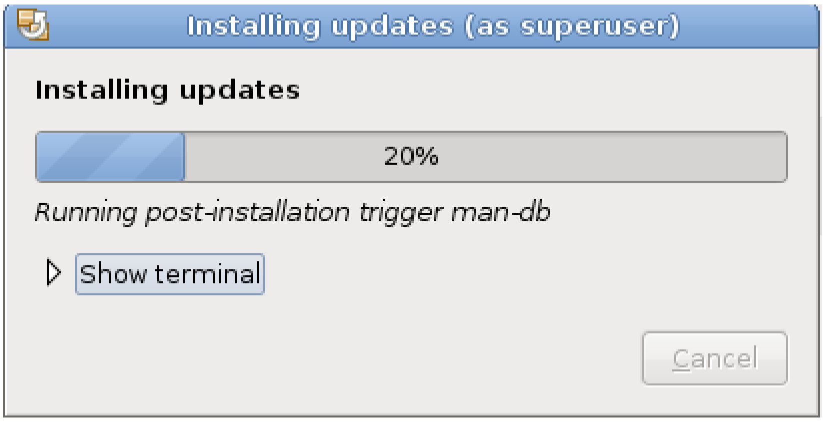
☑ **gnupg**
GNU privacy guard - a free PGP replacement
From version 1.4.10-4+squeeze1 to 1.4.10-4+squeeze2 (Size: 2.0 MB)

☑ **gpgv**
GNU privacy guard - signature verification tool
From version 1.4.10-4+squeeze1 to 1.4.10-4+squeeze2 (Size: 197 KB)

☑ **libgcrypt11**
LGPL Crypto library - runtime library
From version 1.4.5-2 to 1.4.5-2+squeeze1 (Size: 260 KB)

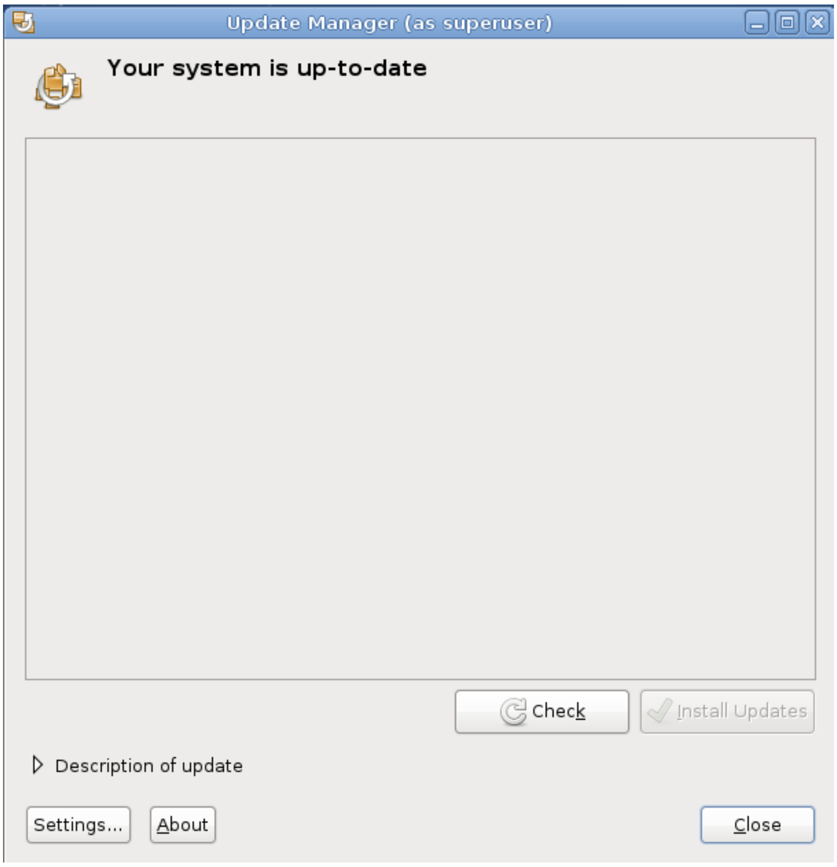
Download size: 2.4 MB

Selecting “Install Updates” the selected updates will be dowloaded from the website of Debian.org and installed in the virtual machine



Running post-installation trigger man-db

Finally press OK to return to use the system.



Your system is up-to-date

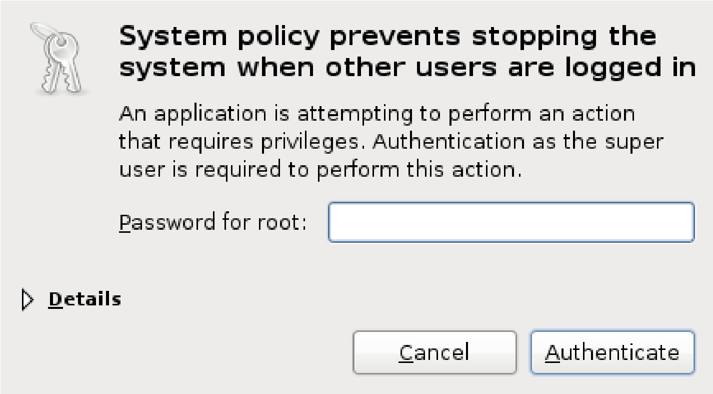
The installed DBMS products need to be updated separately. From time to time we will update the DebianDB available on our website. Currently we are limited by the 32bit architecture and we try to keep the zip file in a reasonable size.

Closing DebianDB

To shutdown the DebianDB select

System -> Shut Down… -> Shut Down

and if there still are sessions of other users the system will prompt for root's password to commit the shutdown



System policy prevents stopping the system when other users are logged in

An application is attempting to perform an action that requires privileges. Authentication as the super user is required to perform this action.

Password for root: