June 10 report

Date:	10/06/2020	Name:	Ankitha c c
Course:	Vlsi	USN:	4al16ec004
Topic:	MOS transistor basics-I	Semester & Section:	8th & "A" section
Github Repository:	ankitha-c-c		

FORENOON SESSION DETAILS

Image of session

CMOS Digital VLSI Design

Lecture-01
MOS TRANSISTER BASICS-I

Report – Report can be typed or hand written for up to two pages.

1.MOS transistor basics-I

A metal–oxide–semiconductor field-effect transistor (MOSFET, MOS-FET, or MOS FET) is a field-effect transistor (FET with an insulated gate) where the voltage determines the conductivity of the device. It is used for switching or amplifying signals. The ability to change conductivity with the amount of applied voltage can be used for amplifying or switching electronic signals. MOSFETs are now even more common than BJTs (bipolar junction transistors) in digital and analog circuits.

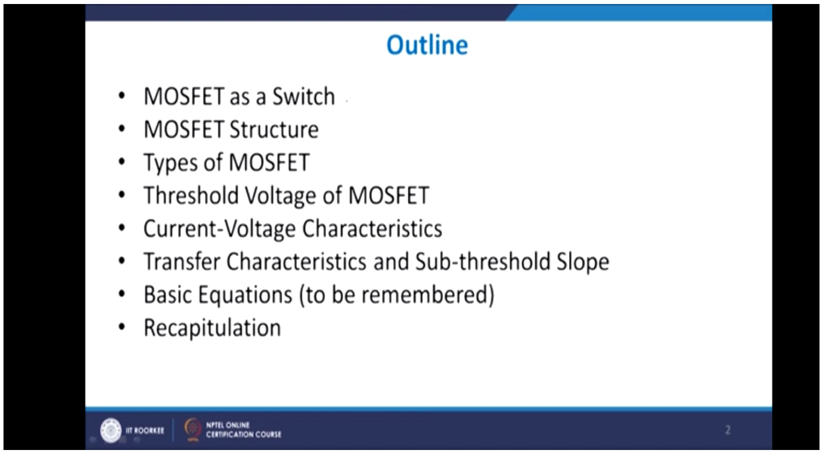
A MOSFET is by far the most common transistor in digital circuits, as hundreds of thousands or millions of them may be included in a memory chip or microprocessor. Since they can be made with either p-type or n-type semiconductors, complementary pairs of MOS transistors can be used to make switching circuits with very low power consumption, in the form of CMOS logic.

Metal Oxide Semiconductor Fet (Mosfet) from stooty s

MOSFETs are particularly useful in amplifiers due to their input impedance being nearly infinite which allows the amplifier to capture almost all the incoming signal. The main advantage is that it requires almost no input current to control the load current, when compared with bipolar transistors. MOSFETs are available in two basic forms:

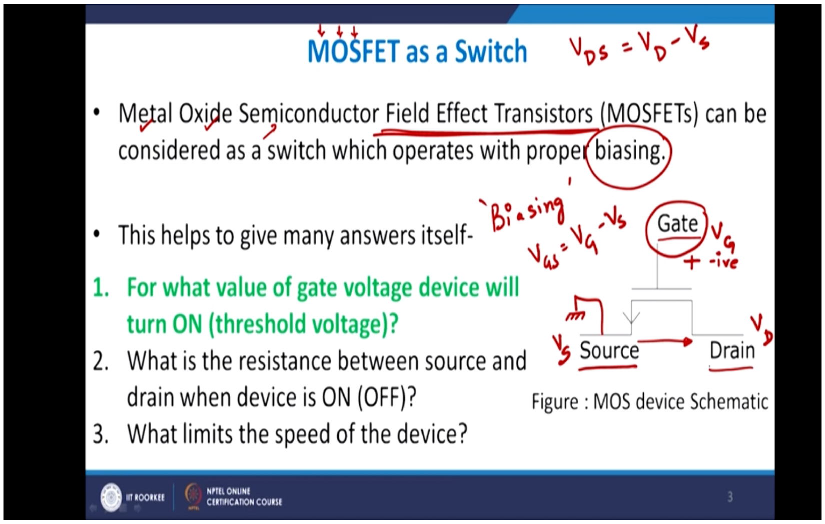
Depletion Type: The transistor requires the Gate-Source voltage (VGS) to switch the device "OFF". The depletion-mode MOSFET is equivalent to a "Normally Closed" switch.

Enhancement Type: The transistor requires a Gate-Source voltage(VGS) to switch the device "ON". The enhancement-mode MOSFET is equivalent to a "Normally Open" switch.



Outline

- MOSFET as a Switch
- MOSFET Structure
- Types of MOSFET
- Threshold Voltage of MOSFET
- Current-Voltage Characteristics
- Transfer Characteristics and Sub-threshold Slope
- Basic Equations (to be remembered)
- Recapitulation



MOSFET as a Switch

$V_{DS} = V_D - V_S$

- Metal Oxide Semiconductor Field Effect Transistors (MOSFETs) can be considered as a switch which operates with proper biasing.

'Biasing'

$V_{GS} = V_G - V_S$

- This helps to give many answers itself-

1. For what value of gate voltage device will turn ON (threshold voltage)?
2. What is the resistance between source and drain when device is ON (OFF)?
3. What limits the speed of the device?

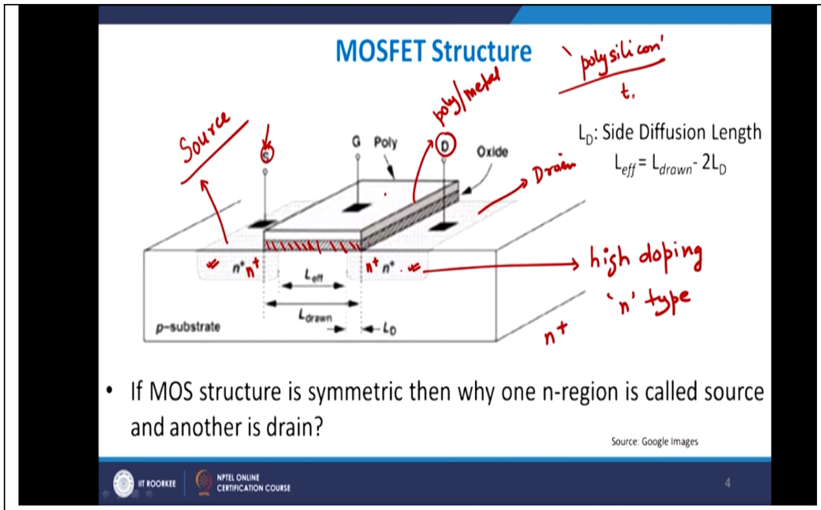
Figure : MOS device Schematic

It is a four-terminal device with source(S), gate (G), drain (D) and body (B) terminals. The body is frequently connected to the source terminal, reducing the terminals to three. It works by varying the width of a channel along which charge carriers flow (electrons or holes).

The charge carriers enter the channel at source and exit via the drain. The width of the channel is controlled by the voltage on an electrode is called gate which is located between source and drain. It is insulated from the channel near an extremely thin layer of metal oxide. A metal-insulator-semiconductor field-effect transistor or MISFET is a term almost synonymous with MOSFET. Another synonym is IGFET for the insulated-gate field-effect transistor.

MOSFET Operation

The working of a MOSFET depends upon the MOS capacitor. The MOS capacitor is the main part of MOSFET. The semiconductor surface at the below oxide layer which is located between source and drain terminals. It can be inverted from p-type to n-type by applying positive or negative gate voltages.



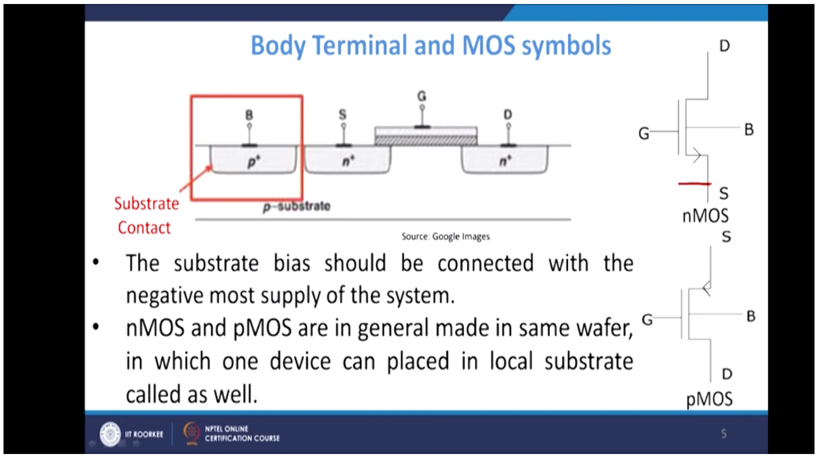
MOSFET Structure

L_D: Side Diffusion Length

$L_{eff} = L_{drawn} - 2L_D$

- If MOS structure is symmetric then why one n-region is called source and another is drain?

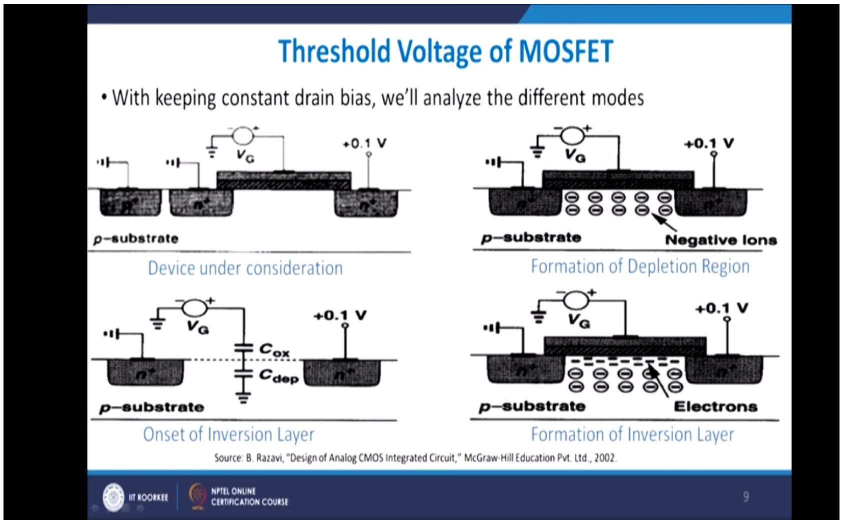
Source: Google Images



Body Terminal and MOS symbols

Source: Google Images

- The substrate bias should be connected with the negative most supply of the system.
- nMOS and pMOS are in general made in same wafer, in which one device can placed in local substrate called as well.



Threshold Voltage of MOSFET

- With keeping constant drain bias, we'll analyze the different modes

Device under consideration

Formation of Depletion Region

Onset of Inversion Layer

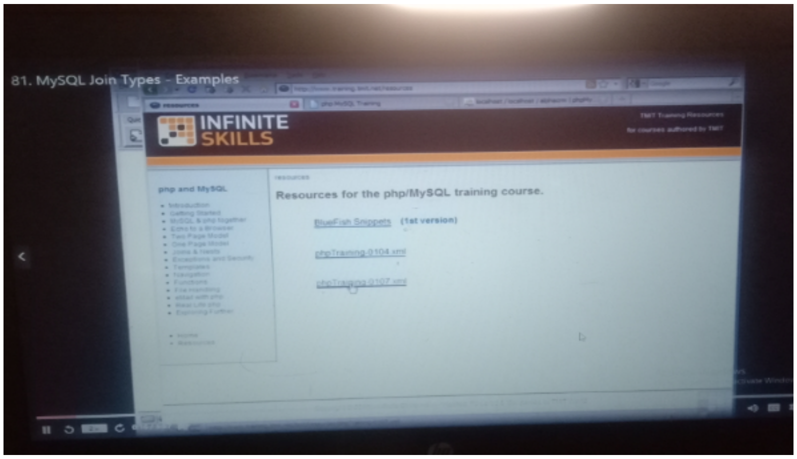
Formation of Inversion Layer

Source: B. Razavi, "Design of Analog CMOS Integrated Circuit," McGraw-Hill Education Pvt. Ltd., 2002.

Date:	10/06/2020	Name:	Ankitha c c
Course:	Mysql	USN:	4al16ec004
Topic:		Semester & Section:	8th & a section

AFTERNOON SESSION DETAILS

Image of session



81. MySQL Join Types - Examples

php and MySQL

- Introduction
- Getting Started
- MySQL & php together
- Echo to a Browser
- Two Page Model
- One Page Model
- Joins & Heists
- Exceptions and Security
- Templates
- Navigation
- Functions
- File Handling
- eMail with php
- Real Life php
- Exploring Further

- Home
- Resources

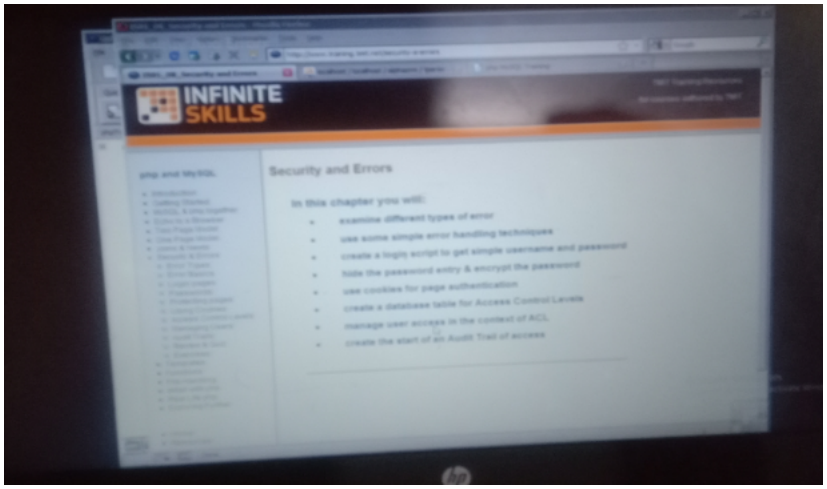
resources

Resources for the php/MySQL training course.

BlueFish Snippets (1st version)

phpTraining-0104.xml

phpTraining-0107.xml



Security and Errors

In this chapter you will:

- examine different types of error
- use some simple error handling techniques
- create a login script to get simple username and password
- hide the password entry & encrypt the password
- use cookies for page authentication
- create a database table for Access Control Levels
- manage user access in the context of ACL
- create the start of an Audit Trail of access

Report – Report can be typed or hand written for up to two pages.

1. MYSQL JOINS

This MySQL tutorial explains how to use MySQL JOINS (inner and outer) with syntax, visual illustrations, and examples.

Description

MySQL JOINS are used to retrieve data from multiple tables. A MySQL JOIN is performed whenever two or more tables are joined in a SQL statement.

There are different types of MySQL joins:

MySQL INNER JOIN (or sometimes called simple join)

MySQL LEFT OUTER JOIN (or sometimes called LEFT JOIN)

MySQL RIGHT OUTER JOIN (or sometimes called RIGHT JOIN)

So let's discuss MySQL JOIN syntax, look at visual illustrations of MySQL JOINS, and explore MySQL JOIN examples.

INNER JOIN (simple join)

Chances are, you've already written a statement that uses a MySQL INNER JOIN. It is the most common type of join. MySQL INNER JOINS return all rows from multiple tables where the join condition is met.

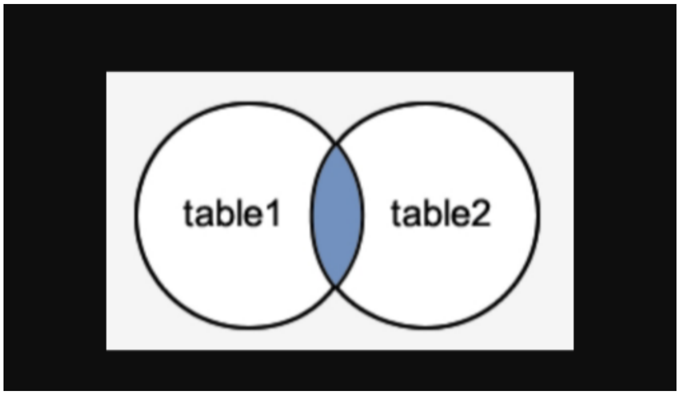
Syntax

The syntax for the INNER JOIN in MySQL is:

```
SELECT columns

FROM table1

INNER JOIN table2

ON table1.column = table2.column;
```

Visual Illustration

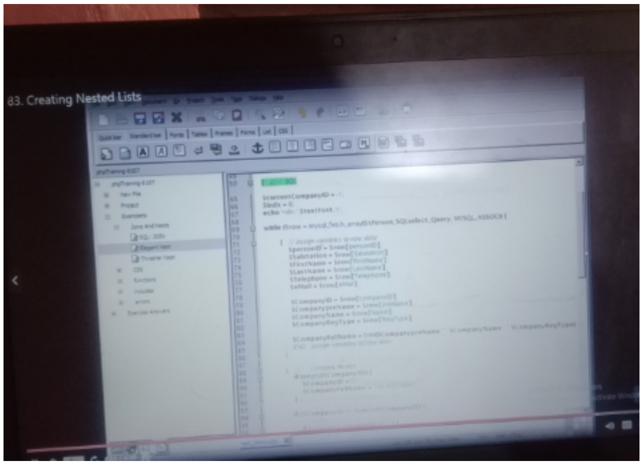
In this visual diagram, the MySQL INNER JOIN returns the shaded area:



Example

Here is an example of a MySQL INNER JOIN:

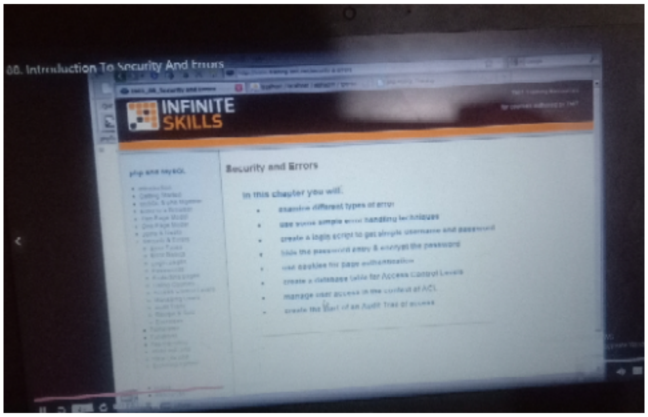
```
SELECT suppliers.supplier_id, suppliers.supplier_name, orders.order_date

FROM suppliers

INNER JOIN orders

ON suppliers.supplier_id = orders.supplier_id;
```



83. Creating Nested Lists

```
$currentCompanyID = -1;
$indx = 0;
echo "<div> $textFont ";

while ($row = mysql_fetch_array($Person_SQLselect_Query, MYSQL_ASSOC))

    { // assign variables to row data
      $personID = $row[[illegible]];
      $Salutation = $row[[illegible]];
      $FirstName = $row[[illegible]];
      $LastName = $row[[illegible]];
      $Telephone = $row[[illegible]];
      $eMail = $row[[illegible]];

      $companyID = $row[[illegible]];
      $CompanypreName = $row[[illegible]];
      $CompanyName = $row[[illegible]];
      $CompanyRegType = $row[[illegible]];

      $CompanyFullName = trim($CompanypreName . $CompanyName . $CompanyRegType);
```

2. PHP Errors and security



88. Introduction To Security And Errors

Security and Errors

In this chapter you will:

- examine different types of error
- use some simple error handling techniques
- create a login script to get simple username and password
- hide the password away & encrypt the password
- use cookies for page authentication
- create a database table for Access Control Levels
- manage user access in the context of ACL
- create the start of an Audit Trail of access

Security

Database security entails allowing or disallowing user actions on the database and the objects within it. When you will create an database application, the security policy is the first step. An application security policy is a list of application security requirements and rules that regulate user access to database objects. This chapter discusses aspects of application security and MySQL Database features which contains the following topics :

Contents:

MySQL General Security Issues

The MySQL Access Privilege System

MySQL User Account Management

MySQL general security issues

Security Guidelines :

Except MySQL root account does not permit anyone to access the user table in the MySQL database.

Use the GRANT and REVOKE statements to control access to MySQL. Do not grant unnecessary privileges and never grant privileges to all hosts.

Never store simple text passwords in your database. Store the hash value using SHA2(), SHA1(), MD5() functions or other hashing function in a different way. Try to use a complex password.

Try to use a firewall and put MySQL behind the firewall.

3306 is the default user port of MySQL and this port should not be accessible from untrusted hosts. You can scan the ports from Internet using a tool such as nmap. From a remote machine you can check whether the port is open or not with this command: shell> telnet server_host 3306. If telnet hangs or the connection is refused, the port is blocked. If you get a connection and some garbage characters, the port is open and should be closed on your firewall or router, unless you really have a good reason to keep it open.

Some applications access MySQL database for different a purpose. Never trust these input data entered by the user and must validate properly before access database.

Do not transmit unencrypted data over the Internet. Use an encrypted protocol such as SSL (MySQL supports internal SSL connections) or SSH.

Use tcpdump and strings utilities. By issuing this command shell> tcpdump -l -i eth0 -w - src or dst port 3306 | strings you can check whether MySQL data streams are unencrypted or not.

Keeping Passwords Secure:

End-User Guidelines for Password Security

Use the -p or --password option on the command line with no password value specified. Here is the command

```
shell> mysql -u user_id -p database_name

Enter password : ***********
```

When you input the password it will not visible.

Store your password in an option file. For example Unix you can list your password in [client] section of the .my.cnf file in your home directory and to keep password safe, set the file access mode to 400 or 600.

Administrator Guidelines for Password Security : MySQL stores passwords for user accounts in the mysql.user table. Therefore this table should not be accessed by any nonadministrative accounts. User account password must reset time to time.

Passwords and Logging : Passwords can be written as plain text in SQL statements such as CREATE USER, GRANT, and SET PASSWORD, or statements that invoke the PASSWORD() function. If these statements are logged by the MySQL server as written, such passwords become available to anyone with access to the logs. Beginning with MySQL 5.6.3, statement logging is modified so that passwords do not appear in plain text for the following statements:

CREATE USER ... IDENTIFIED BY ...GRANT ... IDENTIFIED BY ...SET PASSWORD ...SLAVE START ... PASSWORD = ... (as of 5.6.4)CREATE SERVER ... OPTIONS(... PASSWORD ...) (as of 5.6.9)ALTER SERVER ... OPTIONS(... PASSWORD ...) (as of 5.6.9)

Passwords in those statements are rewritten not to appear literally in statement text, for the general query log, slow query log, and binary log. Rewriting does not apply to other statements.

Password Hashing in MySQL : MySQL lists user accounts in the user table of the MySQL database. Each MySQL account can be assigned a password, although the user table does not store the cleartext version of the password, but a hash value computed from it.

Implications of Password Hashing Changes in MySQL 4.1 for Application Programs : An upgrade to MySQL version 4.1 or later can cause compatibility issues for applications that use PASSWORD() to generate passwords for their own purposes. Applications really should not do this, because PASSWORD() should be used only to manage passwords for MySQL accounts.

The validate_password plugin (available as of MySQL 5.6.6) can be used to test passwords and improve security.

Making MySQL Secure Against Attackers :

To make a MySQL system secure, you should maintain the following suggestions :

Require all MySQL accounts to have a password.

Make sure that the only Unix user account with read or write privileges in the database directories is the account that is used for running mysqld.

Never run the MySQL server as the Unix root user

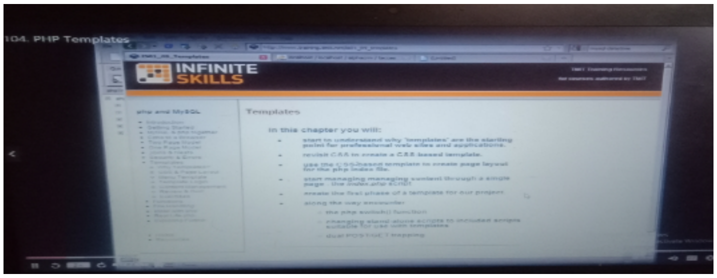
Do not grant the FILE privilege to nonadministrative users

Do not permit the use of symlinks to tables.

Stored programs and views should be written using the security guidelinesIf you do not trust your DNS, you should use IP addresses rather than hostnames in the grant tables.

If you want to restrict the number of connections permitted to a single account, you can do so by setting the max_user_connections variable in mysqld.

3. Building a template page



Templates

In this chapter you will:

- start to understand why "templates" are the starting point for professional web sites and applications.
- revisit CSS to create a CSS based template.
- use the CSS-based template to create page layout for the php index file.
- start managing managing content through a single page - the index.php script.
- create the first phase of a template for our project.
- along the way encounter:
  - the php switch() function
  - changing stand-alone scripts to included scripts suitable for use with templates
  - dual POST/GET trapping

Creating a Custom Template

In the simplest case, a template consists of two files: a template file, which has a .tpl extension, and a special file info.xml. The info.xml file has important metadata about the template. A third file is optional, which is the preview image file. This preview file provides a thumbnail image illustrating the appearance of the generated report.

One of the easiest ways to create a custom template is to make a copy of any existing template.

For example, the following procedure describes how to make a custom template based on the Text Basic template.

Navigate to the folder where the templates are stored. Assuming that MySQL Workbench has been installed into the default location on Windows, this would be C:\Program Files\MySQL\MySQL Workbench 5.0 SE\modules\data\wb_model_reporting.

Copy the Text_Basic.tpl folder. The copy can be given any suitable name; for example, Custom_Basic.tpl.

Edit the info.xml file to reflect your custom template. The unedited file in this case is shown here:

```
<?xml version="1.0"?>
<data>
  <value type="object" struct-name="workbench.model.reporting.TemplateInfo"
  id="{BD6879ED-814C-4CA3-A869-9864F83B88DF}" struct-checksum="0xb46b524d">
    <value type="string" key="description">
      A basic TEXT report listing schemata and objects.
    </value>
    <value type="string" key="name">HTML Basic Frame Report</value>
    <value type="list" content-type="object"
    content-struct-name="workbench.model.reporting.TemplateStyleInfo"
    key="styles">
      <value type="object" struct-name="workbench.model.reporting.TemplateStyleInfo"
      id="{7550655C-CD4B-4EB1-8FAB-AAEE49B2261E}" struct-checksum="0xab08451b">
        <value type="string" key="description">
          Designed to be viewed with a fixed sized font.
        </value>
        <value type="string" key="name">Fixed Size Font</value>
        <value type="string" key="previewImageFileName">
          preview_basic.png
        </value>

        <value type="string" key="styleTagValue">fixed</value>

      </value>

    </value>

    <value type="string" key="mainFileName">report.txt</value>

  </value>

</data>
```

The file defines wwo objects: the TemplateInfo object and the TemplateStyleInfo object. These objects contain information about the template that will be displayed in the DBDoc Model Reporting wizard main page.

Change the object GUIDs that are used in the file. In this example, there are two that need replacing:

```
id="{BD6879ED-814C-4CA3-A869-9864F83B88DF}"

...

id="{7550655C-CD4B-4EB1-8FAB-AAEE49B2261E}"
```

Generate two new GUIDS. This is done using a suitable command-line tool, and there are also free online tools that generate GUIDs. MySQL's UUID() function also generates GUIDs:

```
mysql> SELECT UUID();

+--------------------------------------+

| UUID()                               |

+--------------------------------------+

| 648f4240-7d7a-11e0-870b-89c43de3bd0a |

+--------------------------------------+
```

Once you have the new GUID values, edit the info.xml file accordingly.

Edit the textual information for the TemplateInfo and TemplateStyleInfo objects to reflect the purpose of the custom template.