**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: | 10/06/2020 | Name: | Gaganashree P |
| Course: | Vlsi | USN: | 4al15ec024 |
| Topic: | MOS transistor basics-I | Semester & Section: | 8th &"A" section |
| Github Repository: | Gaganashree-P |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| Report –  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.    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. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DAILY ASSESSMENT FORMAT**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  |  | | --- | --- | --- | --- | | Date: | 10-06-20 | Name: | Gaganashree P | | Course: | Beginner PHP and MySQL Tutorial | USN: | 4AL15EC024 | | Topic: | PHP and MySQL Tutorial | Semester & Section: | 8th A | | Github Repository: | Gaganashree-P |  |  | | |  | |  | |  | |
|  |  |  | |  | |
|  |  |  | |  | |
| **AFTERNOON SESSION DETAILS** | | | | | |
| **Image of session** | | | | | |
| MySQL JOINS  MySQL JOINS are used with SELECT statement. It is used to retrieve data from multiple tables. It is performed whenever you need to fetch records from two or more tables.  There are three types of [MySQL](https://www.javatpoint.com/mysql-tutorial) 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)   MySQL Inner JOIN (Simple Join)  The [MySQL INNER JOIN](https://www.javatpoint.com/mysql-inner-join) is used to return all rows from multiple tables where the join condition is satisfied. It is the most common type of join.  **Syntax:**   1. **SELECT** columns 2. **FROM** table1 3. **INNER** JOIN table2 4. **ON** table1.**column** = table2.**column**;   **Image representation:**  mysql join 1  **Let's take an example:**  Consider two tables "officers" and "students", having the following data.  mysql join 2  **Execute the following query:**   1. **SELECT** officers.officer\_name, officers.address, students.course\_name 2. **FROM** officers 3. **INNER** JOIN students 4. **ON** officers.officer\_id = students.student\_id;   **Output:**  mysql join 3  MySQL Left Outer Join  The LEFT OUTER JOIN returns all rows from the left hand table specified in the ON condition and only those rows from the other table where the join condition is fulfilled.  **Syntax:**   1. **SELECT** columns 2. **FROM** table1 3. LEFT [OUTER] JOIN table2 4. **ON** table1.**column** = table2.**column**;   **Image representation:**  mysql join 4  **Let's take an example:**  Consider two tables "officers" and "students", having the following data.  mysql join 5  **Execute the following query:**   1. **SELECT**  officers.officer\_name, officers.address, students.course\_name 2. **FROM** officers 3. LEFT JOIN students 4. **ON** officers.officer\_id = students.student\_id;   **Output:**  mysql join 6  MySQL Right Outer Join  The MySQL Right Outer Join returns all rows from the RIGHT-hand table specified in the ON condition and only those rows from the other table where he join condition is fulfilled.  **Syntax:**   1. **SELECT** columns 2. **FROM** table1 3. RIGHT [OUTER] JOIN table2 4. **ON** table1.**column** = table2.**column**;   **Image representation:**  mysql join 7  **Let's take an example:**  Consider two tables "officers" and "students", having the following data.  mysql join 8  **Execute the following query:**   1. **SELECT** officers.officer\_name, officers.address, students.course\_name, students.student\_name 2. **FROM** officers 3. RIGHT JOIN students 4. **ON** officers.officer\_id = students.student\_id;   **Output:**  mysql join 9  PHP Error Handling  When creating scripts and web applications, error handling is an important part. If your code lacks error checking code, your program may look very unprofessional and you may be open to security risks.  Different error handling methods:   * Simple "die()" statements * Custom errors and error triggers * Error reporting  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.   1. 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. 2. Copy the Text\_Basic.tpl folder. The copy can be given any suitable name; for example, Custom\_Basic.tpl. 3. Edit the info.xml file to reflect your custom template. The unedited file in this case is shown here: 4. <?xml version="1.0"?> 5. <data> 6. <value type="object" struct-name="workbench.model.reporting.TemplateInfo" 7. id="{BD6879ED-814C-4CA3-A869-9864F83B88DF}" struct-checksum="0xb46b524d"> 8. <value type="string" key="description"> 9. A basic TEXT report listing schemata and objects. 10. </value> 11. <value type="string" key="name">HTML Basic Frame Report</value> 12. <value type="list" content-type="object" 13. content-struct-name="workbench.model.reporting.TemplateStyleInfo" 14. key="styles"> 15. <value type="object" struct-name="workbench.model.reporting.TemplateStyleInfo" 16. id="{7550655C-CD4B-4EB1-8FAB-AAEE49B2261E}" struct-checksum="0xab08451b"> 17. <value type="string" key="description"> 18. Designed to be viewed with a fixed sized font. 19. </value> 20. <value type="string" key="name">Fixed Size Font</value> 21. <value type="string" key="previewImageFileName"> 22. preview\_basic.png 23. </value> 24. <value type="string" key="styleTagValue">fixed</value> 25. </value> 26. </value> 27. <value type="string" key="mainFileName">report.txt</value> 28. </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. | | | | | |