

**NET320-Database Systems**

**SWS213-Database Design**

**Spring ‘17**

**Lab 3b**

**Lab 3b- Creating Table, Query(Select, Where, Orderby), Datatype. – 2**

**Objective**

Upon successful completion of this lab the student will be able to:

* Create schemas and tables.
* Datatypes
* Use of basic SQL commands- SELECT

**Grading Scheme**

* This class assignment is due on 24 January 2017 at 5:00 PM.
* Students should save the Workbench file and upload to moodle.
* Save the file with your Name followed by ID.
  + Task 1 - 10 pts
  + Task 2 - 10 pts

**Back ground**

* Students should be familiar with the Workbench.
* Before starting the lab, make sure MySQL is running.
* Students can refer to the previous lab material (Lab 2a - Workbench Administration) for reference.
* Refer to the last page for different data types.

**Task 1**

1. Create a Schema named Hospital.
2. Create a table called ‘Patient’ that contains Patient\_ID, Patient\_Name, Patient\_Age, contact-number.
3. The model should save as .PatientDirectory\_Lab4’.
4. A live database server should be assigned and name it as ‘Hospital Server’.
5. Give inputs for the various columns.

|  |  |  |  |
| --- | --- | --- | --- |
| Patient |  |  |  |
| Patient\_ID | Patient\_Name | Patient\_Age | contact-number |
| 1 | Name1 | 12 | 12345 |
| 2 | Name2 | 21 | 123456 |
| 3 | Name3 | 12 | 46566 |
| 4 | Name4 | 12 | 655525 |
| 5 | Name5 | 34 | 85258 |
| 6 | Name1 | 25 | 955562 |
| 7 | Name7 | 25 | 255223 |

**Task 2**

1. What are the data types used for (2 pts)
   1. Patient\_ID ----
   2. Patient\_Name ----
   3. Patient\_Age ----
   4. contact-number ----
2. What is the query only to display (1 pts)
   1. Patient Name
   2. Patient ID
3. Apply the query- 3pts
   1. SELECT \* FROM hospital.patient

order by Patient\_Age;

What is your observation?

* 1. SELECT \* FROM hospital.patient

order by Contact\_number;

What is your observation?

* 1. SELECT \* FROM hospital.patient

order by Patient\_id;

What is your observation?

1. Apply the query- 4pts
   1. SELECT \* FROM hospital.patient

WHERE Patient\_Age='25';

What is your observation?

* 1. SELECT \* FROM hospital.patient

WHERE Patient\_Age='12';

What is your observation?

* 1. SELECT \* FROM hospital.patient

WHERE Patient\_Name='Name1';

What is your observation?

* 1. SELECT \* FROM hospital.patient

WHERE Patient\_Name='Name2';

What is your observation?

|  |  |
| --- | --- |
| **Data type** |  |
| **Data type** | **Description** |
| CHAR(size) | Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis. Can store up to 255 characters |
| VARCHAR(size) | Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters. Note: If you put a greater value than 255 it will be converted to a TEXT type |
| TINYTEXT | Holds a string with a maximum length of 255 characters |
| TEXT | Holds a string with a maximum length of 65,535 characters |
| BLOB | For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data |
| MEDIUMTEXT | Holds a string with a maximum length of 16,777,215 characters |
| MEDIUMBLOB | For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data |
| LONGTEXT | Holds a string with a maximum length of 4,294,967,295 characters |
| LONGBLOB | For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data |
|  |  |
| **Number Types** |  |
| **Data type** | **Description** |
| TINYINT(size) | -128 to 127 normal. 0 to 255 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| SMALLINT(size) | -32768 to 32767 normal. 0 to 65535 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| MEDIUMINT(size) | -8388608 to 8388607 normal. 0 to 16777215 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| INT(size) | -2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| BIGINT(size) | -9223372036854775808 to 9223372036854775807 normal. 0 to 18446744073709551615 UNSIGNED\*. The maximum number of digits may be specified in parenthesis |
| FLOAT(size,d) | A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |
| DOUBLE(size,d) | A large number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |
| DECIMAL(size,d) | A DOUBLE stored as a string , allowing for a fixed decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter |
| LONGBLOB | For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data |
|  | Let you enter a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. |
|  |  |
| **Date types** |  |
| **Data type** | **Description** |
| DATE() | A date. Format: YYYY-MM-DD |
|  |  |
|  | Note: The supported range is from '1000-01-01' to '9999-12-31' |
| DATETIME() | \*A date and time combination. Format: YYYY-MM-DD HH:MM:SS |
|  |  |
|  | Note: The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59' |
| TIMESTAMP() | \*A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD HH:MM:SS |
|  |  |
|  | Note: The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC |
| TIME() | A time. Format: HH:MM:SS |
|  |  |
|  | Note: The supported range is from '-838:59:59' to '838:59:59' |
| YEAR() | A year in two-digit or four-digit format. |
|  |  |
|  | Note: Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069 |
|  |  |
|  | \*Even if DATETIME and TIMESTAMP return the same format, they work very differently. In an INSERT or UPDATE query, the TIMESTAMP automatically set itself to the current date and time. TIMESTAMP also accepts various formats, like YYYYMMDDHHMMSS, YYMMDDHHMMSS, YYYYMMDD, or YYMMDD. |