**CSIT115 Spring 2025**

**Week3 Lab2 – Basic DDL and DML**

*Create a single table and insert, search, update and delete instances*

**Working with MySql workbench**

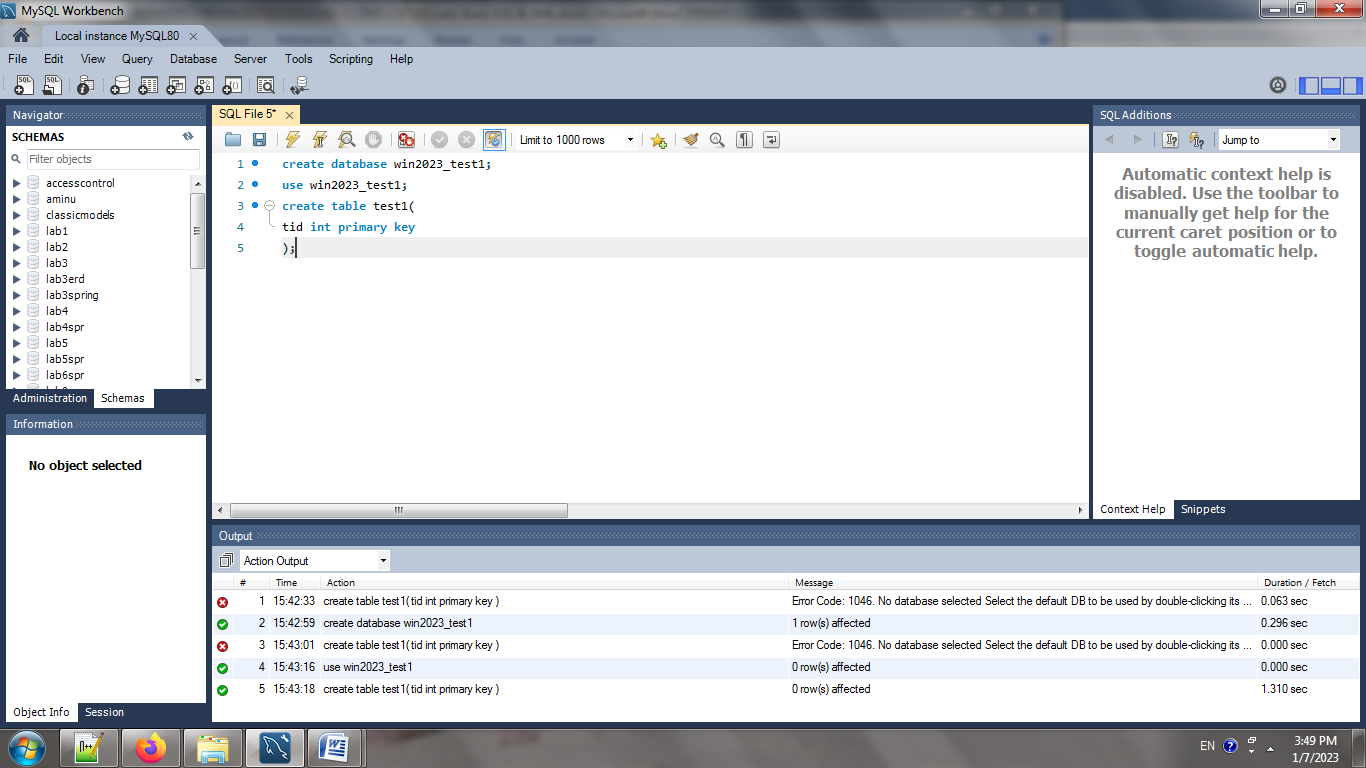
Step1: create connection

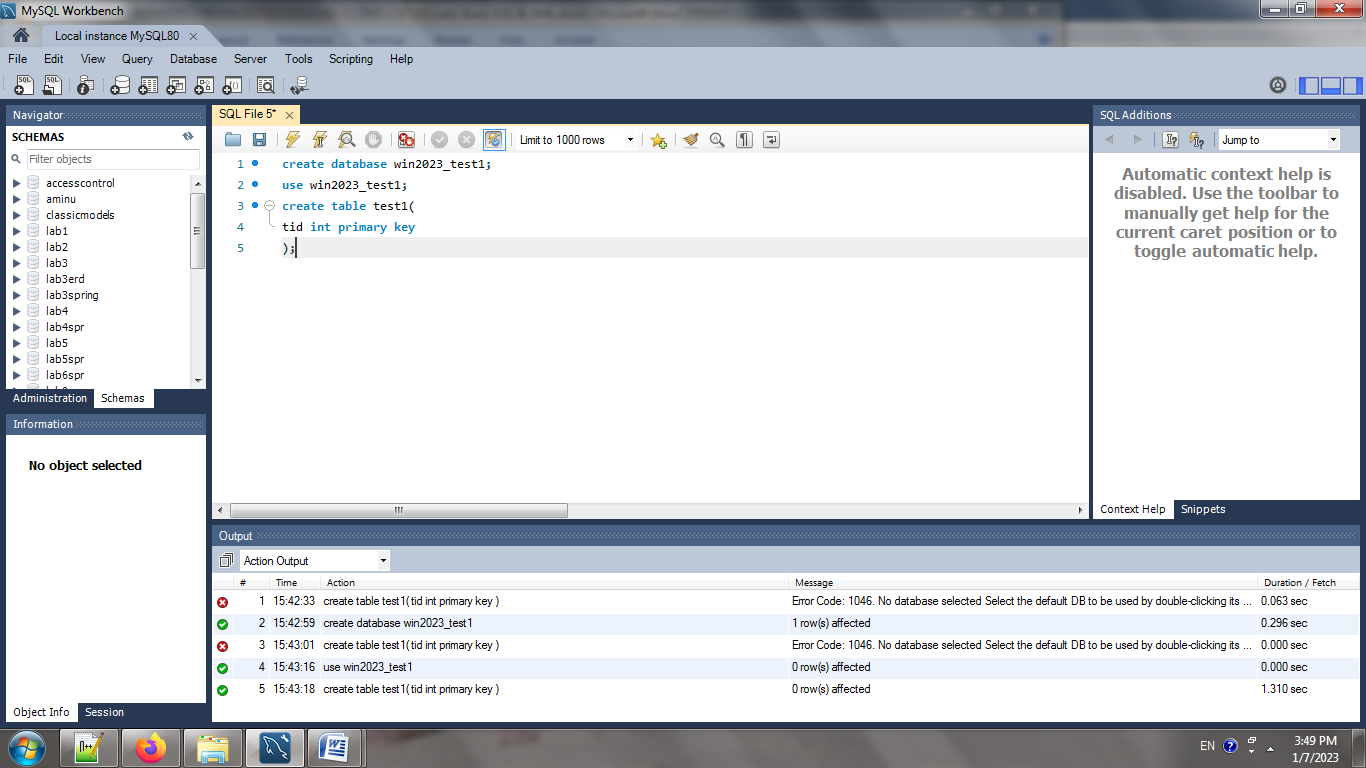
Connect to MySql Server using your username/password

If you are using lab terminals username and password are both **root**

Step2: create a database

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| CREATE DATABASE database\_name | CREATE DATABASE lab1 |

To run the code, select the statement and click on 

To run a single line that is pointed to by the cursor click on 

Step3: choose the database you are to run statements on

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| USE database\_name | USE lab1 |

Step4: ENJOY CODING!

**Data types**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Spec** | **Syntax** | **Example** |
| CHAR | String(0-255) allocates fix size memory | Var\_name CAHR(N) | Gender CHAR(1) |
| VARCHAR | String(0-255) allocates dynamic memory | Var\_name VARCAHR(N) | Stu\_name VARCHAR(50) |
| TEXT | String(0-65535) | Var\_name TEXT | stu\_recommendation TEXT |
| LONGTEXT | String(0-4294967295) | Var\_name LONGTEXT | Stu\_app\_summary LONGTEXT |
| INT | Integer | Var\_name INT | Stu\_id INT |
| FLOAT | Decimal, precise to 23 digits | Var\_name FLOAT | Grade FLOAT |
| DECIMAL | From 24 up to 53 digits | DECIMAL(digits,precision) | School\_GPA (4,2) |
| DATE | YYYY-MM-DD | Var\_name DATE | Date\_of\_birth DATE |
| DATETIME | YYYY-MM-DD HH:MM:SS | Var\_name DATETIME | Enroll\_date DATETIME |
| Year | YYYY | Var\_name YEAR | Year\_graduate YEAR |
| BOOLEAN | True/False | Var\_name BOOLEAN | Accepeted BOOLEAN |
| ENUM | Choose ONLY one of options | Var\_name ENUM (op1,op2,…,opn) | City ENUM ('Dubai', 'Abu Dhabi', 'Ajman', ' Sharjah') |
| SET | Choose one or more from options | Var\_name SET(op1,op2,…,opn) | Clubs SET ('Sport', 'Music', 'Book', 'Drama') |

**Basic Constraints on Columns**

|  |  |  |  |
| --- | --- | --- | --- |
| **Constraint** | **Explanation** | **Syntax** | **Example** |
| PRIMARY KEY | Uniquely identifies the row | Var\_name data\_type PRIMARY KEY | Stu\_id INT PRIMARY KEY |
| UNIQUE | Has a unique value. Candidate keys can become unique key | Var\_name data\_type UNIQUE | Stu\_email VARCHAR(50) UNIQUE |
| NOT NULL | Must have a value | Var\_name data\_type NOT NULL | Stu\_name VARCHAR(50) NOT NULL |
| DEFAULT | Setting a default value | Var\_name data\_type DEFAULT value | Stu\_gpa DECIMAL(4,2) DEFAULT 0 |

**BASIC Create table statement**

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| CREATE TABLE [IF NOT EXISTS] table\_name(  Column\_name data\_type [constraint],  Column\_name data\_type [constraint],  ...,  Column\_name data\_type [constraint],  [constraint column]  ) | CREATE TABLE students(  Stu\_id INT PRIMARY KEY,  Stu\_name VARCHAR(100) NOT NULL,  Stu\_gender CHAR(1),  Stu\_email VARCHAR(100) UNIQUE,  Stu\_gpa DECIMAL(4,2),  Stu\_dob DATE,  Stu\_city ENUM ('Dubai', 'Abu Dhabi', 'Ajman', 'Sahrjah'),  Unique(Stu\_dob, Stu\_name)  ); |

**INSERT Statement**

Before we insert. Just remember that you need to change string to date when you have DATE, or DATETIME variables. One option for casting string to date is

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| INSERT INTO table\_name VALUES(  Value1,value2,…) | INSERT INTO students VALUES  (123456,'Sara', 'Sara@gmail.com', 4.20,  '2006-12-09', 'Ajman') |

If some columns have default value or can be null, then you can write an insert statement for only the columns that you have value for by mentioning the column name after the table name

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| INSERT INTO table\_name(col1,col2,..) VALUES(  Value1,value2,…) | INSERT INTO students (Stu\_id,Stu\_name) VALUES(123456,'Asma'); |

**Searching/Updating/Deleting:**

Mysql by default is on safe mode which does not allow for delete or update statement, so you have to run the statement below to turn off the safe mode: (not needed when only running queries)

**SET SQL\_SAFE\_UPDATES = 0;**

**Basic Operators**

|  |  |
| --- | --- |
| **Type** | **Symbol** |
| **Logical** | AND, OR, IS NULL (IS NOT NULL) |
| **Equality check** | =, <>, != |
| **Arithmetic Comparison** | Between, >, >=, <, <= |
| **Arithmetic Operations** | +, -, \*, /, |

**Basic Aggregate Functions**

|  |  |
| --- | --- |
| **Function** | **Symbol** |
| **Count (column)** | Count number of rows |
| **SUM (column)** | sum of numerical value of a column |
| **MIN/MAX(column)** | Find minimum or maximum in a set of rows |
| **AVG(column)** | Find the average |
| **DISTINCT (column)** | Show only unique values |

**Searching: SELECT Statement**

|  |  |  |
| --- | --- | --- |
| **Explanation** | **Syntax** | **Example** |
| Show all columns and all rows in the table | SELECT \* FROM table\_name | SELECT \* FROM students; |
| Select specific column but all the rows from a table | SELECET colum\_name1,column\_name2 FROM Table\_NAME | SELECT stu\_id, stu\_name, stu\_email FROM students; |
| SELECT all columns but a specific rows | SELECT \* FROM table\_name WHERE column\_name=value | SELECT \* FROM students WHERE stu\_id=123456; |
| Using AND with conditions | SELECT \* FROM table\_name WHERE column\_name=value AND column\_name2=value2 | SELECT \* FROM students WHERE stu\_gpa>3.5 AND stu\_city='Abu Dhabi'; |
| Using OR with condition | SELECT \* FROM table\_name WHERE column\_name=value OR column\_name2=value2 | SELECT \* FROM students WHERE stu\_gpa<4 OR stu\_city='Ajman'; |
| Using between | SELECT \* FROM table\_name WHERE column\_name BETWEEN value1 and value2 | SELECT \* FROM students WHERE stu\_gpa between 3 and 4; |
| SORT the output | SELECT \* FROM table\_name ORDER BY column\_name [ASC|DESC] | SELECT \* FROM students ORDER BY stu\_gpa |
| Finding total number of rows | SELECT count(column) FROM table\_name | SELECT count(stu\_id) FROM students ; |
| Finding distinct values | SELECT distinct(column) FROM table\_name | SELECT distinct(stu\_gender) FROM students; |
| Get average of numerical column | SELECT avg(column) FROM table\_name | SELECT avg(stu\_gpa) FROM students ; |
| Add 0.25 to all students gpa |  | SELECT stu\_gpa+0.25 as new\_grade FROM students; |

**Basic DELETE Statement**

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| DELETE FROM table\_name WHERE conditions | Delete From students Where stu\_dob<'2005-05-24'; |

**Basic Update Statement**

|  |  |
| --- | --- |
| **Syntax** | **Example** |
| UPDATE table\_name SET col1=val1, col2=val2,…. WHERE conditions | Update Students Set stu\_gpa=stu\_gpa+0.1; |

**TASK0:** Create a database and use it!

**TASK1 (1.5):** Create a table Boxing\_Events based on below column specification

|  |  |
| --- | --- |
| **Column** | **Details** |
| Event id | A 4 digit number used to identify each event |
| country | Can be one of these countries only UAE, USA, UK, KSA |
| city | A string of size 50 |
| Venue name | A string of size 100, must always have value |
| Ticket price | Check instances for data type |
| Discount | Cannot be negative, if no data provided should be set to zero |
| Sold tickets | cannot be negative, if no data provided should be set to zero |
| Boxing date | Value must be provided all the time, make sure there are no events scheduled on the same day |
| boxer1 | String of size 100, must have value |
| boxer2 | String of size 100, must have value |
| Gender cat | Can be only 1 character |
| Weight scale | Number |
| Contact person | String of size 100 |
| Contact number | String or number |
| Contact email | String of size 50 |

**TASK2 (1.25):** insert below data (fix whatever needs to be fixed!)

1111, UAE, 'Dubai', 'Coca-Cola Arena', 135.85, 0 , 210, '2020-02-14', 'Ryad Ali','Max Millon', 'M',150,'Rudra',00971525678980,'rudra@boxing.com'

1112, 'UAE', ‘Dubai’, 'Coca-Cola Arena', 180.00, 5 , 440, '2023-09-09', 'Omar Atia','Owen Hank ', 'M',160,'Rudra',00971525678980,'rudra@boxing.com'

1113, 'UAE', 'Abu Dhabi', 'MBZ Sport Complex ', 222.22, 5 , 600, '2022-12-24', 'Areesha Hamad','Romana Shaikh', 'F',65,'Bilal',00971508989890,'bilal@boxing.com'

1114, 'UK', 'London', 'York Hall Leisure Centre', 79.99, 15 , 200, '2023-04-13', 'Eihab Malik','Michael Logan', 'M',150,'Miguel', 00448006893293,'miguel@boxing.com'

1115, 'UK', 'London', 'York Hall Leisure Centre', 129.99, 0 , 350, '2022-02-22', 'Max Millon', 'Michael Logan ', 'M',150,'Miguel', 00448006893293,'miguel@boxing.com'

1116,'USA', 'New York', 'World Boxing Hall ', 121.00, 0 , 400, '2023-02-23', 'Asma D','Amal A', 'F',180,'Elisa', 0016467050332,'elisa@boxing.com'

1116,'Iran', 'New York', NULL, 121.00, -10 , 400, '2023-02-23', 'Bilal','Mino', 'Male', '180','Elisa', 0016467050332,'elisa@boxing.com'

**TASK3:**

a)display all the boxing data

b)display all boxing data sorted by ticket price from highest to lowest

**TASK4:** show list of venues (no duplicates) as Boxing Venue, along with the country and city they are located in sort by country then city

**TASK5:** how many boxing events are happening in Dubai or any other city that is not in UAE (do not hardcode the city name!)

**TASK6:** show boxers and venue details for boxing events hosted in UAE with weight scale of 150kg

**TASK7:** show price, discounted price, boxers, contact person details of female boxing events which are for weight 150 or 65

**TASK8:**

a) show the average, maximum and minimum price of boxing events in a single statement (show proper column name)

b) show the total, average of sales amount (use discounted price)

**TASK9:** for all the boxing events with sold tickets less than 250 reduce the price by 15% and add 5% to discount

**TASK10:** delete all boxing events hosted within August 2022 to February 2023