|  |  |  |  |
| --- | --- | --- | --- |
| Title | DBMS project document | | |
| Author | 張家倫 | | |
| Version | 1.0 | Date | 2015/01/1 |

* **系統架構與環境及使用說明**

本系統以網頁的形式呈現database management system，前端網頁輸入指令、顯示資料，後端伺服器執行查詢指令，使用畫面截圖如figure 1。

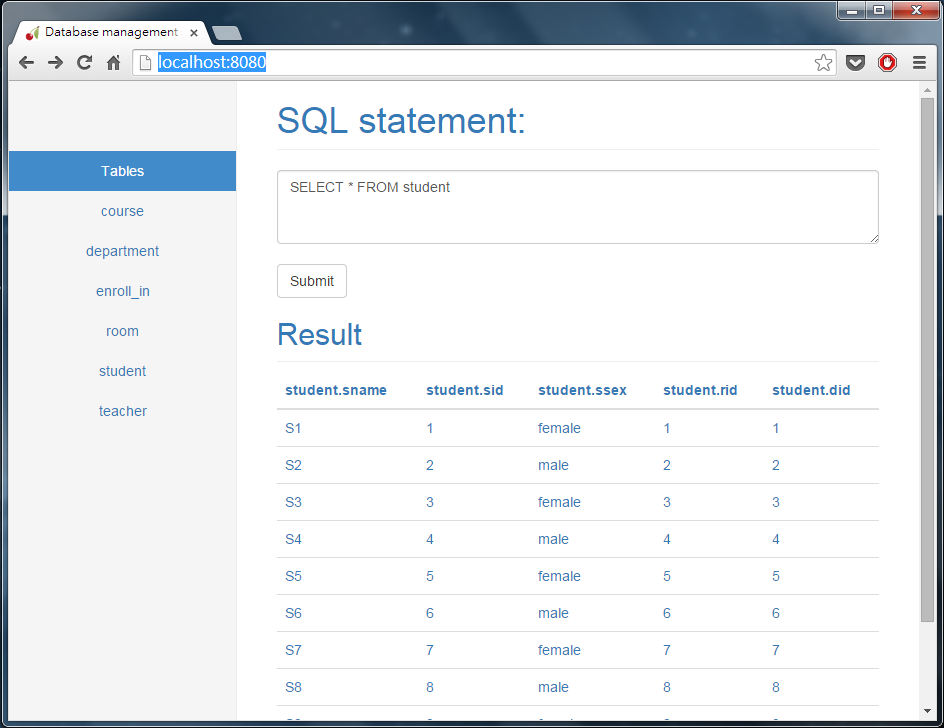


Figure 1. DBMS畫面截圖

畫面左方顯示了所有tables，使用者可以快速切換；使用者可以在畫面上方SQL statement部分輸入SQL查詢指令，詳細的SQL指另請參照”SQL指令說明”；SQL statement下則是呈現搜尋結果的區域。

**系統環境**

本系統運行在windows7, chrome瀏覽器下，後端以python 2.7語言撰寫，使用cherrypy[[1]](#footnote-1)做為web framework，務必在使用前安裝。

**使用說明**

使用前須先啟動伺服器環境，先找到 ***~/DBMS/webapp.py***，此檔案是伺服器的主程式，開啟windows command line (當前資料夾按下Shift+滑鼠右鍵，選擇在此處開啟命令視窗)，輸入指令:

**python webapp.py**

成功執行時出現訊息如figure 3。伺服器成公開啟後，打開瀏覽器chrome，輸入網址: **localhost:8080**

即可開始使用本系統。

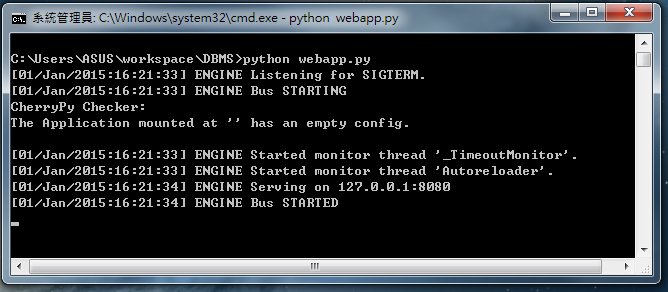


Figure 3. 命令視窗開啟伺服器

* **Database design**

本資料庫系統目標是模擬成大學生選課系統[[2]](#footnote-2)，包含了學生、課程、老師、系所、教室等資訊，以研究所課程為主(當中又以成大電通所為資料來源)，大部分的資料可以在選課系統中取得，像是課程代碼、授課教室、授課老師等等。除了選課的資料，本系統建立了研究生、實驗室、系所的關係，如學生屬於甚麼實驗室、系所擁有哪些實驗室等資訊，ER diagram如下圖figure 1。

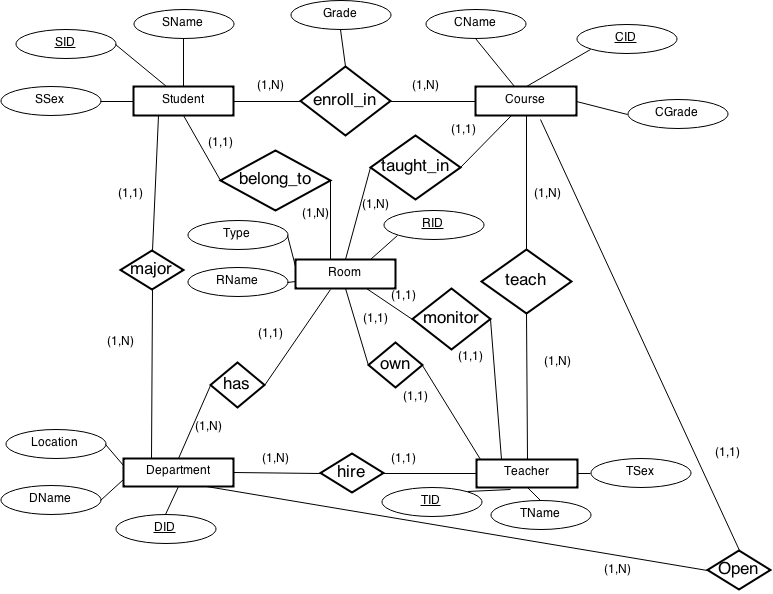


Figure. 1 ER diagram of NCKU course system

以下分別就ER diagram entity design, relation design作說明:

**Entity design**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Attribute** | **Description** |
| Student | SID | 學生學號，ex:Q36034138 |
| SName | 學生姓名，ex:張家倫 |
| SSex | 學生性別，ex:男 |
| Course | CID | 課程序號，ex:001 |
| CName | 課程名稱，ex:專題討論(二) |
| CGrade | 課程班別，ex:碩博 |
| Teacher | TID | 老師教師證號，ex:T001 |
| TName | 老師姓名，ex:李強 |
| TSex | 學生性別，ex:男 |
| Department | DID | 系所代號，ex:Q3(電通所系號) |
| DName | 系所名稱，ex:電通所 |
| Location | 系所校區，ex:自強校區 |
| Room | RID | Room ID(教室代號、辦公室代號、實驗室代號)，ex:電機系繁城講堂的代號為92127 |
| RName | Room名稱，ex:繁城講堂，如果沒有特別命名則與RID相同 |
| Type | Room的種類，ex:教室、實驗室、辦公室 |

**Relation design**

|  |  |  |
| --- | --- | --- |
| **Relation** | **Involved entities** | **Description** |
| enroll\_in | Student, Course | 學生修課，並有一分數 |
| taught\_in | Course, Room | 課程在哪間教室上課 |
| belong\_to | Studebt, Room | 學生屬於哪間實驗室 |
| teach | Teacher, Course | 老師教哪一堂課 |
| monitor | Teacher, Room | 老師管理哪間實驗室(Room’s type為實驗室) |
| own | Teacher, Room | 老師擁有一間辦公室(Room’s type為辦公室) |
| has | Department, Room | 系所有許多Room |
| major | Student, Department | 學生主修哪一系 |
| hire | Department, Teacher | 系所雇用老師 |

確認完entity design and relation design之後，需要將ER diagram轉換成SQL tables，以下分別就table design作說明:

**Table design**

Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SID | SName | SSex | DID | RID |

Course

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CID | CName | CGrade | DID | RID | TID |

Teacher

|  |  |  |  |
| --- | --- | --- | --- |
| TID | TName | TSex | DID |

Department

|  |  |  |
| --- | --- | --- |
| DID | DName | Location |

Room

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RID | RName | Type | DID | OwnerID |

enroll\_in

|  |  |  |
| --- | --- | --- |
| SID | CID | Grade |

|  |  |
| --- | --- |
| Table | Description |
| Student | 學生有學號(SID)姓名(SName)、性別(SSex)，所屬系所(DID)及實驗室(RID)。 |
| Course | 課程有序號(CID)、名稱(CName)、班別(CGrade)，開課系所(DID)、上課地點(RID)、授課老師(TID)。 |
| Teacher | 老師有教師證號(TID)、姓名(TName)、性別(TSex)，雇傭系所(DID)。 |
| Department | 系所有代號(DID)、名稱(DName)、位置(Location)。 |
| Room | Room有代號(RID)、名稱(RName)、種類(Type)，Room屬於某系所(DID)，Room的種類有三: 教室、辦公室、實驗室，三種Room都有一位老師管理(OwnerID)，。 |
| enroll\_in | 學生(SID)修習課程(CID)，並有一分數(Grade)。 |

* **SQL指令說明**

**Basic queries**

**SELECT** <attribute list> **FROM** <table list> **WHERE** <condition>

SELECT, FROM, WHERE請大寫，其他attribute, table, condition均使用小寫，多項attributes, tables請用”,”隔開，多項conditions請用AND隔開(目前不支援OR)，例如:

Q1: SELECT tname, tsex FROM teacher WHERE tid = 1 AND tsex = male

但如遇到兩個tables都有同名的attribute，則須在attribute加上table名稱，例如:

Q2: SELECT \* FROM teacher, department WHERE teacher.did = department.did

**Complex queries (IN, NOT IN, Aggregation functions)**

**SELECT** <attribute list> **FROM** <table list> **WHERE** <condition> **IN**

**( SELECT** <attribute list> **FROM** <table list> **WHERE** <condition> **)**

使用到**IN, NOT IN**時**，**後面的query請用括號括起，括號內query規則同basic query，例如:

Q3: SELECT \* FROM teacher WHERE did IN ( SELECT did FROM department WHERE location = L3)

使用aggregation functions (COUNT, MAX, MIN, SUM, AVG)時請將aggregation functions大寫，並以”,”隔開，例如:

Q4: SELECT COUNT(\*) FROM student

Q5: SELECT MAX(grade), MIN(grade), AVG(grade) FROM enroll\_in

* **Project file structure**

DBMS/

├── darren/ // main source package

│ ├── \_\_init\_\_.py

│ ├── datagenerator.py // generate data automatically

│ ├── db.py // db functions, e.g. join, load data

│ ├── main.py // main logic to execute a query

│ ├── sqlparser.py // sql query parser

├── data/… // table data saved in this directory

├── dist/… // bootstrap framework, js, css

└── index.html

└── webapp.py // server code

Table資料說明:

Project一開始希望將選課系統上的資料擷取下來，當作本系統的資料來源，不過由於python中文編碼的問題，用UTF-8存成中文時.txt file的資料不是人能看懂得(因為是ㄧ堆編碼)，unicode編碼雖可看懂，卻不支援中文編碼，所以採取自動產生英文的資料以便初期開發，未來可以延伸此系統支援中文資料。

* **Demo queries**

Q1: SELECT tname, tsex FROM teacher WHERE tid = 1

Q2: SELECT \* FROM teacher, department WHERE teacher.did = department.did AND location = 3

Q3: SELECT \* FROM teacher WHERE did IN ( SELECT did FROM department WHERE location = L3)

Q4: SELECT \* FROM teacher WHERE did NOT IN ( SELECT did FROM department WHERE location = L3)

Q5: SELECT COUNT(\*) FROM enroll\_in

Q6: SELECT MAX(grade), MIN(grade), AVG(grade) FROM enroll\_in

1. Cherrypy, <http://www.cherrypy.org/> [↑](#footnote-ref-1)
2. 國立成功大學課程查詢系統, <http://course-query.acad.ncku.edu.tw/qry/> [↑](#footnote-ref-2)