Daniel Patrick(CS), Lauren Vaughn(CS)

Assignment 3

15 November 2017

**Physical Database Design**

Assignment four is to develop a physical database design for the MUHousing Problem. To complete this task, the following steps are followed:

1. Determine data types for all attributes.
2. Determine index attributes.
3. Calculate overall database size.

In order to implement the MUHousing database design, the latest SQL server will be used. To begin, the MUHousing relational data model (RDM) is displayed in figure 1.

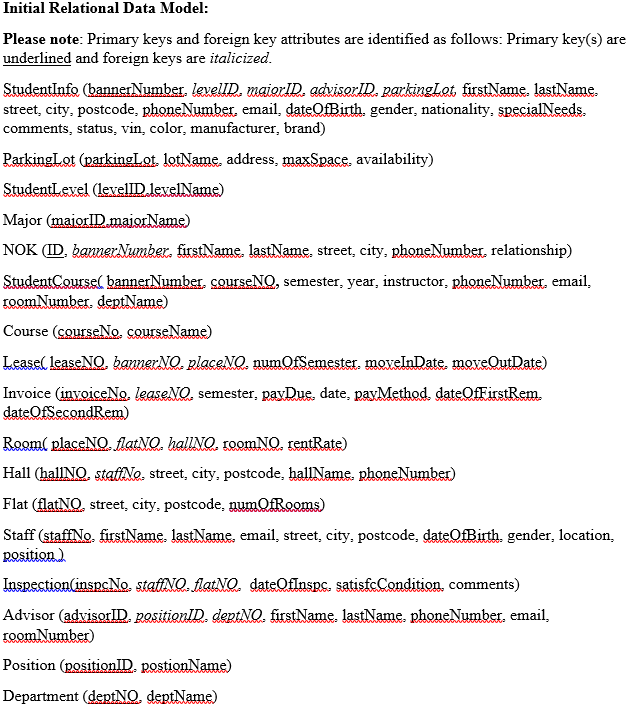


Figure : MUHousing Relational Data Model

**Index Determination**

Each index requires 40% disk-space overhead, but also enables a faster query response. So, there is a trade-off between less disk space and faster response. Any attribute that is to be used in a conditional expression of a query is to be indexed.

Following the “when to index” rules and the users’ search requirements (queries), it is proposed to index the tables on the following attributes:

1. firstName
2. lastName
3. street
4. city
5. postCode
6. phoneNumber
7. email
8. dateOfBirth
9. gender
10. nationality
11. specialNeeds
12. comments
13. status
14. vin
15. lotName
16. levelName
17. numOfSemester
18. moveInDate
19. moveOutDate
20. payDue
21. date
22. payMethod
23. roomNO
24. rentRate
25. hallName
26. position
27. dateOfInspc
28. satisfcCondition

Please note that both primary and foreign keys are always indexed.

Using the relational data model, the SQL terminology is used to define the attributes, indexes, and security requirements for each attribute.

**STUDENTINFO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| bannerNumber | Integer(9 bytes) | No | Yes(Primary) | N/A |
| levelID | Integer (4 bytes) | No | Yes(Foreign) | N/A |
| majorID | Integer (4 bytes) | No | Yes(Foreign) | N/A |
| advisorID | Integer (9 bytes) | No | Yes(Foreign) | N/A |
| parkinglot | Integer (1 byte) | No | Yes(Foreign) | N/A |
| firstName | VarChar(20) | No | Yes | N/A |
| lastName | VarChar(20) | No | Yes | N/A |
| street | VarChar(30) | No | Yes | N/A |
| city | VarChar(15 bytes) | No | Yes | N/A |
| postCode | Integer (5 bytes) | No | Yes | N/A |
| phoneNumber | Char(10 bytes) | No | Yes | N/A |
| email | VarChar(25) | No | Yes | N/A |
| dateOfBirth | Date (10 bytes) | No | Yes | N/A |
| gender | Char(1 bytes) | No | Yes | N/A |
| nationality | VarChar(11 bytes) | No | Yes | N/A |
| specialNeeds | VarChar(3 bytes) | No | Yes | N/A |
| comments | VarChar( 50 bytes) | No | Yes | N/A |
| status | VarChar(7 bytes) | No | Yes | N/A |
| vin | Char(17 bytes) | No | Yes | N/A |
| color | VarChar(6 bytes) | Yes | No | N/A |
| manufacturer | VarChar(10 bytes) | Yes | No | N/A |
| brand | VarChar (15 bytes) | Yes | No | N/A |

**PARKINGLOT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| parkingLot | Integer (1 byte) | No | Yes(Primary) | N/A |
| lotName | VarChar(10 bytes) | No | Yes | N/A |
| address | VarChar(20 bytes) | Yes | No | N/A |
| maxSpace | Integer (3 bytes) | Yes | No | N/A |
| availability | VarChar (3 bytes) | Yes | No | N/A |

**STUDENTLEVEL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| levelID | Integer (4 bytes) | No | Yes(Primary) | N/A |
| levelName | VarChar(10) | No | Yes | N/A |

**MAJOR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| majorID | Integer (4 bytes) | No | Yes(Primary) | N/A |
| majorName | VarChar(20) | Yes | No | N/A |

**NOK**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| ID | Integer (9 bytes) | No | Yes(Primary) | N/A |
| bannerNumber | Integer (9 bytes) | No | Yes(Foreign) | N/A |
| firstName | VarChar(20) | Yes | No | N/A |
| lastName | VarChar(20) | Yes | No | N/A |
| street | VarChar(30 bytes) | Yes | No | N/A |
| city | VarChar(15 bytes) | Yes | No | N/A |
| phoneNumber | Char(10 bytes) | Yes | No | N/A |
| Relationship | VarChar(12 bytes) | Yes | No | N/A |

**STUDENTCOURSE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| bannerNumber | Integer (9 bytes) | No | Yes (Primary), Foreign | N/A |
| courseNO | Integer (4 bytes) | No | Yes (Primary), Foreign | N/A |
| semester | VarChar(5 bytes) | Yes | No | N/A |
| year | Integer (4 bytes) | Yes | No | N/A |
| instructor | VarChar(40 bytes) | Yes | No | N/A |
| phoneNumber | Char(10 bytes) | Yes | No | N/A |
| email | VarChar(25 bytes) | Yes | No | N/A |
| roomNumber | VarChar(4 bytes) | Yes | No | N/A |
| deptName | VarChar(20 bytes) | Yes | No | N/A |

**COURSE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| courseNo | Integer (4 bytes) | No | Yes (Primary) | N/A |
| courseName | VarChar (25 bytes) | Yes | No | N/A |

**LEASE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| leaseNO | Integer(5 bytes) | No | Yes (Primary) | N/A |
| bannerNumber | Integer (9 bytes) | No | Yes (Foreign) | N/A |
| placeNO | Integer (4 bytes) | No | Yes (Foreign) | N/A |
| numOfSemester | Integer (1 byte) | No | Yes | N/A |
| moveInDate | Date (10 bytes) | No | Yes | N/A |
| moveOutDate | Date (10 bytes) | No | Yes | N/A |

**INVOICE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| invoiceNO | Integer(5 bytes) | No | Yes (Primary) | N/A |
| leaseNO | Integer(5 bytes) | No | Yes (Foreign) | N/A |
| semester | VarChar(5 bytes) | Yes | No | N/A |
| payDue | Date (10 Bytes) | No | Yes | N/A |
| date | Date (10 bytes) | No | Yes | N/A |
| payMethod | VarChar(11 bytes) | No | Yes | encrypted |
| dateOfFirstRem | Date (10 bytes) | Yes | No | N/A |
| dateOfSecondRem | Date (10 bytes) | Yes | No | N/A |

**ROOM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| placeNO | Integer (4 bytes) | No | Yes (Primary) | N/A |
| flatNO | Integer (3 bytes) | No | Yes (Foreign) | N/A |
| hallNO | Integer (1 byte) | No | Yes (Foreign) | N/A |
| roomNO | Integer (3 bytes) | No | Yes | N/A |
| rentRate | Money (7 bytes) | No | Yes | N/A |

**HALL**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| hallNO | Integer (1 byte) | No | Yes (Primary) | N/A |
| staffNO | Integer (9 bytes) | No | Yes (Foreign) | N/A |
| Street | VarChar(30 bytes) | Yes | No | N/A |
| City | VarChar(15 bytes) | Yes | No | N/A |
| postCode | Integer (5 bytes) | Yes | No | N/A |
| hallName | VarChar(20 bytes) | No | Yes | N/A |
| phoneNumber | Char(10 bytes) | No | Yes | N/A |

**FLAT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| flatNO | Integer (3 bytes) | No | Yes (Primary) | N/A |
| street | VarChar(30 bytes) | Yes | No | N/A |
| city | VarChar(15 bytes) | Yes | No | N/A |
| postcode | Integer (5 bytes) | Yes | No | N/A |
| numOfRooms | Char (1 byte) | Yes | No | N/A |

**STAFF**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| staffNO | Integer (9 bytes) | No | Yes (Primary) | N/A |
| firstName | VarChar(20) | No | Yes | N/A |
| lastName | VarChar(20) | No | Yes | N/A |
| email | VarChar(25 bytes) | No | Yes | N/A |
| street | VarChar(30 bytes) | No | Yes | N/A |
| city | VarChar(15 bytes) | No | Yes | N/A |
| postcode | Integer (5 bytes) | No | Yes | N/A |
| dateOfBirth | Date (10 bytes) | No | Yes | N/A |
| Gender | Char(1 byte) | Yes | No | N/A |
| location | VarChar( 13 bytes) | Yes | No | N/A |
| position | VarChar (15 bytes) | No | Yes | N/A |

**INSPECTION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| inspcNo | Integer (5 bytes) | No | Yes (Primary) | N/A |
| staffNO | Integer (9 bytes) | No | Yes (Foreign) | N/A |
| flatNO | Integer (3 bytes) | No | Yes (Foreign) | N/A |
| dateOfInspc | Date (10 bytes) | No | Yes | N/A |
| satisfcCondition | VarChar(3 bytes) | No | Yes | N/A |
| comments | VarChar( 50 Bytes) | No | Yes | N/A |

**ADVISOR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| advisorID | Integer (9 bytes) | No | Yes (Primary) | N/A |
| positionID | Integer (5 bytes) | No | Yes (Foreign) | N/A |
| deptNO | Integer (5 bytes) | No | Yes (Foreign) | N/A |
| firstName | VarChar(20) | No | Yes | N/A |
| lastName | VarChar(20) | No | Yes | N/A |
| phoneNumber | Char(10 bytes) | No | Yes | N/A |
| email | VarChar(25 bytes) | Yes | No | N/A |
| roomNumber | VarChar (4 bytes) | Yes | No | N/A |

**POSITION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| positionID | Integer (5 bytes) | No | Yes (Primary) | N/A |
| positionName | VarChar(20 bytes) | Yes | No | N/A |

**DEPARTMENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Data Type** | **Null(Yes, No)** | **Index** | **Security** |
| deptNO | Integer (5 bytes) | No | Yes (Primary) | N/A |
| deptName | VarChar(20 bytes) | Yes | No | N/A |

**Database Size (Disk Space Requirement)**

It is necessary to estimate the total database size based upon table sizes and volumes. Using the following guidelines, the database size was computed in bytes.

1. Compute basic size by adding attribute sizes and multiplying by the table volumes determined by the integrity constraints.
2. Add 40% for each indexed attribute (including the primary and foreign keys).
3. Add up all of the tables to compute the database size.
4. Multiply the database size by 1.5 in order to account for unexpected future growth.

The calculations are carried out in the four tables (figure 2) below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | StudentInfo | ParkingLot | StudentLevel | Major | NOK |
| Basic Size of One Row | 282 | 37 | 14 | 24 | 120 |
| Basic Size of a Table | 282\*6,500=  1,833,000 | 37\*5=185 | 14\*6=84 | 24\*100=2400 | 120\*6,500=  780,000 |
| Index Overhead | 7.6\*2808000=  13,930,800 | 0.8\*185  =148 | 0.8\*84  =67.2 | 0.4\*2400  =960 | 0.8\*780,000  =624,000 |
| Total for Each Table | 15,763,800 | 333 | 151.2 | 3360 | 1,404,000 |
| Database Size |  |  |  |  | 17,171,645 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | StudentCourse | Course | Lease | Invoice | Room |
| Basic Size of One Row | 121 | 29 | 39 | 66 | 18 |
| Basic Size of a Table | 121\*6500=  786,500 | 29\*500  =14,500 | 39\*25,000  =975,000 | 66\*25,000  =1,650,000 | 18\*6,000  =108,000 |
| Index Overhead | 0.8\*786,500=  629,200 | 0.4\*14,500  =5,800 | 2.4\*390,000  =2,340,000 | 2.0\*660,000  =3,300,000 | 2.0\*108,000  =216,000 |
| Total for Each Table | 1,415,700 | 20,300 | 3,315,000 | 4,950,000 | 324,000 |
| Database Size |  |  |  |  | 10,025,000 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Hall | Flat | Staff | Inspection | Advisor |
| Basic Size of One Row | 90 | 54 | 135 | 80 | 98 |
| Basic Size of a Table | 90\*5=  450 | 54\*300  =16,200 | 135\*300  =40,500 | 80\*1000  =8,000 | 98\*200  =19,600 |
| Index Overhead | 1.6\*450=  720 | 0.4\*16,200  =6,480 | 3.6\*40,500  =145,800 | 2.4\*8,000  =19,200 | 2.4\*19,600  =47,040 |
| Total for Each Table | 1,170 | 22,680 | 186,300 | 27,200 | 66,640 |
| Database Size |  |  |  |  | 303,990 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Position | Department |  |
| Basic Size of One Row | 25 | 25 |  |
| Basic Size of a Table | 25\*10=  250 | 25\*25  =625 |  |
| Index Overhead | 0.4\*250=  100 | 0.4\*625  =250 |  |
| Total for Each Table | 350 | 875 | 27,501,860  OVERALL TOTAL |
| Database Size |  |  | 27,501,860\*1.5  =41,252,790 |

*Figure 2: Database size calculations for MUHousing Database (four tables above).*

The estimated database size is approximately 42MB (million bytes).