

B. Student Database Schema

Table and Column Descriptions

This appendix lists all the tables from the STUDENT database schema that are used in the book. Each table is listed here with the database table name first; what follows are the columns in that table, an indicator of whether a null value is allowed, the column data type, and a description. The scripts to install this database can be found on the companion website for this book.

Column Name	Null	Type	Comments
COURSE_NO	NOT NULL	NUMBER (8, 0)	The unique course number
DESCRIPTION	NULL	VARCHAR2 (50)	The full name for this course
COST	NULL	NUMBER (9, 2)	The dollar amount charged for enrollment in this course
PREREQUISITE	NULL	NUMBER (8, 0)	The ID number of the course that must be taken as a prerequisite to this course
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

COURSE: Information for a Course

Column Name	Null	Type	Comments
SECTION_ID	NOT NULL	NUMBER (8, 0)	The unique ID for a section
COURSE_NO	NOT NULL	NUMBER (8, 0)	The course number for which this is a section
SECTION_NO	NOT NULL	NUMBER (3)	The individual section number within this course
START_DATE_TIME	NULL	DATE	The date and time at which this section meets
LOCATION	NULL	VARCHAR2 (50)	The meeting room for the section
INSTRUCTOR_ID	NOT NULL	NUMBER (8, 0)	The ID number of the instructor who teaches this section
CAPACITY	NULL	NUMBER (3, 0)	The maximum number of students allowed in this section
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

SECTION: Information for an Individual Section (Class) of a Particular Course

Column Name	Null	Type	Comments
STUDENT_ID	NOT NULL	NUMBER (8, 0)	The unique ID for a student
SALUTATION	NULL	VARCHAR2 (5)	This student's title (e.g., Ms., Mr., Dr.)
FIRST_NAME	NULL	VARCHAR2 (25)	This student's first name
LAST_NAME	NOT NULL	VARCHAR2 (25)	This student's last name
STREET_ADDRESS	NULL	VARCHAR2 (50)	This student's street address
ZIP	NOT NULL	VARCHAR2 (5)	The postal ZIP code for this student
PHONE	NULL	VARCHAR2 (15)	The phone number for this student, including area code
EMPLOYER	NULL	VARCHAR2 (50)	The name of the company where this student is employed
REGISTRATION_DATE	NOT NULL	DATE	The date this student registered in the program
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

STUDENT: Profile Information for a Student

Column Name	Null	Type	Comments
STUDENT_ID	NOT NULL	NUMBER (8, 0)	The ID for a student
SECTION_ID	NOT NULL	NUMBER (8, 0)	The ID for a section
ENROLL_DATE	NOT NULL	DATE	The date this student registered for this section
FINAL_GRADE	NULL	NUMBER (3, 0)	The final grade given to this student for all work in this section (class)
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

ENROLLMENT: Information for a Student Registered for a Particular Section of a Particular Course (Class)

Column Name	Null	Type	Comments
INSTRUCTOR_ID	NOT NULL	NUMBER (8)	The unique ID for an instructor
SALUTATION	NULL	VARCHAR2 (5)	This instructor's title (e.g., Mr., Ms., Dr., Rev.)
FIRST_NAME	NULL	VARCHAR2 (25)	This instructor's first name
LAST_NAME	NULL	VARCHAR2 (25)	This instructor's last name
STREET_ADDRESS	NULL	VARCHAR2 (50)	This instructor's street address
ZIP	NULL	VARCHAR2 (5)	The postal ZIP code for this instructor
PHONE	NULL	VARCHAR2 (15)	The phone number for this instructor, including area code
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

INSTRUCTOR: Profile Information for an Instructor

Column Name	Null	Type	Comments
ZIP	NOT NULL	VARCHAR2 (5)	The ZIP code number, unique for a city and state
CITY	NULL	VARCHAR2 (25)	The city name for this ZIP code
STATE	NULL	VARCHAR2 (2)	The postal abbreviation for the U.S. state
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

ZIPCODE: City, State, and ZIP Code Information

Column Name	Null	Type	Comments
GRADE_TYPE_CODE	NOT NULL	CHAR (2)	The unique code that identifies a category of grade (e.g., MT, HW)
DESCRIPTION	NOT NULL	VARCHAR2 (50)	The description for this code (e.g., Midterm, Homework)
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

GRADE_TYPE: Lookup Table of a Grade Type (Code) and its Description

Column Name	Null	Type	Comments
SECTION_ID	NOT NULL	NUMBER (8)	The ID for a section
GRADE_TYPE_CODE	NOT NULL	CHAR (2)	The code that identifies a category of grade
NUMBER_PER_SECTION	NOT NULL	NUMBER (3)	How many of these grade types can be used in this section (i.e., there may be three quizzes)
PERCENT_OF_FINAL_GRADE	NOT NULL	NUMBER (3)	The percentage this category of grade contributes to the final grade
DROP_LOWEST	NOT NULL	CHAR (1)	Is the lowest grade in this type removed when determining the final grade? (Y/N)
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

GRADE_TYPE_WEIGHT: Information on How the Final Grade for a Particular Section is Computed; For Example, the Midterm Constitutes 50%, the Quiz 10%, and the Final Examination 40% of the Final Grade

Column Name	Null	Type	Comments
STUDENT_ID	NOT NULL	NUMBER (8)	The ID for a student
SECTION_ID	NOT NULL	NUMBER (8)	The ID for a section
GRADE_TYPE_CODE	NOT NULL	CHAR (2)	The code that identifies a category of grade
GRADE_CODE_OCCURRENCE	NOT NULL	NUMBER (38)	The sequence number of one grade type for one section. For example, there could be multiple assignments numbered 1, 2, 3, etc.
NUMERIC_GRADE	NOT NULL	NUMBER (3)	Numeric grade value (e.g., 70, 75)
COMMENTS	NULL	VARCHAR2 (2000)	Instructor's comments on this grade
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

GRADE: The Individual Grades a Student Received for a Particular Section (Class)

Column Name	Null	Type	Comments
LETTER_GRADE	NOT NULL	VARCHAR (2)	The unique grade as a letter (A, A-, B, B+, etc.)
GRADE_POINT	NOT NULL	NUMBER (3, 2)	The number grade on a scale from 0 (F) to 4 (A)
MAX_GRADE	NOT NULL	NUMBER (3)	The highest grade number that corresponds to this letter grade
MIN_GRADE	NOT NULL	NUMBER (3)	The lowest grade number that corresponds to this letter grade
CREATED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates the user who inserted data
CREATED_DATE	NOT NULL	DATE	Audit column—indicates the date of data insertion
MODIFIED_BY	NOT NULL	VARCHAR2 (30)	Audit column—indicates who made the last update
MODIFIED_DATE	NOT NULL	DATE	Audit column—date of last update

GRADE_CONVERSION: Converts a Number Grade to a Letter Grade

The entity-relationship diagram for the tables in the student schema shown in *Figure B-1* shows the tables and their foreign key relationships using the standard crow's feet arrows.

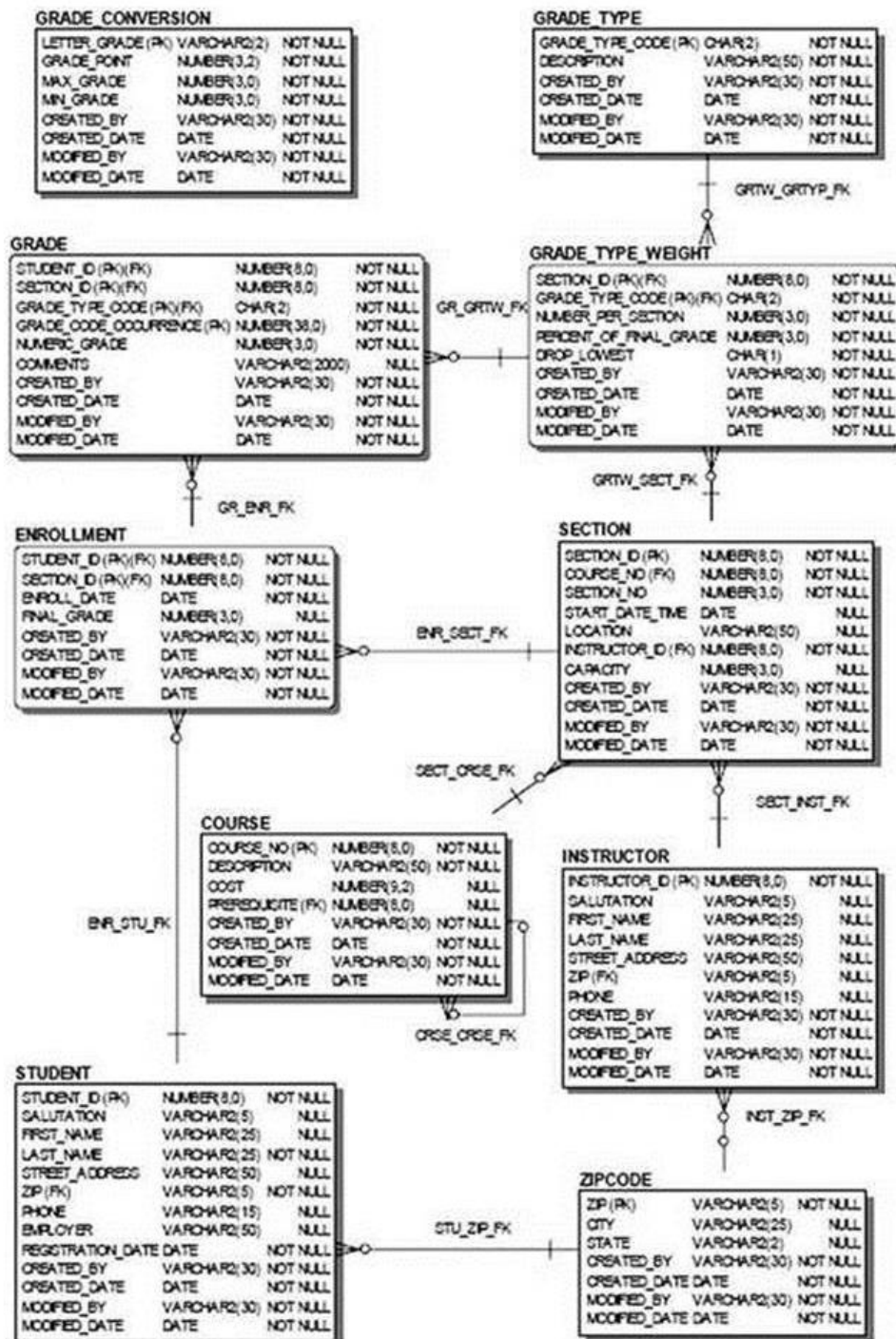


Figure B-1

The Student Schema Diagram

Lab Objectives

After this lab, you will be able to:

- Understand the STUDENT Schema Diagram and Identify Table Relationships

Throughout this series of Oracle Interactive Workbooks, the database for a school's computer education program is used as a case study upon which all exercises are based. If you have worked through the previous two labs, you know that the schema diagram is a model of data that reflects the relationships among data in a database. The name of the case study schema diagram is Student. Before you begin to write SQL statements against the database, it is important to familiarize yourself with the diagram. You can find this graphical representation in Appendix D, "Student Database Schema."

NOTE

In this book you will be frequently referring to the STUDENT schema diagram shown in Appendix D, "STUDENT Database Schema." Rather than flipping back and forth, you may find it more convenient to print out the schema diagram from the companion Web site of this book located at <http://www.phptr.com/rischert>.

THE STUDENT TABLE

Examine the STUDENT schema diagram and locate the STUDENT table. This table contains data about individual students, such as their name, address, employer, and date they registered in the program.

DataTYPES

Next to each column name in the diagram you find the datatype of the column. Each column contains a different kind of data, which can be classified by a datatype. You will notice that the FIRST_NAME column is of datatype VARCHAR2(25). This means that a variable length of (with a maximum of 25) alphanumeric characters (letters or numbers) may be stored in this column. Another datatype, the CHAR datatype, also stores alphanumeric data, but is a fixed-length datatype and pads any unused space in the column with blanks until it reaches the defined column length. The STUDENT_ID column is of datatype NUMBER with a maximum number of eight integer digits and no decimal place digits; the column is the primary key as denoted with the "(PK)" symbol. Oracle also provides a DATE datatype (as seen on the CREATED_DATE and MODIFIED_DATE columns) that stores both the date and time. You will learn more about the various datatypes in the next chapter.

Next to each column, the schema diagram indicates if a column allows NULL values. A NULL value is an unknown value. A space or value of zero is not the same as NULL. When a column in a row is defined as

allowing NULL values, it means that a column does not need to contain a value. When a column is defined as NOT NULL it must always contain a value.

You will observe that the STUDENT table does not show the city and state. This information can be looked up via the foreign key column ZIP as indicated with the "(FK)" symbol after the column name. The ZIP column is a NOT NULL column and requires that every student row have a corresponding zip code entered.

The COURSE TABLE

The COURSE table lists all the available courses that a student may take. The primary key of the table is the COURSE_NO column. The DESCRIPTION column shows the course description and the COST column lists the dollar amount charged for the enrollment in the course. The PREREQUISITE column displays the course number, which must be taken as a prerequisite to this course. This column is a foreign key column and its values refer back the COURSE_NO column. Only valid COURSE_NO values may be listed in this column. The relationship line of the COURSE table to itself represents a *recursive* or *self-referencing relationship*.

RECURSIVE RELATIONSHIP

As the term recursive or self-referencing relationship implies, a column in the course table refers back to another column in the same table. The prerequisite column refers back to the course_no column, which provides the list of acceptable values (also referred to as a *domain*) for the prerequisite column. Because the relationship is optional, the foreign key column PREREQUISITE column allows null. Recursive relationships are always optional relationships; otherwise, there is no starting point in the hierarchy.

[Figure 1.32](#) lists an excerpt of data from the COURSE table. Notice that the courses with the COURSE_NO column values of 10 and 20 do not have a value in the PREREQUISITE column, those are the courses which a student must take to be able to take any subsequent courses (unless equivalent experience can be substituted). Course number 20 is a prerequisite course for course number 100, Hands-On-Windows, and course number 140, Structured Analysis. You will explore more about the intricacies of recursive relationships in Chapter 15, "Advanced SQL Queries."

COURSE_NO	DESCRIPTION	PREREQUISITE	...
10	DP Overview		...
20	Intro to Computers		...
100	Hands-On Windows	20	...
140	Structured Analysis	20	...
25	Intro to Programming	140	...
...

The SECTION TABLE

The SECTION TABLE includes all the individual sections a course may have. An individual course may have zero, one, or many sections, each of which can be taught at different rooms, times, and by different instructors. The primary key of the table is the SECTION_ID. The foreign key that links back to the COURSE table is the COURSE_NO column. The SECTION_NO column identifies the individual section number. For example, for the first section of a course, it contains the number 1; the second section lists the number 2, and so on. The two columns, course_no and section_no, also uniquely identify a row, but section_id has been created instead. This SECTION_ID column is called a surrogate key because it does not have any meaning to the user.

The column START_DATE_TIME shows the date and time the section meets for the first time. The LOCATION column lists the classroom. The CAPACITY column shows the maximum number of students that may enroll in this section. The instructor_ID column is another foreign key column within the SECTION table; it links back to the INSTRUCTOR table. The relationship between the SECTION and the INSTRUCTOR table indicates that an instructor must always be assigned to a section. The INSTRUCTOR_ID column of the SECTION table may never be null and when you read the relationship from the opposite end, you can say that an individual instructor may teach zero, one, or multiple sections.

The relationship line leading from the course table to the section table means that a course may have zero, one, or multiple sections. Conversely, every individual section *must* have a corresponding row in the course table.

Relationships between tables are based on *business rules*. In this case, the business rule is that a course can exist without a section, but a section cannot exist unless it is assigned to a course. As mentioned, this is indicated with the bar (|) on the other end of the relationship line. Most of the child relationships on the schema diagram are considered mandatory relationships (with two exceptions); this dictates that the foreign key

columns in the child table must contain a value (must be NOT NULL) and that value must correspond to a row in the parent table via its primary key value.

The INSTRUCTOR TABLE

The INSTRUCTOR table lists information related to an individual instructor, such as name, address, phone, and zip code. The ZIP column is the foreign key column to the ZIPCODE table. The relationship between the INSTRUCTOR and the ZIPCODE is an optional relationship so a null value in the ZIP column is allowed. For a given ZIP column value there is one and only one value in the ZIPCODE table. For a given ZIP value in the ZIPCODE table you may find zero, one, or many of the same value in the INSTRUCTOR table. Another foreign key relationship exists to the SECTION table: an instructor may teach zero, one, or multiple sections and an individual section can be taught by one and only one instructor.

THE ZIPCODE TABLE

The primary key of zipcode is the ZIP column. For an individual zip code it allows you to look up the corresponding CITY and STATE column values. The datatype of this column is VARCHAR2 and not a NUMBER, as it allows you to enter leading zeros. Both the STUDENT and the INSTRUCTOR table reference the ZIPCODE table. The relationship between the ZIPCODE and STUDENT tables is mandatory: For every ZIP value in the STUDENT table there must be a corresponding value in the ZIPCODE table, and for one given zip code, there may be zero, one, or multiple students with that zip code. In contrast, the relationship between the INSTRUCTOR and ZIPCODE table is optional; the ZIP column of the INSTRUCTOR table may be null.

What about Delete Operations?

Referential integrity does not allow deletion in a parent table of a primary key value that exists in a child table as a foreign key value. This would create orphan rows in the child table. There are many ways to handle deletes and you will learn about this topic and the effects of the deletes on other tables in Chapter 10, "Insert, Update, and Delete."

THE ENROLLMENT TABLE

The enrollment table is an intersection table between the STUDENT and the SECTION table. It lists the students enrolled in the various sections. The primary key of the table is a composite primary key consisting of the STUDENT_ID and SECTION_ID columns. This unique combination does not allow a student to register for the same section twice. The ENROLL_DATE column contains the date the student registered for the section and the FINAL_GRADE column lists the student's final grade. The final grade is to be computed from individual grades such as quizzes, homework assignments, and so on.

The relationship line between the ENROLLMENT and STUDENT tables indicates that one student may be enrolled in zero, one, or many sections. For one row of the enrollment table you can find one and only one

corresponding row in the STUDENT table. The relationship between the ENROLLMENT and SECTION table shows that a section may have zero, one, or multiple enrollments. A single row in the ENROLLMENT table always links back to one and only one row in the SECTION table.

The GRADE_TYPE table

The GRADE_TYPE table is a lookup table for other tables as it relates to grade information. The table's primary key is the GRADE_TYPE_CODE column that lists the unique category of grade, such as MT, HW, PA, and so on. The DESCRIPTION column describes the abbreviated code. For example, for the GRADE_TYPE_CODE of MT you will find the description Midterm, for HW you see Homework.

The GRADE TABLE

This table lists the grades a student received for an individual section. The primary key columns are STUDENT_ID, SECTION_ID, GRADE_TYPE_CODE, and GRADE_CODE_OCCURRENCE. For an individual student you will find the all the grades related to the section the student is enrolled in. For example, the listed grades in the table may include the midterm grade, individual quizzes, final examination grade, and so on. For some grades (e.g., quizzes, homework assignments) there may be multiple grades and the sequence number is shown in the GRADE_CODE_OCCURRENCE column. [Figure 1.33](#) displays an excerpt of data from the GRADE table. The NUMERIC_GRADE column lists the actual grade received. This grade may be converted to a letter grade with the help of the GRADE_CONVERSION table discussed later.

STUDENT_ID	SECTION_ID	GRADE_TYPE_CODE	GRADE_CODE_OCCURRENCE	NUMERIC_GRADE	...
221	104	FI	1	77	...
221	104	HM	1	76	...
221	104	HM	2	76	...
221	104	HM	3	86	...
221	104	HM	4	96	...
221	104	MT	1	90	...
221	104	PA	1	83	...
221	104	QZ	1	84	...
221	104	QZ	2	83	...
...

1.33 Data from the GRADE table.

From the relationship between the ENROLLMENT and GRADE table, you can learn that rows only exist in the GRADE table if the student is actually enrolled in the section listed in the ENROLLMENT table. In other words, it is not possible for a student to have grades for a section in which he or she is not enrolled. The foreign key columns STUDENT_ID and SECTON_ID from the ENROLLMENT table enforce this relationship.

THE GRADE_TYPE_WEIGHT table

The GRADE_TYPE_WEIGHT table aids in computation of the final grade a student receives for an individual section. This table lists how the final grade for an individual section is computed. For example, the midterm may constitute 50 percent of the final grade, all the quizzes 10 percent, and the final examination 40 percent. If there are multiple grades for a given GRADE_TYPE_CODE, the lowest grade may be dropped if the column DROP_LOWEST contains the value "Y". The final grade is determined by using the individual grades of the student and section in the GRADE table in conjunction with this table. This computed final grade value is stored in the FINAL_GRADE column of the ENROLLMENT table discussed previously. (The FINAL_GRADE column is a derived column. As mentioned, the values to compute this number are available in the GRADE and GRADE_TYPE_WEIGHT tables, but because the computation of this value is complex, it is stored to simplify queries.)

The primary key of this table consists of the SECTION_ID and GRADE_TYPE_CODE columns. A particular grade_type_cd value may exist zero, one, or multiple times in the GRADE_TYPE_WEIGHT table. For every row of the GRADE_TYPE_WEIGHT table you will find one and only one corresponding GRADE_TYPE_CODE value in the GRADE_TYPE table.

The relationship between the GRADE_TYPE_WEIGHT table and the SECTION table indicates that a section may have zero, one, or multiple rows in the GRADE_TYPE_WEIGHT table for a given SECTION_ID value. For one SECTION_ID value in the GRADE_TYPE_WEIGHT table there must always be one and only one corresponding value in the SECTION table.

The GRADE_CONVERSION TABLE

The purpose of the GRADE_CONVERSION table is to convert a number grade to a letter grade. The table does not have any relationship with any other tables. The column LETTER_GRADE contains the unique grades, such as A+, A, A-, B, and so forth. For each of these letter grades, there is an equivalent number range. For example, for the letter B, the range is 83 through 86 and is listed in the MIN_GRADE and MAX_GRADE columns.

You can find the individual table and column descriptions of all the tables listed in the STUDENT schema diagram in Appendix E, "Table and Column Descriptions."