Entity-Relationship Model

Employee Scheduling Program

Michael Kramarczyk
mdk5@njit.edu
CIS 631

Table of Contents

Introduction 3
Employee 3
  Employee Entity-Relationship Diagram 3
  Employee Table 4
  Employee Sample Data 4
Job 5
  Job Entity-Relationship Diagram 5
  Job Table 5
  Job Sample Data 5
Employee Job 6
  Employee Job Entity-Relationship Diagram 1 6
  Employee Job Entity-Relationship Diagram 2 6
  Employee Job Table 7
  Employee Job Sample Data 7
Employee Schedule 8
  Employee Schedule Entity-Relationship Diagram 1 8
  Employee Schedule Entity-Relationship Diagram 2 9
  Employee Schedule Table 9
  Employee Schedule Sample Data 9
Employee Vacation 10
  Employee Vacation Entity-Relationship Diagram 1 10
  Employee Vacation Entity-Relationship Diagram 2 10
  Employee Vacation Table 11
  Employee Vacation Sample Data 11

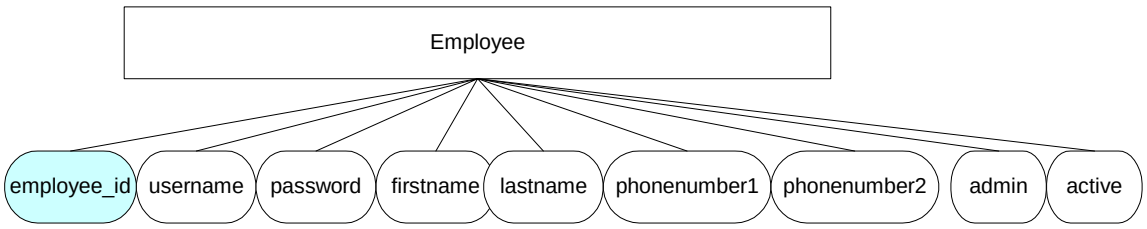
Introduction

The Entity-Relationship diagrams for the employee scheduling program are described in the below paper. The color: light blue * is used for the key value in the Entity-Relationship diagram. The color: yellow * is used for foreign key values in the Entity-Relationship diagram. There are several Entity-Relationship diagrams which have multiple fields marked in light blue. In these instances the multiple fields make up one unique key.

Employee

The employee entity-relationship model has several fields: employee_id, username, password, firstname, lastname, phonenumber1, phonenumber2, admin, and active. The employee_id is a key value which uniquely identifies an employee. Its beginning value is 10000. Each employee will have a unique username and password, which they use to enter into the system. The firstname and lastname combination does not have to be unique. In other words, the system will allow multiple users with the same first name and last name. The phonenumber1 field must be filled but the phonenumnber2 field can be left blank. The admin field must contain either a one or zero. If the value is one then the employee has admin rights when they log into the system. Otherwise, they do not have admin rights. The active field must contain a zero or one. If it is one then the employee is active and can be assigned jobs, scheduled for work, and scheduled for vacation. If the active value is set to zero the employee will remain in the system but can not be assigned jobs, scheduled for work, or scheduled for vacation. This will allow employers to keep historical employee information in the event that they return in the future.

Employee Entity-Relationship Diagram



Employee Table

Field	Value Description	Rules
employee_id (Key)	INTEGER	Key Value Starts at 10000. Must be unique. Must be filled.
username	VARCHAR(10)	Must be unique. Must be filled.
password	VARCHAR(10)	Must be filled.
firstname	VARCHAR(15)	Must be filled.
lastname	VARCHAR(15)	Must be filled.
phonenumber1	VARCHAR(15)	Must be filled.
phonenumber2	VARCHAR(15)	
admin	INTEGER	Must be filled. (1 or 0)
active	INTEGER	Must be filled. (1 or 0)

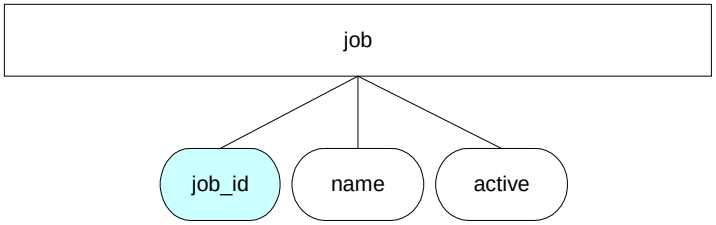
Employee Sample Data

employee_id	username	password	firstname	lastname	phonenumber1	phonenumber2	admin	active
10000	Mike	What	Mike	Kamp	732-923-1213		1	1
10001	Dave	Ok	Dave	Loose	781-123-1224	732-932-0121	0	1

Job

The job-entity relationship diagram model has several fields: job_id, name, and active. The job_id is a key value, which uniquely identifies a job. The name field is a unique value as well, but it can be changed. This particular design was chosen to allow administrators to change the name for a job function without affecting other tables in the system. For instance a job record with the name Waitress/Waiter can be changed to the more politically correct term: Wait Person. The active field must contain either a one or zero. One if the record is allowed as a valid choice in employee schedule, and employee Job or zero if the record is historical.

Job Entity-Relationship Diagram



Job Table

Field	Value Description	Rules
job_id (Key)	INTEGER	Key Value Starts at 1. Must be unique. Must be filled.
name	VARCHAR(15)	Must be unique. Must be filled.
active	INTEGER	Must be filled. (1 or 0)

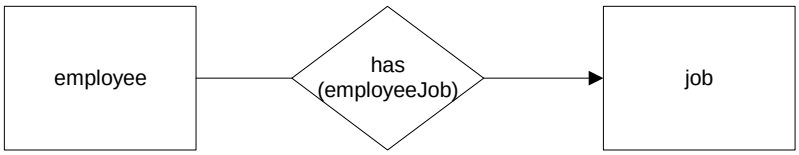
Job Sample Data

job_id	name	active
1	Cook	1
2	Prep Cook	1

Employee Job

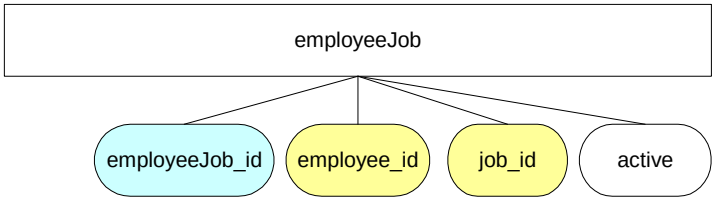
An employee can perform many different jobs as described in the diagram below. The employee job entity-relationship model describes the relationship between the employee table and job table.

Employee Job Entity-Relationship Diagram 1



The employeeJob table has several fields: employeeJob_id, employee_id, job_id, and active. The employeeJob_id is a unique key value which starts at 1. The employee_id is the same as employee_id in the employee table. It uniquely identifies an employee record. The job_id is the same as job_id in the job table. It uniquely identifies a job record. The active field must contain either a one or zero. One if the record is allowed as a valid combination and zero if the record is historical. Employers might mark an employeeJob record as historical if the employee can do a particular job but no longer performs it.

Employee Job Entity-Relationship Diagram 2



Employee Job Table

Field	Value Description	Rules
employeeJob_id (Key)	INTEGER	Key Value Starts at 1. Must be unique. Must be filled.
employee_id (Foreign Key)	INTEGER	Must be filled.
job_id (Foreign Key)	INTEGER	Must be filled
active	INTEGER	Must be filled. (1 or 0)

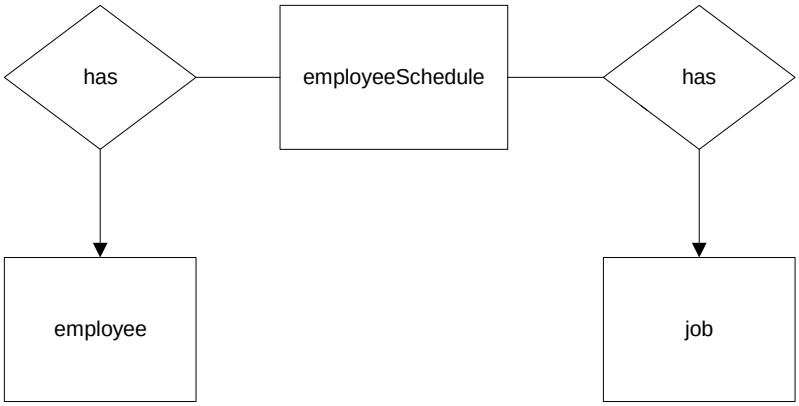
Employee Job Sample Data

employeeJob_id	employee_id	job_id	active
1	10000	1	1
2	10000	2	1

Employee Schedule

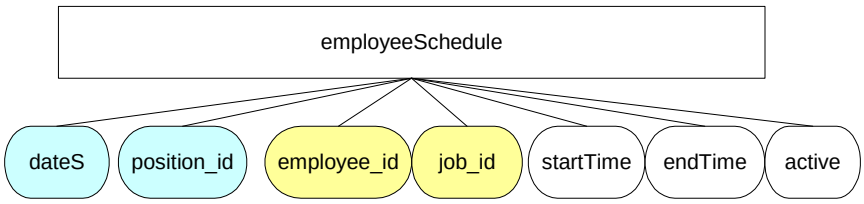
Employers often use a spreadsheet format for scheduling employees for work. This gives them the ability to see all the people scheduled working on a particular day. In order to mimic this format, each employee schedule record has a date (dateS) and a position (position_id). The position is a line number in the spreadsheet. The combination of date and position is unique and together they form the key value for a record. The diagram below shows the relationship between the employeeSchedule table and the employee table and the relationship between the employeeSchedule table and job table. An employeeSchedule record has an employee. An employeeSchedule record also has a job. The employeeJob table is also searched when creating an employeeSchedule record to make sure that employee is qualified to work that particular job. Also the employeeVacation table is scanned for that particular day to ensure that the employee is not on vacation.

Employee Schedule Entity-Relationship Diagram 1



The employeeSchedule table has several fields: dateS, position_id, employee_id, job_id, startTime, endTime, and active. As described above, the dateS and position_id are combined to form a unique identifier for the employeeSchedule record. The current limitation for the system is 75 positions per dateS. This is not a limitation of the database, but an imposed restriction on the program. The empoloyee_id is linked to employee_id in the employee table. It uniquely identifies an employee in the employee table. The job_id is linked to job_id in the job table. It uniquely identifies a job in the job table. The startTime is the beginning time of shift. The endTime is the ending time of the shift. The program ensures that the endTime is greater than or equal to the startTime. The active field contains either a one or zero. If it is one, then the record is valid. The necessary checks will be made to ensure that the employee is active, and the employee can be scheduled to work that particular job on that particular day.

Employee Schedule Entity-Relationship Diagram 2



Employee Schedule Table

Field	Value Description	Rules
dateS (Part of Key)	INTEGER	Key value together with position_id. The combination of dateS and position_id must be unique. Must be filled.
position_id (Part of Key)	INTEGER	Key value together with dateS. The combination of dates and position_id must be unique. Must be filled.
employee_id (Foreign Key)	INTEGER	Employee must not be scheduled for this day and may not be on vacation this day. Employee must have a job. The employee_id must exist.
job_id (Foreign Key)	INTEGER	Employee must be able to perform job. The job_id must exist. There must be an employee_id if there is a job_id.
startTime	VARCHAR(4)	Must be filled.
endTime	VARCHAR(4)	Must be filled.
Active	INTEGER	Must be filled. (1 or 0)

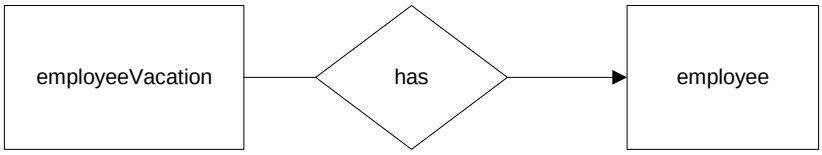
Employee Schedule Sample Data

dateS	position_id	employee_id	job_id	startTime	endTime	active
20051001	1	10000	1	0900	1600	1
20051001	2	10001	1	0900	1600	1
20051002	3			0900	1600	1

Employee Vacation

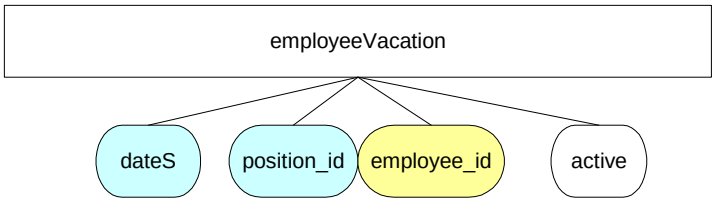
Employers often use a spreadsheet format for scheduling employees for vacation. This gives them the ability to see all the people scheduled for vacation on a particular day. In order to mimic this format, each employee vacation record has a date (dateS) and a position (position_id). The position is a line number in the spreadsheet. The combination of date and position is unique and together they form the key value for a record. The diagram below shows the relationship between the employeeVacation table and the employee table. An employeeSchedule record has an employee.

Employee Vacation Entity-Relationship Diagram 1



The employeeVacation table contains several fields: dateS, position_id, employee_id, and active. As described above, the dateS and position_id are combined to form a unique identifier for the employeeVacation record. The current limitation for the system is 75 positions per dateS. This is not a limitation of the database, but an imposed restriction on the program. The empoloyee_id is linked to employee_id in the employee table. It uniquely identifies an employee in the employee table. The active field contains either a one or zero. If it is one, then the employeeVacation record is valid. The necessary checks will be made to ensure that the employee is valid and they are not scheduled to work on that particular day. If the active field is marked zero, the checks will not be made.

Employee Vacation Entity-Relationship Diagram 2



Employee Vacation Table

Field	Value Description	Rules
dateS (Part of Key)	INTEGER	Key value together with position_id. The combination of dateS and position_id must be unique. Must be filled.
position_id (Part of Key)	INTEGER	Key value together with dateS. The combination of dates and position_id must be unique. Must be filled.
employee_id (Foreign Key)	INTEGER	Employee must not be scheduled for this day. The employee_id must exist.
active	INTEGER	Must be filled. (1 or 0)

Employee Vacation Sample Data

dateS	position_id	employee_id	active
20051001	1	10008	1
20051001	2	10009	1
20051001	3		