CS27020
Assignment: Ski Lifts and Pistes

Author:
James Euesden (jee22)

Address:
Department of Computer Science
Aberystwyth University
Aberystwyth
Ceredigion
SY23 3DB

Date: February 6, 2014

Contents

1 Introduction

2 Analysis

2.1 Functional Dependencies

To see the functional dependencies, I made a number of assumptions about the data, based upon the sample data provided.

2.1.1 Piste

pisteName ->grade, length, fall, open

pisteName does not functionally determine the liftName, as liftName could be a number of things. It does, however, determine the grade, length, fall and whether or not it is open. A logical primary key, based on the assumption that no Piste will be named the same, would be piesteName. This must be brought into First Normal Form (1NF) in order for this to work though.
By using pisteName, we are assuming that no 2 Pistes are named the same. This could be modified were 2 Pistes named the same, by including another attribue and making a composite key. A candidate for this would be piesteName and length. This would be more likely unique. However, for the sake of this design we shall assume no 2 Pistes may be named the same.

2.1.2 Lift

liftName ->type, summit, rise, length, operating

liftName implies all other attributes of the Lift. Assuming that no 2 lifts are named the same, we can accept liftName as an appropriate Primary Key. However, should it be that some Lifts are named the same, we could adapt this to make a composite key composed of liftName and length, giving the key a uniquely identifiable key. For this design however, we shall continue to assume no 2 lifts are named the same.

2.2 Unnormalized Structure

Based upon the sample data provided, the unnormalized structure of the Database is as follows:

<u>pisteName</u>
grade
length
fall
liftName*
open

<u>liftName</u>
type
summit
rise
length
operating

3 Normalizing the Data

To bring the Piste table into 1NF, we must remove the repeated groups of data (liftName). To do this, we create a new relation, "Connection". This relation contains the Primary Keys of both the Piste and Lifts, composing a composite key using the foreign keys of both. Resulting relations:

3.0.1 Piste

<u>pisteName</u>
grade
length
fall
open

3.0.2 Lift

<u>liftName</u>
type
summit
rise
length
operating

3.0.3 Connection

<u>pisteName*</u> (from Piste)
<u>liftName*</u> (from Lift)

The data is now in 1NF. Lift was already in 1NF as it had no repeating groups of data. The data is also in Second Normal Form (2NF) as it has only one Primary Key, and each of the attributes of the relations relies solely on this single Primary Key and not partially on something else.

4 PostgreSQL