Trevor Pirone 11/4/17

Database Management Lab #7

Part 1:

1. The table contains a lot of helpful data. It is a good start and it is on the right track to where the table should ultimately be. However, the table needs to be re-organized. This is because the table is not in any forms, so it has to be normalized. This means that the table will have to be re-arranged into 1NF. There will be no repeating groups meaning the values will be atomic.

PackageID	TagNumber	InstallDate	SoftwareCostUSD
AC01	32808	09-13-2005	754.95
DB32	32808	12-03-2005	380.00
DB32	37691	06-15-2005	380.00
DB33	57772	05-27-2005	412.77
WP08	32808	01-12-2006	185.00
WP08	37691	06-15-2005	227.50
WP08	57222	05-27-2005	170.24
WP09	59836	10-30-2005	35.00
WP09	77740	05-27-2005	35.00

The primary key is PackageID and TagNumber.

Part 2:

PackageID	TagNumber	InstallDate	SoftwareCostUSD	SoftwarePackageName	ComputerModel
AC01	32808	09-13-2005	754.95	macOS	Apple
DB32	32808	12-03-2005	380.00	Linux Mint	Apple
DB32	37691	06-15-2005	380.00	Linux Mint	Asus
DB33	57772	05-27-2005	412.77	LabouseurOS	Labouseur
WP08	32808	01-12-2006	185.00	Microsoft Windows	Apple
WP08	37691	06-15-2005	227.50	Microsoft Windows	Asus
WP08	57222	05-27-2005	170.24	Microsoft Windows	MSI
WP09	59836	10-30-2005	35.00	Linux Ubuntu	Alienware
WP09	77740	05-27-	35.00	Linux Ubuntu	Gateway

		2005			

The functional dependencies are as follows:

PackageID → SoftwarePackageName

TagNumber → ComputerModel

(PakageID, TagNumber) → InstallDate

(PakageID, TagNumber) → SoftwareCostUSD

The table is not in 3NF because the table is not in 2NF. This means that if we decide to change the tag number, then computer model would have an incorrect relationship which violates 2NF since computer model is dependent on tag number. To get a table into 2NF, all non-key attributes would have to be fully functional dependent on the primary key. Once the table is in 2NF, then we can get the table into 3NF if there is no transitive functional dependency.

Part 3:

Software Packages

PackageID	**SoftwarePackageName**
AC01	macOS
DB32	Linux Mint
DB33	LabouseurOS
WP08	Microsoft Windows
WP09	Linux Ubuntu

Computer Models

TagNumber	**ComputerModel**
32808	Apple
37691	Asus
57772	Labouseur
57222	MSI
59836	Alienware
77740	Gateway

Installation Details

PackageID	TagNumber	InstallDate	SoftwareCostUSD	SoftwarePackageName	ComputerModel
AC01	32808	09-13-2005	754.95	macOS	Apple
DB32	32808	12-03-2005	380.00	Linux Mint	Apple
DB32	37691	06-15-2005	380.00	Linux Mint	Asus
DB33	57772	05-27-2005	412.77	LabouseurOS	Labouseur
WP08	32808	01-12-2006	185.00	Microsoft Windows	Apple
WP08	37691	06-15-2005	227.50	Microsoft Windows	Asus
WP08	57222	05-27-2005	170.24	Microsoft Windows	MSI
WP09	59836	10-30-2005	35.00	Linux Ubuntu	Alienware
WP09	77740	05-27-2005	35.00	Linux Ubuntu	Gateway

Primary Keys

Software Packages – PackageID

Computer Models – TagNumber

Installation Details – (PackageID, Tag Number)

Functional Dependencies

(PakageID, TagNumber) → InstallDate

(PakageID, TagNumber) → SoftwareCostUSD

The tables are now in 3NF because not only is in in 2NF, it has no transitive functional dependency. This means all non-key attributes are fully functional dependent on the primary key. For example, in the Installation Details table, InstallDate and SoftwareCostUSD are only

dependent on (PackageID, TagNumber). In the Computer Models table, ComputerModel is only dependent on TagNumber. In the Software Packages table, SoftwarePackageName is only dependent on PackageID.

ER Diagram