CIS 310-01  
Due: 1/13/2016

Assignment 1

Chad Tracy

Chad Tracy

CIS 310-01

Assignment 1

Due 1/13/2016

5. Data Redundancy issues found in the table:

* Every individual working on each project is listed, even if they have been before. This creates redundancies in the employee records listed on the table. For example, John Newson is listed multiple times because he’s working on multiple projects.
* Fields for project name and number are redundant. They’re both listed every time the project is listed, which creates clutter and redundancies for this table. Projects should be broken down into its own table.
* Employee name info, including how much they get paid for their job, can be its own table in order to keep this one simple and reduced redundancies. This includes name, job code, charge, phone number, and employee number.

6. Changes to EMP\_PHONE and EMP\_NAME that need to be made:

The employee name and employee phone fields need to be repurposed into a separate table for employees in order to avoid redundancies on the main chart. The employee table will hold information such as employee number, name, occupation, and other contact info that will be linked to the current table with the employee number field. This will help to avoid update anomalies, as you should only have to update the employee table instead of having to manage both tables.

7. Data sources from Problem 5:

The sources of this data seem to come from only a handful of sources, which appears to have created redundancies for the file.

8. New files that must be created in order to eliminate redundancies:

One of the issues with this file is that it attempts to fit too many fields into one file when it isn’t necessary. There should be files created for employee contact data, the projects, and the jobs each employee does. Then the files need to be linked to each other so that all of their data can be linked together.

Projects and employees need to have a many-to-many relationship because multiple employees can work on multiple projects, projects should have a many-to-many as there are multiples of both, and lastly there should be one employee for every one job since they can only do one job on a project at a time.