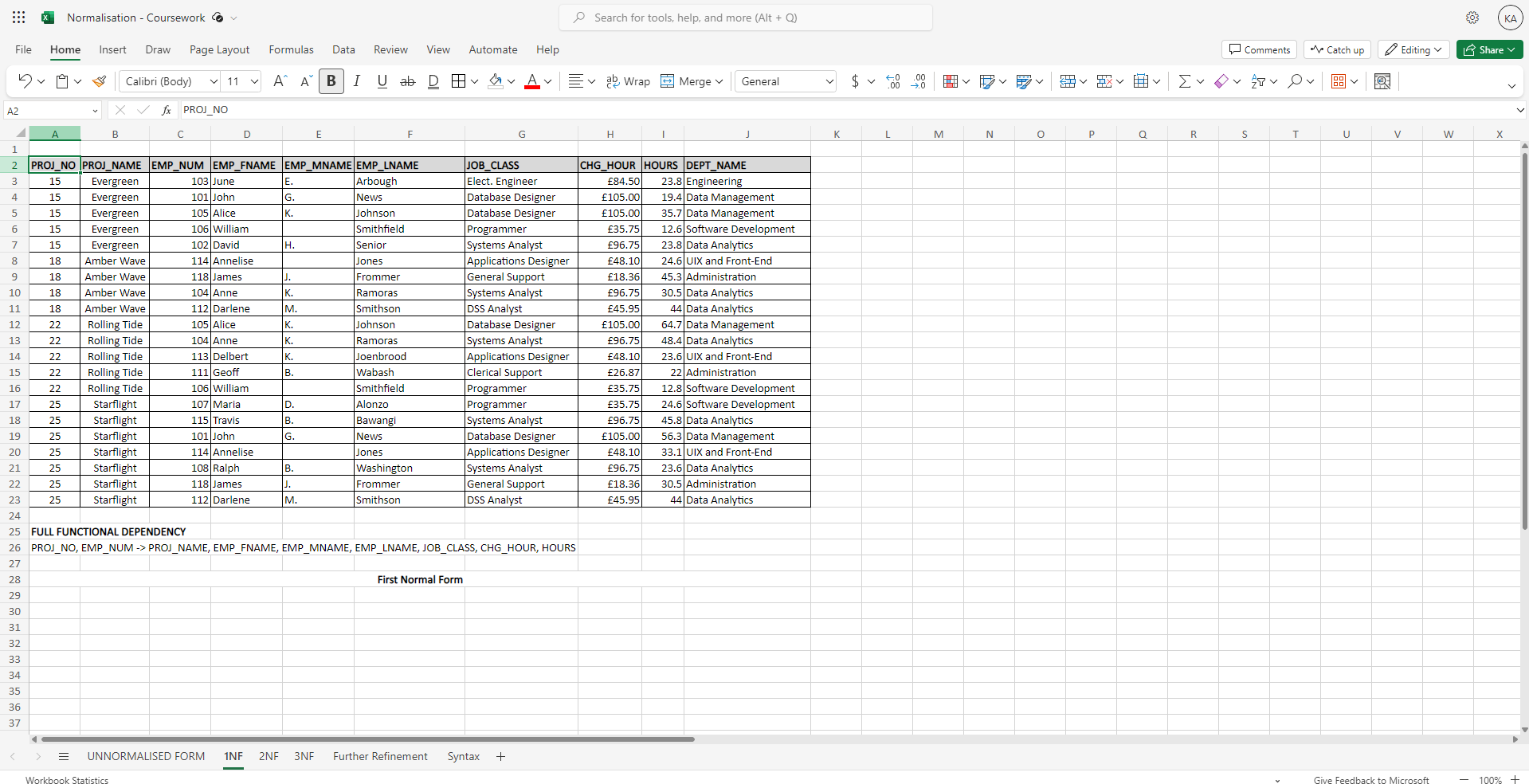
**DBCoursework Normalisation**

**1NF:**

Here, I’ve unmerged all the cells to ensure that it can clearly be seen. Therefore, I looked at the column EMP\_NAME which I then broke it down into 3. This is important as it is the first step to ensure that there is no atomicity. Now we have a table representing a single atomic value.

To ensure to identify the full functional dependency. I decided to make the Employee Number the Primary key and the Project number the Composite primary key. Therefore we will combine Project Number and Employee Number.

**Tale Example:**

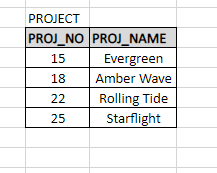


PROJ\_NO, EMP\_NUM -> PROJ\_NAME, EMP\_FNAME, EMP\_MNAME, EMP\_LNAME, JOB\_CLASS, CHG\_HOUR, HOURS

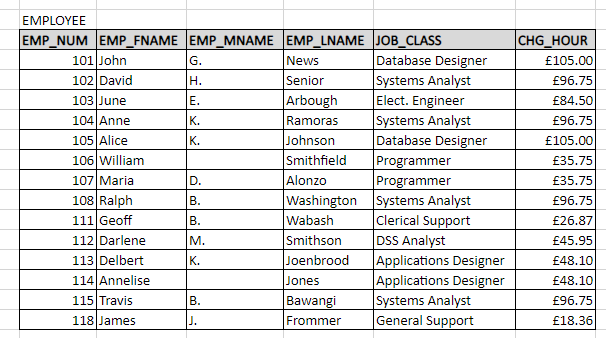
Above we can see that I have created a composite key: PROJ\_NO and EMP\_NUM.

**2NF:**

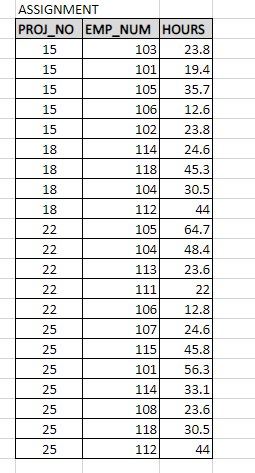
Second normal form is when I start to decompose the table that Enigma Inc has given me. Therefore, I have ensured to carry some attributes into new tables. We have gotten every attribute that is partially dependent on a certain other attribute. One of the examples is PROJECT\_NAME and PROJECT\_NUMBER.. Below, I emphasised how it has they have moved to a single table:



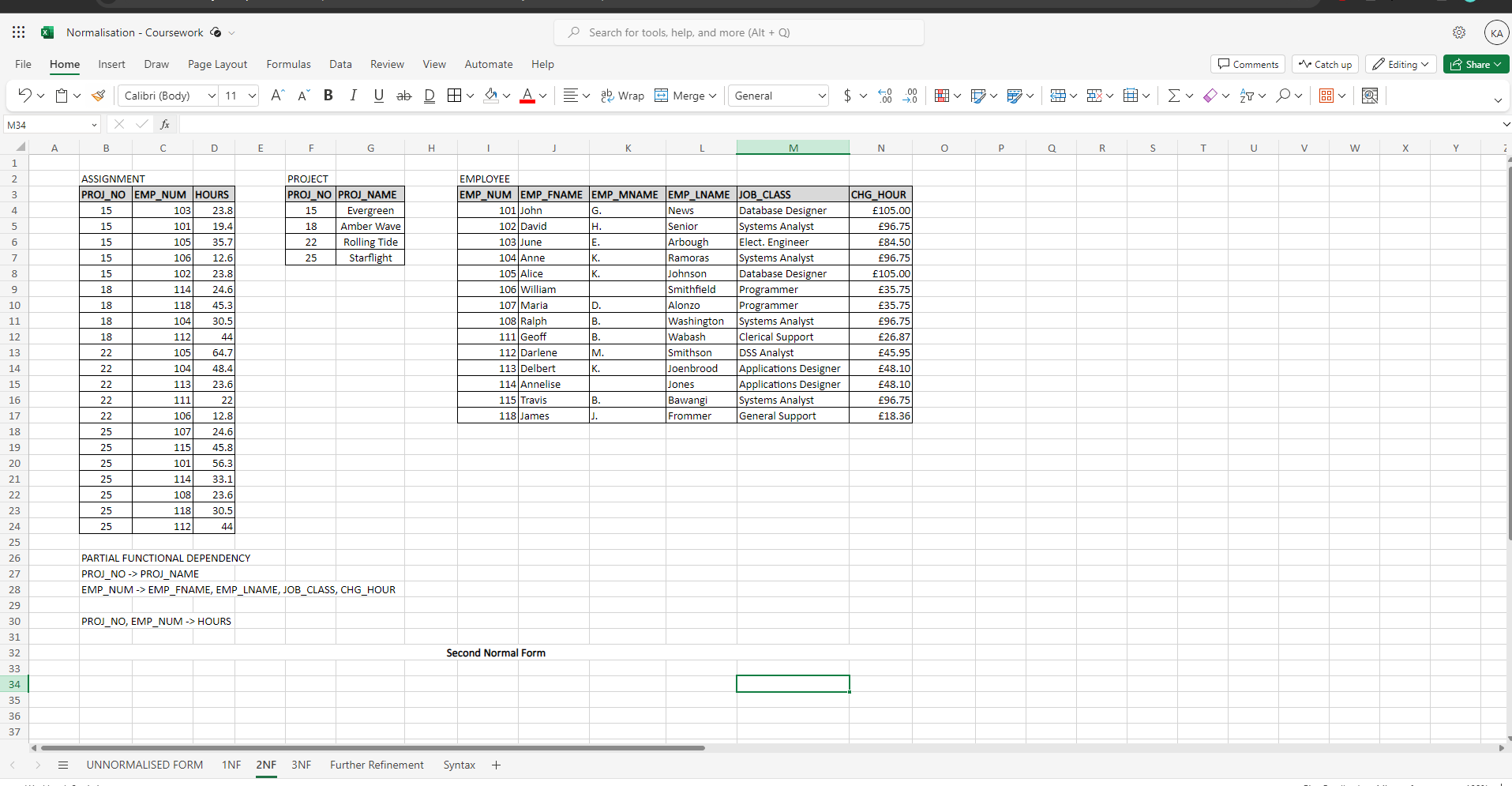
Moreover, there are other tables created that a partially dependant:



**And**



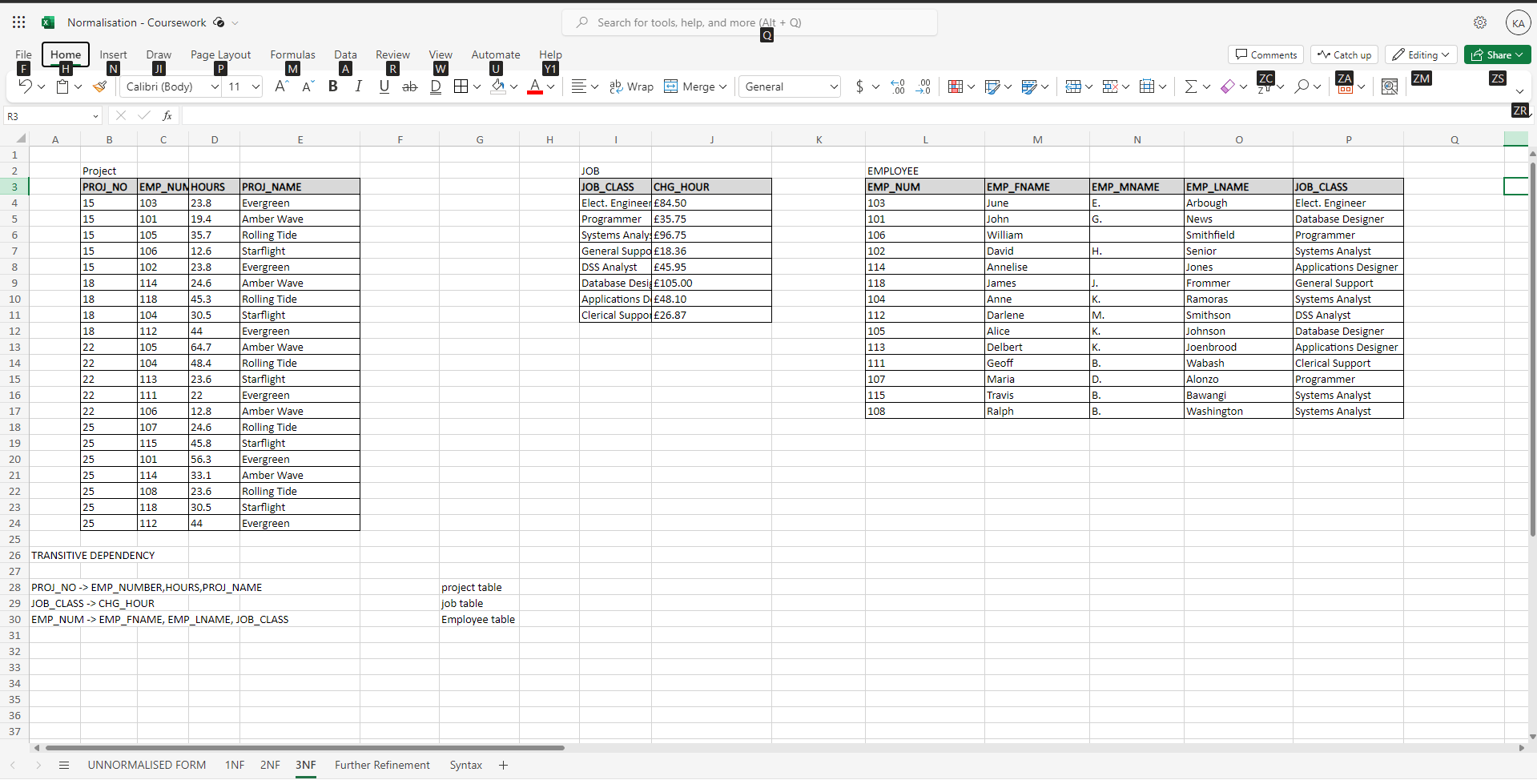
Moreover, everything has now been moved to their designated tables. However, we can still see in ASSIGNMENT Table. PROJ\_NO and EMP\_NUM are still the foreign keys.



**3NF:**

Here we did the 3rd normalisation. This includes moving PROJECT\_NAME into the assignment table which is now called project table and giving every staff a project to ensure that there is no Transitive Dependency by ensuring all non primary attributes that are dependent on a primary key. Thus, by ensuring that all attributes are fully dependent on a primary key. Tables can also own foreign keys that will be used for connections using Crows Foot.

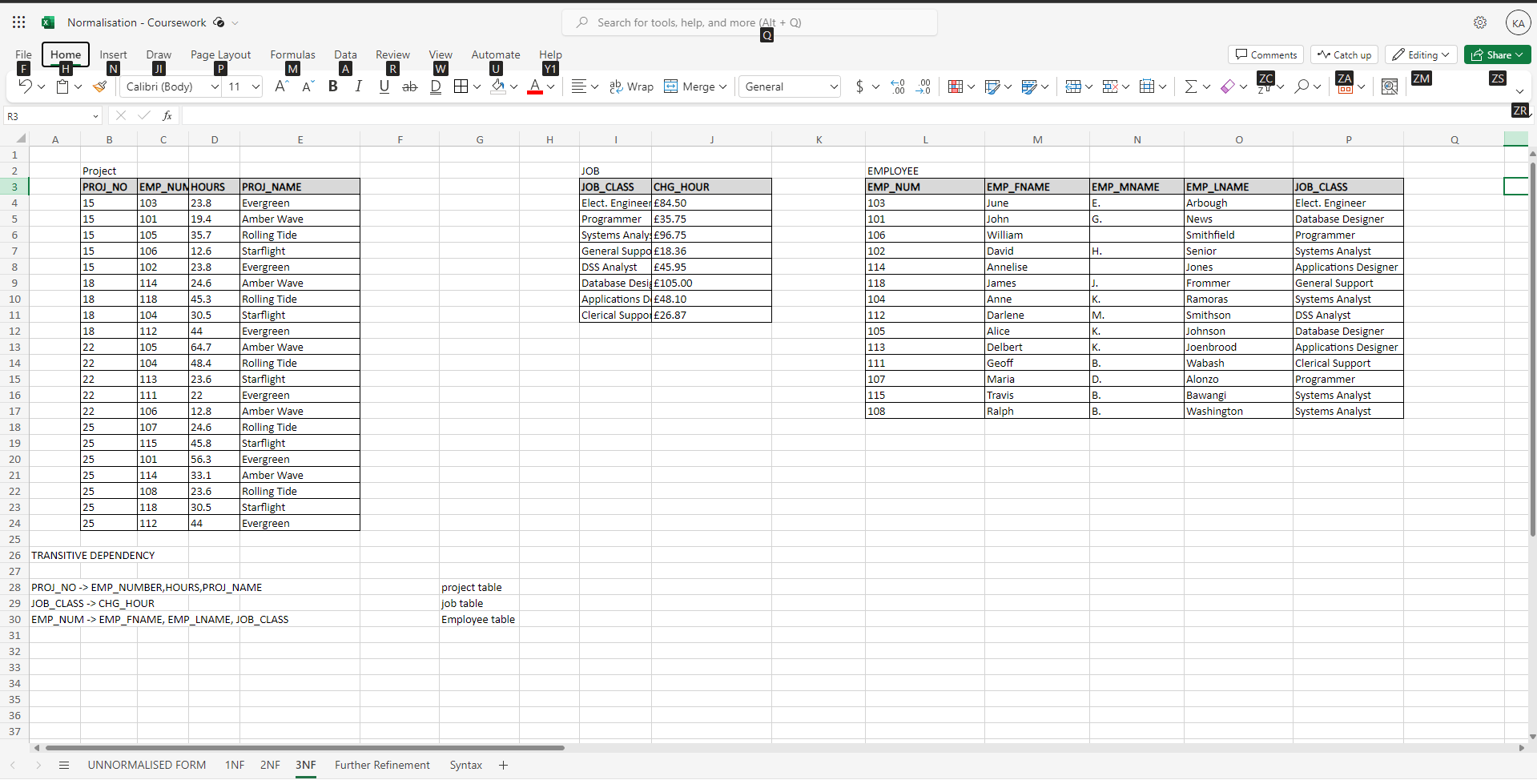
I will show this here:



1

**Finished 3NF:**

**Here we can see that every non primary attribute are dependent on a primary key.**



Project Table: PROJ\_NO -> EMP\_NUMBER, HOURS, PROJ\_NAME

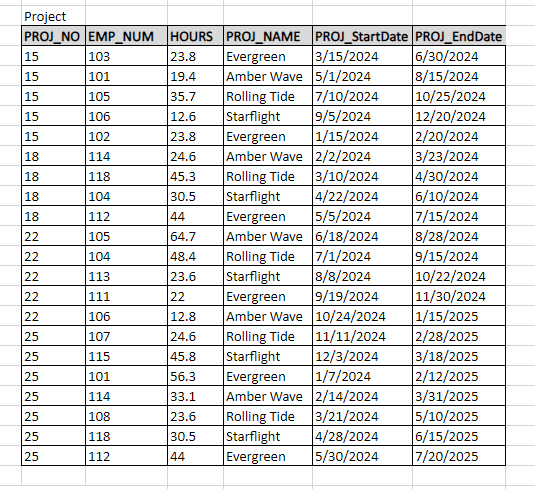
Job Table: JOB\_CLASS -> CHG\_HOUR

Employee Table: EMP\_NUM -> EMP\_FNAME, EMP\_LNAME, JOB\_CLASS

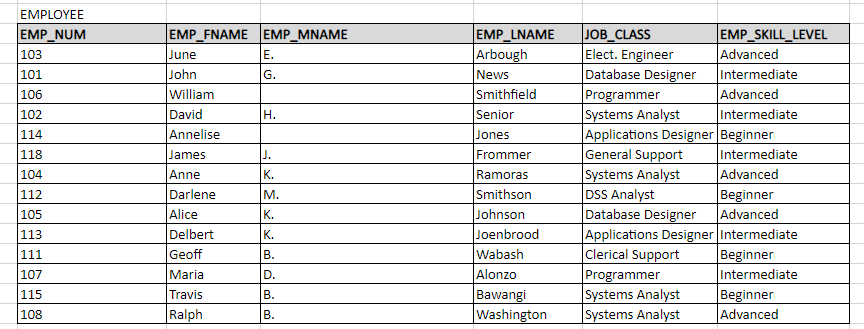
**Further Refinement:**

Here I decided to add the attributes Project\_StartDate and Project\_EndDate. This is something extra Enigma Inc can have to ensure that the projects get done in time as there is a deadline.

-I will show this here:



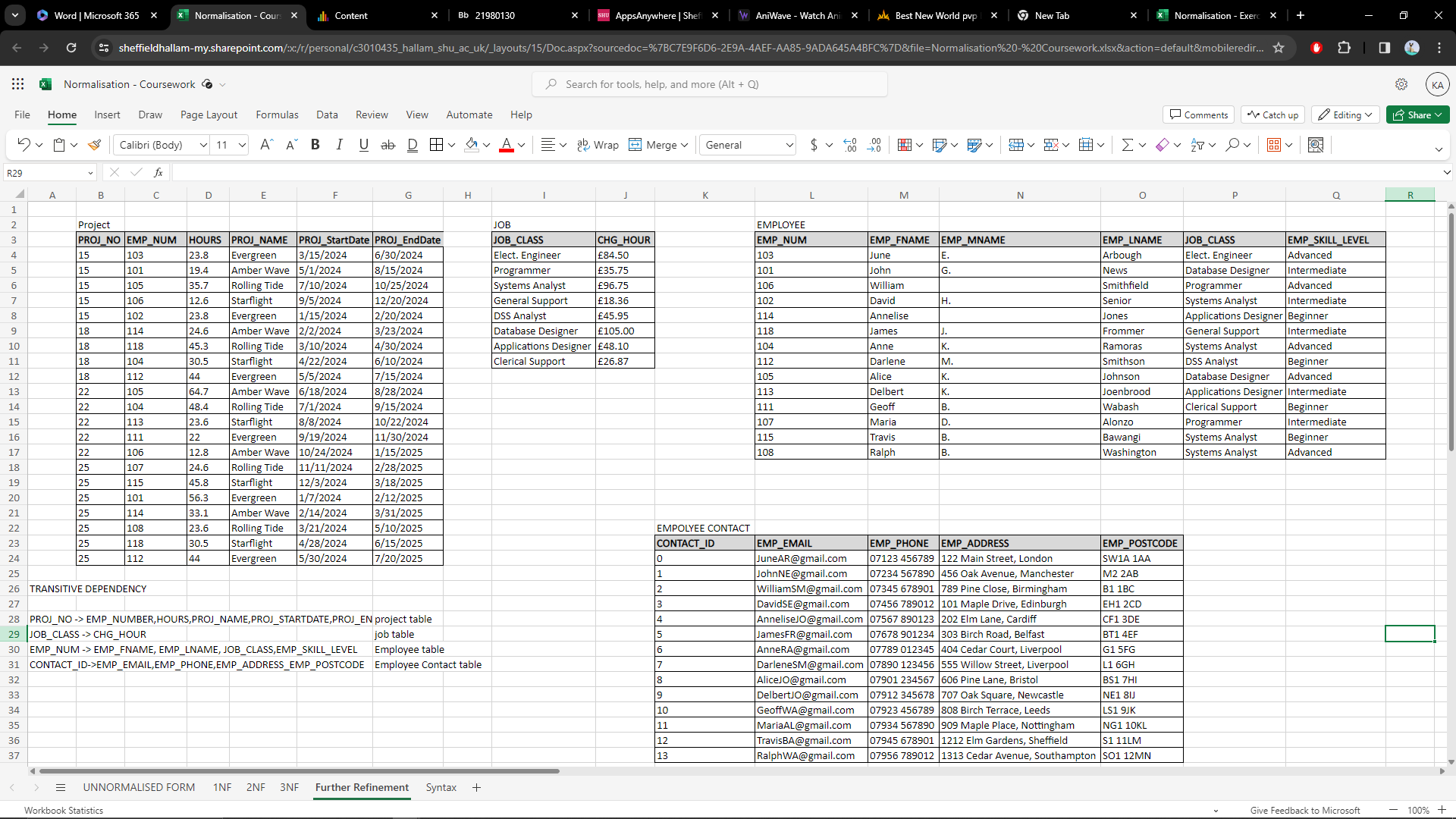
I then decided to add an attribute named EMP\_SKILL\_LEVEL. This is important for Enigma Inc as they will be able to allow more experienced workers to work on the difficult project. That being said, it’s overall a good way to manage a team. Here is the table:



Finally, I decided to add an Employee Contact table. This ensures that each an every Enigma Inc member of staff is able to get in touch in case something goes wrong such as a staff member not being able to attend work on a certain day. Moreover, the CONTACT\_ID attribute is important as it can define a unique staff member. Therefore, making it a primary key. Enigma Inc can also think about using biometric access with the CONTACT\_ID to ensure that each staff member has access to their office or maybe their designated project room. Here’s the table:



**Full Table:**



**Project Table:**

**PROJ\_NO -> EMP\_NUMBER,HOURS,PROJ\_NAME,PROJ\_STARTDATE,PROJ\_ENDDATE**

**Job Table:**

**JOB\_CLASS -> CHG\_HOUR**

**Employee Table:**

**EMP\_NUM -> EMP\_FNAME, EMP\_LNAME, JOB\_CLASS,EMP\_SKILL\_LEVEL**

**Employee Contact Table:**

**CONTACT\_ID->EMP\_EMAIL,EMP\_PHONE,EMP\_ADDRESS\_EMP\_POSTCODE**

Query to Retrieve Job Information:

To retrieve job-related information for an employee, you can use a JOIN operation between the Employees and Job Class tables based on the JOB\_CLASS.

sql

Copy code

SELECT Employees.EMP\_NUM, Employees.EMP\_FNAME, JobClass.JOB\_CLASS, JobClass.CHG\_HOUR

FROM Employees

JOIN JobClass ON Employees.JOB\_CLASS = JobClass.JOB\_CLASS;