Section 1: Using the table below, answer the following questions relating to Functional Dependencies, closures and keys. Save your results to a word document, PDF or a clear picture of a hand written solution. Pay close attention to the content of each column. Also, consider this to be the only data available to you. For example, knowing that there may be more two towns, in different states, with the same name, does not apply to this small sample set. Use only the data presented to answer the questions.

FirstName	LastName	DOB	Street	City	State	ZIP
John	Doe	09/20	Pacific Ave	Tacoma	WA	98402
John	Doe	09/21	Market Ave	Tacoma	WA	98405
John	Oberg	09/21	12th St	Portland	OR	97035
Steve	Adkisson	05/05	12th St	Seattle	WA	98104
Earnest	Smith	06/04	Pacific Ave	Tacoma	WA	98402
Mark	Smith	06/04	Main St	Arlington	TX	76001
Mark	Smith	05/05	Pacific Ave	Los Angeles	CA	90012
John	Doe	10/20	Market Ave	Los Angeles	CA	90012

1. List at least 12 completely non-trivial functional dependencies (list only dependencies with a single attribute on the right side (e.g $AB \rightarrow C, AB \rightarrow D$)

2. Calculate $\{Street, City\}^+$

3. Calculate $\{City, ZIP\}^+$

4. What columns if combined would make a key for this table? Use closure to prove it is the key. (Calculate the closure, show your steps).

Section 2: Using a digital tool of your choice (draw.io, Visio, PowerPoint, etc), draw an Entity Relationship diagram (using UML notation) that would describe the following system.

You are starting a new business to promote reading through organized book club events. If you are unfamiliar with a book club, it is a group of individuals that meet regularly to discuss a book that all of the members have read. Your system should allow for individuals to start new clubs or join existing clubs. Consider the following requirements when drawing your ERD. Suggested Entity names are in **Bold**. Choose 3 attributes per entity and identify what would make an instance Unique (what would likely be the key).

- A **Member** can belong to any number of **Club**s, and **Club**s can have any number of **Member**s.
- A **Club** must have a single Organizer. Organizers are also considered **Member**s. This Organizer can only Organize a single **Club**.
- Each **Member** donates a different amount to each **Club** they join. This can be considered a membership fee, and can be seen as an attribute of the membership relationship/association.
- Organizers/**Member**s also have backup **Member**s in case they cannot host a particular meeting.
- Each **Club** may choose to hold a bank **Account** for which to serve as a coffer. The **Account** is related to the **Club** itself, as well as the **Member** identified as the Organizer.

You can choose to implement Organizers explicitly as a subclass of Members or simply consider Organizers as a special case of Member.