A mail delivery service company **receives**packages/mail from customers at a specific **location or picks up packages**from a specific location. These packages/mail are **processed** by an **employee**. These packages/mail are then sent into a **processing system** where they will later be **delivered** to a recipient. There will be a **time and date** when the package/mail was **deposited** and when it was **received**.

Did you ever think about the system that makes a delivery system happen? What are the logical connections between data structures? For instance, it is clear that a single package is somehow connected to a customer, as well as the drop-off point - but how does all this fit together in a logical structure? What fields do you need in your tables, and how are they structured?

Now it is time to design a database for such a mail delivery company. The company doesn’t have any special requirements; the specification is opened for interpretation.

For this CodeReview, the following criteria will be graded:

**Basic points:**

1. (25 points) Use **4 steps for database development process** - ensure that your database is in 3rd Normal Form. Ensure that all elements given as bold in the task description (Task description: Mail Delivery Service) are covered in your database design. (hint: use draw.io tool, but try to draw the delivery process first on a paper before creating any diagrams in digital form). Please save your diagram(s) as **pdf files** in one folder and upload this folder to the project database to GitHub.
2. (45 points) Recreate your database design in MySQL. Name your database as **cr09\_john\_doe\_delivery** (use your name instead of "john" and "doe"). Export your database as cr09\_john\_doe\_delivery.sql dump file and then upload the .sql file with the project to GitHub.
3. (10 points) Insert **test data** into the database (insert some dummy data in your database, at least 3 sets of data per table)
4. (20 points) Create at least**6 different SQL queries** to show the power of your database (save these queries to  **cr09\_john\_doe\_delivery\_queries.sql** file and upload it to the project) Note: The use of JOINS is required for at least two queries). These queries should show similar results to:  how many packages/mail were sent from this specific city, who sent packages/mail on this date, who sent packages/mail between this and that date, etc.