-: ONLINE STORE STARTUP:-

Question: - Consider that you have got an internship in an upcoming unicorn startup. The organisation wants to create an online survey feature for the different categories of the products sold by them to help with their business intelligence and make out useful strategies to improve. Now, you are given the very first task which is to design a database schema for this scenario to store the necessary data.

Few of the requirements for the survey are as follows:

1. The survey can be reused to get feedback for multiple products. It would contain the ratings given by the buyer.

2. A customer is the registered user to whom the shopping account belongs. A customer will take up a survey when he/she will purchase a product.

3. A survey will have multiple questions. These questions are selected from a large dataset containing different types of questions. The

questions can be subjective as well as objective types of questions.

4. Customers and buyers (the customers who have made a purchase) are stored separately.

5. Buyers have to answer the questions given in the feedback, some of which are mandatory and some are optional. These answers should also be stored separately so that analysis can be done on all the answers received till date for the products.

6. It should have all the details of the customer and the product bought with it's feedback.

Solution: -

1. Entities:
   - Customer
   - Product
   - Survey
   - Question
   - Answer
   - Feedback

2.Attributes:

- Customer: customer_id(primary key), name,email,phone_number,address,registration_date
- Product: product_id(primary key),name,category,price,description,image_url
-
- Survey: survey_id(primary key), title,description,start_date,end_date
- Question: question_id(primary key) , text, type(subjective/objective) , mandatory
- Answer: answer_id(primary key), question_id(foreign key), buyer_id(foreign key), answer_text , rating
- Feedback: feedback_id(primary key), buyer_id(foreign key), product_id(foreign key), survey_id(foreign key)

Relationships:

- A customer can have multiple feedbacks, but a feedback can only belong to one customer. (one-to-many relationship between Customer and Feedback)

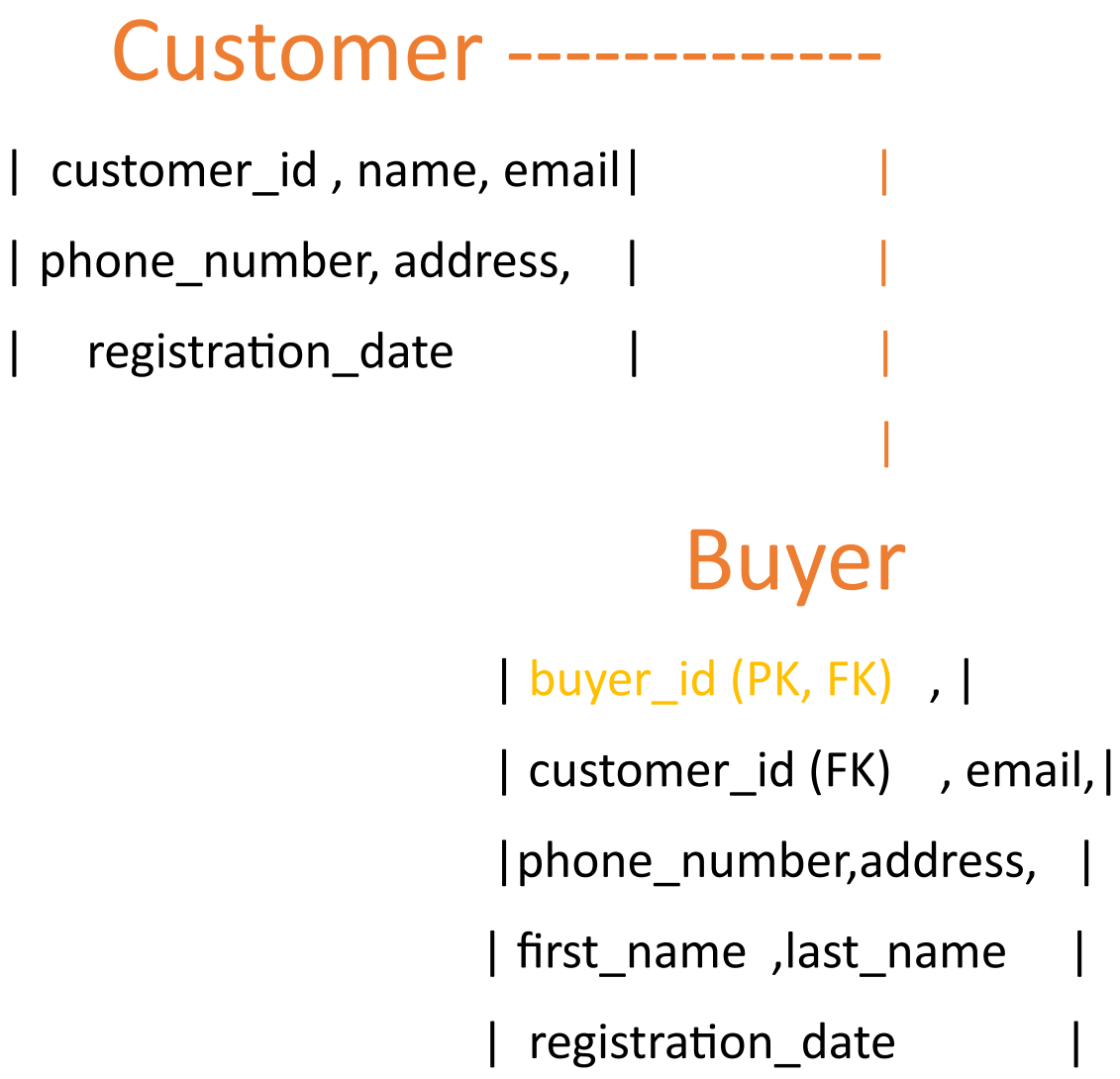
- A product can have multiple feedbacks, but a feedback can only be for one product. (One-to-many relationship between Product and Feedback)

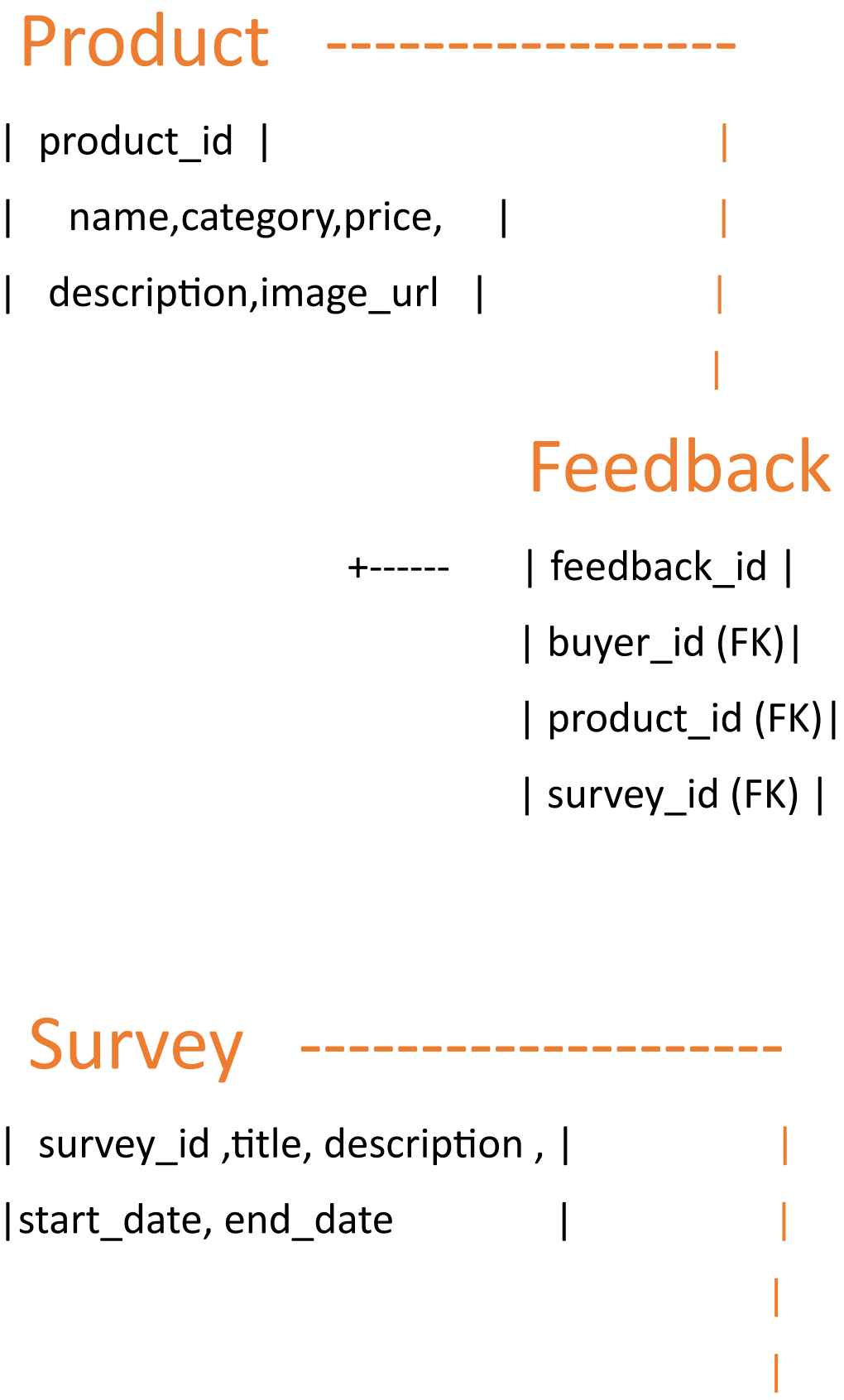
- A survey can have multiple questions, and a question can belong to multiple surveys. (Many-to-many relationship between Survey and Question)

- A buyer can give multiple answers in feedback, but an answer can only belong to one buyer. (one-to-many relationship between Buyer and Answer)

- A feedback can have multiple answers, but an answer can only belong to one feedback. (One-to-many relationship between Feedback and Answer)

ER Diagram:


Customer ----------
| customer_id , name, email|
| phone_number, address, |
| registration_date |
Buyer
| buyer_id (PK, FK) , |
| customer_id (FK) , email,|
|phone_number,address, |
| first_name ,last_name |
| registration_date |


Product
| product_id |
| name,category,price, |
| description,image_url |
Feedback
+------
| feedback_id |
| buyer_id (FK)|
| product_id (FK)|
| survey_id (FK) |
Survey
| survey_id ,title, description , |
|start_date, end_date |

|

Question

| question_id, text, type, |

| mandatory |

Answer

| answer_id ,question_id,buyer_id,|

| answer_id, rating|

Relational Schema:

- Customer (customer_id PK, name, email, phone_number, address, registration_date)
- Product (product_id PK, name, category, price, description, image_url)
- Survey (survey_id PK, title, description, start_date, end_date)

- Question (question_id PK, text, type, mandatory)
- Feedback (feedback_id PK, buyer_id FK, product_id FK, survey_id FK)
- Answer (answer_id PK, question_id FK, buyer_id FK, answer_text, rating)
