**GitHub Link:**

**Data Entities, Attributes, Files, and Model**

1. List 5 entities and provide a short description of each to identify what they are. For example:  
  
 Argles – the main product of the business. Tracking a,b,c,d, and e..

Dibbles – people who make argles. Tracking f, g, h, i, and j.

**Your descriptions must be more business-specific.**

**Product:**

Description: Products represent the several items or services that are provided by the business for sale to customers. These clouds include physical goods, digital products or services.

**Attributes:**

* Product\_ID: A unique identifier assigned to each product for tracking purposes.
* Product\_Name: The specific name or title of the product.
* Brand\_Name: The brand or manufacturer associated with the product.
* Price: The price at which the product is sold to customers.

**Customer**

Description: Customers are individuals or entities who purchase the products or services from the business.

**Attributes:**

* Customer\_ID: A unique identifier assigned to each customer for tracking and identification purposes.
* Customer\_Name: The name of the customer or the name of the entity representing the customer (e.g., company name).
* Address: The physical or mailing address of the customer for correspondence and delivery purposes.

**Payment**

Description: Payments represent the financial transactions made by the customers to purchase products or services from the business.

**Attributes:**

* Payment\_ID: A unique identifier assigned to each payment transaction for tracking and reference.
* Product\_Name: The name of the product or service for which the payment is made.
* Quantity: The quantity of the product or service purchased in the payment transaction.
* Price: The price per unit of the product or service.
* Total\_Amount: The total amount paid in the payment transaction.
* Customer\_ID: The unique identifier of the customer making the payment.
* Product\_ID: The unique identifier of the product or service associated with the payment.

**Review**

Description: Reviews represent the feedback provided by the customers regarding their experience with a specific product or service purchased from the business.

**Attributes:**

* Review\_ID: A unique identifier assigned to each review for tracking and reference.
* Product\_ID: The unique identifier of the product or service being reviewed.
* Customer\_ID: The unique identifier of the customer providing the review.
* Review\_Description: The written description or comments provided by the customer as part of the review.
* Review\_Star: The rating given by the customer to the product or service typically represented by stars (out of 5 stars).

**order**

Description: Orders represent requests made by the customers to purchase one or more products or services from the business.

Attributes:

* Order\_ID: A unique identifier assigned to each order for tracking and reference.
* Customer\_Name: The name of the customer placing the order.
* Product\_Name: The name of the product or service being ordered.
* Payment\_ID: The unique identifier of the payment transaction associated with the order.
* Customer\_ID: The unique identifier of the customer placing the order.
* Product\_ID: The unique identifier of the product or service being ordered.

2. Create 5 simple tables describing your entities, their attributes, and the data type of each attribute. For example:

| **Product Info** |  |
| --- | --- |
| Product Detail | varChar[20] |
| Quantity | Integer |
| Cost | Float |

| Product |  |
| --- | --- |
| Product\_Id | varchar[50] |
| Product\_name | varchar[50] |
| Brand\_name | varchar[50] |
| Price | Float |

This entity represents items available for sale in the business. Each product is uniquely identified by a Product\_ID and includes attributes such as Product\_Name (the name of the product), Brand\_Name (the brand or manufacturer of the product), and Price (the price of the product).

| Customer |  |
| --- | --- |
| Customer\_id | varchar[50] |
| Customer\_Name | varchar[50] |
| Address | varchar[50] |
| Product\_id | varchar[50] |

The Customer entity represents individuals who purchase products from the business. Each customer is identified by a unique Customer\_ID and includes attributes like Customer\_Name (the name of the customer) and Address (the address of the customer).

| Payment |  |
| --- | --- |
| Payment\_Id | varchar[50] |
| Product\_name | varchar[50] |
| Quantity | int |
| Price | float |
| Total\_Amount | float |
| Customer\_id | varchar[50] |
| Product\_id | varchar[50] |

The payment entity records transactions made by customers for their purchases. It includes attributes like Payment\_ID which is a unique identifier for each payment transaction, Product\_Name which is the name of the product being purchased. Quantity (the quantity of the product purchased), Price (the price per unit of the product), Total\_Amount (the total amount paid in the transaction), Customer\_ID (the ID of the customer making the payment), and Product\_ID (the ID of the product being purchased).

| Review |  |
| --- | --- |
| Review\_Id | varchar[50] |
| Product\_Id | varchar[50] |
| Customer\_id | varchar[50] |
| Review\_Description | varchar[50] |
| Review\_Star | varchar[5] |

he Review entity captures feedback provided by customers on products they have purchased. It includes attributes like Review\_ID (a unique identifier for each review), Product\_ID (the ID of the product being reviewed), Customer\_ID (the ID of the customer providing the review), Review\_Description (the text description of the review), and Review\_Star (the rating given by the customer, typically in stars).

| Order |  |
| --- | --- |
| Order\_id | varchar(50) |
| Customer\_Name | varchar(50) |
| Product\_name | varchar(50) |
| Payment\_id | varchar(50) |
| Customer\_id | varchar(50) |
| Product\_Id | varchar(50) |

The Order entity represents individual orders made by customers. Each order is identified by a unique Order\_ID and includes attributes like Customer\_Name (the name of the customer placing the order), Product\_Name (the name of the product being ordered), Payment\_ID (the ID of the payment transaction associated with the order), Customer\_ID (the ID of the customer placing the order), and Product\_ID (the ID of the product being ordered).

In this case, the product info is the entity, the items in the left column are the attributes, and the data types are in the right column. You will create one table per entity in your design for a total of 5 tables.

3. Create .CSV files for your entities and attributes. Each entity should have its own file. That means you will end up with 5 CSV files. Each file should also have 3 records that are specific examples of that entity. Save these files to your github.

4. Using Crow's Foot modeling, draw a **logical** model of your data entities and how they are related. Use appropriate arrows to show connections and requirements. Also, write a paragraph that describes the table and its connectivity using short statements. For example:  
 Products are connected to Sales in a One-to-Many relationship.   
  
 Remember, one to many means that a single product can be in multiple sales orders.

Be sure to label keys, attributes, and entities. You can use draw.io, or any modeling software of your choice. If the drawing is clear and visible, you can even use paper and pencil and then take a picture with your camera or phone.

here is a textual representation of the crow’s foot logical model for the given entities,

**Answer:**

**Entities:**

Product (PK: Product\_ID)

Customer (PK: Customer\_ID)

Payment (PK: Payment\_ID, FK: Customer\_ID, FK: Product\_ID)

Review (PK: Review\_ID, FK: Customer\_ID, FK: Product\_ID)

Order (PK: Order\_ID, FK: Customer\_ID, FK: Product\_ID, FK: Payment\_ID)

**Relationships:**

1. One Customer can make many Payments (One-to-Many)
2. One Customer can write many Reviews (One-to-Many)
3. One Customer can place many orders (One-to-Many)
4. one product can be part of many Payments ((One-to-Many)
5. one product can have many reviews (One-to-Many)
6. One Product can be in many Orders (One-to-Many)
7. One Payment is associated with one Customer (Many-to-One)
8. One Payment is associated with one Product (Many-to-One)
9. One Review is written by one Customer (Many-to-One)
10. One Review is for one Product (Many-to-One)
11. One Order is placed by one Customer (Many-to-One)
12. One Order contains one Product (Many-to-One)
13. One Order is associated with one Payment (Many-to-One)

This model describes the relationships between the entities showing how they are connected and the cardinality of each relationship. Each entity has its primary key (PK) identified and foreign keys(FK) are indicated where necessary to represent the relationships between entities.

