**Course Project   
DeVry University  
College of Engineering and Information Sciences**

**Course Number: CEIS236**

**Course Project Deliverable: 4**

**PART I:**

Question 1.

Question 2.

Question 3.

Question 4.

Question 5.

Question 6.

**Answer1:**

Please find below the Primary key and foreign key for the tables:

**1. Table Name:**CUSTOMER

**Primary key:**Cus\_Code

**Foreign key:**NA

**2. Table Name:**VENDOR

**Primary key:**V\_Code

**Foreign key:**NA

**3. Table Name:**INVOICE

**Primary key:**Inv\_Number

**Foreign key:**Cus\_Code: Reference from CUSTOMER table

**4. Table Name:**PRODUCT

**Primary key:**P\_Code

**Foreign key:**V\_Code: Reference from VENDOR table

**5. Table Name:**LINE

**Primary key**: (Inv\_Number, Line\_Number)

**Foreign key:**Inv\_Number : Reference from INVOICE

P\_Code: Reference from PRODUCT

**Answer 2:.**

Entity Integrity: An entity integrity is a feature of the entity which stores each record uniquely and it cannot be null. The given ERD has entity integrity in all tables as:

**1. Table Name:**CUSTOMER , stores the data based on a unique Cus\_Code where Cus\_Code cannot be null.

**2. Table Name:**VENDOR, stores the data based on a unique V\_Code where V\_Code cannot be null.

**3. Table Name:**INVOICE, stores the data based on a unique Inv\_Code where Inv\_Code cannot be null.

**4. Table Name:**PRODUCT, stores the data based on a unique P\_Code where P\_Code cannot be null.

**5. Table Name:**LINE, stores the data based on a unique (Inv\_Number, Line\_Number) which shows a line number with invoice and further each line has a product in it. Thus a record must stores a reference to exactly one product in invoice line and none of the composite attribute can be null.

**Answer 3.**

Referential Integrity is a foreign key reference in the table where it takes data from parent table. The table with foreign key is called a dependent table. The tables have below foreign key integrities:

**1. Table CUSTOMER** : NA

**2. Table VENDOR:**NA

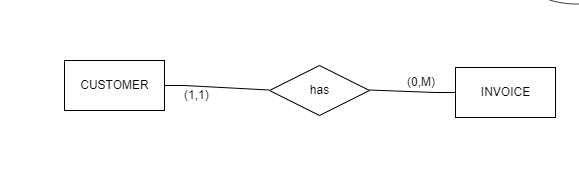
**3. Table INVOCE:**This table has a referential integrity from the table CUSTOMER as invoice is generated for exactly one CUSTMER and an INVOICE cannot exist without a CUSTOMER.

**4. Table PRODUCT:**This table has a referential integrity from the table VENDOR as product is supplied from exactly one VENDOR and a PRODUCT cannot exist without a VENDOR.

**5. Table LINE:**This table has a referential integrity from the table CUSTOMER as invoice is generated for exactly one CUSTMER and an INVOICE cannot exist without a CUSTOMER.

**Answer 4:**

CUSTOMER to INVOICE relation is ONE TO MANY relationship as a customer will have none or many invoices but INVOICE need to have exactly one CUSTOMER. INVOICE has full participation but CUSTOMER has partial participation as below:

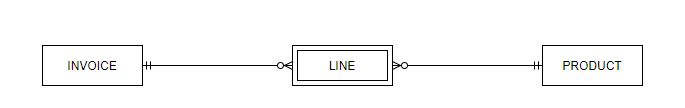


**Answer 5:**

LINE to INVOICE relation is MANY TO ONE relationship as an INVOICE will have none or many PRODUCTs added in LINE and a LINE will refer to exactly one PRODUCT in the records. A PRODUCT can have many LINE as it can be placed in many invoices.

**Answer 6:**

Please note that in ERD, LINE is a weak entity as it borrows Inv\_Number from INVOICE to make its primary key. Thus, conceptual ERD will look as below:



**PART II:** ERD

From the given scenario the following tables can be identified.

1)Team with columns ID,Name,colors where ID is the primary key.

2)Players with columns ID,First Name,LastName,Age where ID is the primary key.

3) Coaches with columns ID, Name,Team-id where ID is the primary key and Team-id as foreign key referencing Team table.

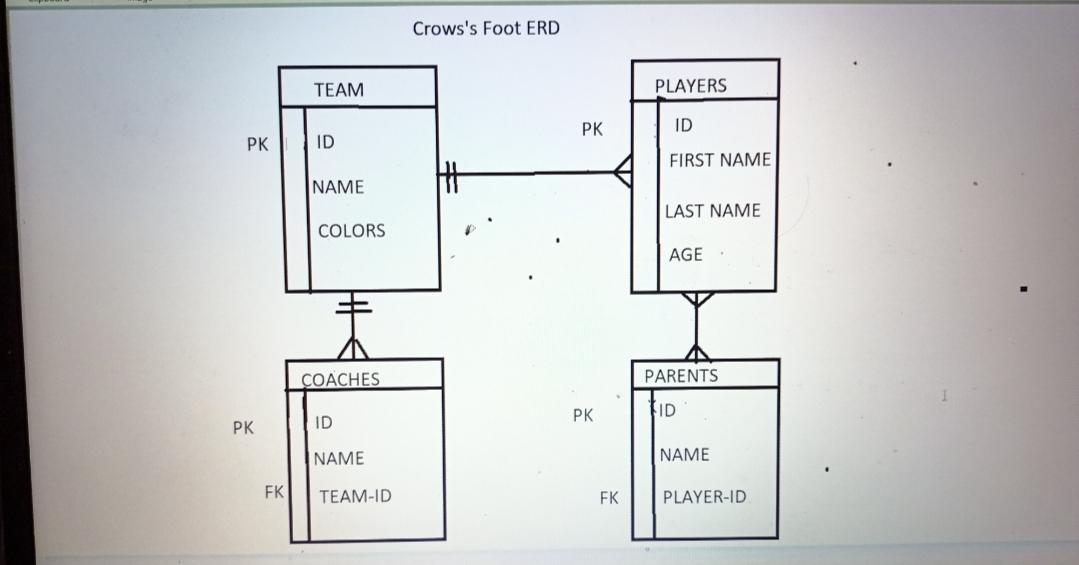
4) Parents with columns ID,Parent\_Name,Player\_ID where ID is primary key and Player\_ID as foreign key referencing players table.

Teams and Players have one to many relationship.ie One Team can have many players but one player can have only one team.

Teams and coaches also have one to many relationship.A team can have somany coaches but one coach can have only one team.

Parents and Players have many to many relationship.A parent can have many players and a player can have many parents.

All these tables and their relationships can be shown as in the following Crow's foot ER diagram.



In conclusion the Crow's foot ERD shows the tables and their primary key, foreign key constraints and the type of relationship between them.

**PART III:** Create Dependency diagram

Timeline

Description automatically generated