**#Solution-1:**

Here are the answers based on the given question.

a. Strong Entity : BANK, ACCOUNT, LOAN, and CUSTOMER.
b. The weak entity is BANK_BRANCHES whose partial key is Bank_no. and BRANCHES is its identifying relationship.
c. A partial key (Branch_no.) and an identifying relationship (BRANCHES) indicate the specific branch of the provided bank, as a bank may have multiple branches.
d. BRANCHES (BANK to BANK_BRANCH): (1,N) from BANK to BANK_BRANCH. A bank has at least one branch but also can have many branches.
   ACCTS (BANK_BRANCH to ACCOUNT):(1,N) from BANK_BRANCH to ACCOUNT. A bank branch may have multiple accounts, but it must have at least one.
   LOANS (BANK_BRANCH to LOAN): (1,N) from BANK_BRANCH to LOAN. A bank branch must have at least one loan but it may have many.

A_C (ACCOUNT to CUSTOMER): (M,N) from ACCOUNT to CUSTOMER. An account can be associated with many customers and a customer can be associated with many accounts.

L_C (LOAN to CUSTOMER): (M,N) from LOAN to CUSTOMER. A customer can be associated with many loans and A loan can be associated with many customers.

e. The given diagram is about the account and loans given to the customers in each branch. ER schema illustrates a system in a bank that should have a system set up for tracking client, account, and loan links in addition to keeping an eye on numerous branches, loans, and accounts, and allowing consumers to have several accounts.

f. For the A_C relationship, the (min, max) constraint on the CUSTOMER side would be (1,N), as every customer must have at least one account. The ACCOUNT side would remain (M,N) since an account can be joint. For the L_C relationship, the (min, max) constraint on the CUSTOMER side would be (0,2), reflecting that a customer can have zero to two loans. The LOAN side would remain (M,N) since a loan can involve multiple customers. For the LOANS relationship, the (min, max) constraint on the BANK_BRANCH side would be (1,1000), as a branch must have at least one loan but cannot exceed 1000 loans.

**#Solution-2**:

**Student to SocialSecurityCard:**

Every student gets a unique Social Security card, and every Social Security card is distinct from the others. So the cardinality is 1:1

**Student to Teacher:**

Teachers are able to instruct many students, while students are able to have several teachers (one for each class). Here cardinality is M:N

**ClassRoom to Wall:**
There are several walls in each classroom, but each wall is a component of a single classroom.
Cardinality: 1:N

**Country to CurrentPresident: Cardinality: 1:1**
Every nation has a single president at a time, and each president is in charge of a single nation.

**Course to TextBook: Cardinality: M:N**
Textbooks may be used for more than one course, and some courses may require more than one.

**Item (that can be found in an order) to Order: Cardinality: M:N**
Items may be found in more than one order, and orders may contain more than one item.

**Student to Class:(N:M)**

A student may register for more than one class, and more than one student may be enrolled in one class.

**Class to Instructor: N:M**
A teacher can teach more than one class, but there is only one main instructor per class.

**Instructor to Office: 1:1**
An instructor may use multiple offices (for example, in different buildings or campuses), but each office is assigned to one instructor at a time.

**Ebay auction item to Ebay bid: 1:N**
Each eBay auction item can have multiple bids placed on it, but each bid is made on one specific auction item.

**#Solution-3:**
**Entities and Attributes:**

1. Computer: Model Number, Manufacturer, Purchase Date, SerialNumber
   Key - Computer ID
2. Laptop/Desktop: CPU Type, Storage Capacity, RAM Size
   Key - Serial Number.
3. Accessory: AccessoryID, Type, Brand, Model
   Key - Accessory ID.
   Subtypes - Keyboard, Mouse, Monitor.
4. Software: SoftwareID, Name, Version, License Type

Key - Software ID.

5. Operating System: KernelVersion, License Key, Supported File
   Key: Operating System ID.
6. Component: Component ID, Type, Manufacturer, Model
   Key: Component ID.

**Cardinality Constraints:**

- An Accessory can be a Keyboard, Mouse, or Monitor (1 to 1), meaning every accessory is defined as one of these types.
- A Computer must have 1 to many Components (1 to N), as a computer cannot function without internal components like memory or a video card.
- A Computer can have 0 to many Software installed on it (0 to N), indicating that a computer might not have any additional software or could have multiple software applications.
- A Laptop or Desktop can be sold with 0 to many Accessories (0 to N), meaning it's optional to have accessories sold with them.
- An Operating System must support 1 to many Components (1 to N), because an operating system must be able to interact with at least one or more internal components to operate.

**Description:**

The EER diagram describes a system where Computers are categorized as either Laptops or Desktops. Each computer can have a variety of Software installed, with Operating Systems being a necessary subtype of software. Computers are composed of several Components, which include Memory, Video Cards, and Sound Cards. These components are essential for the operation of the computer and are supported by the Operating System. Accessories associated with Computers can be such as Keyboards, Mice, and Monitors, which can be sold with the computer but are not mandatory. Because each entity type has a unique identity, every object within the company's systems may be tracked and managed separately.