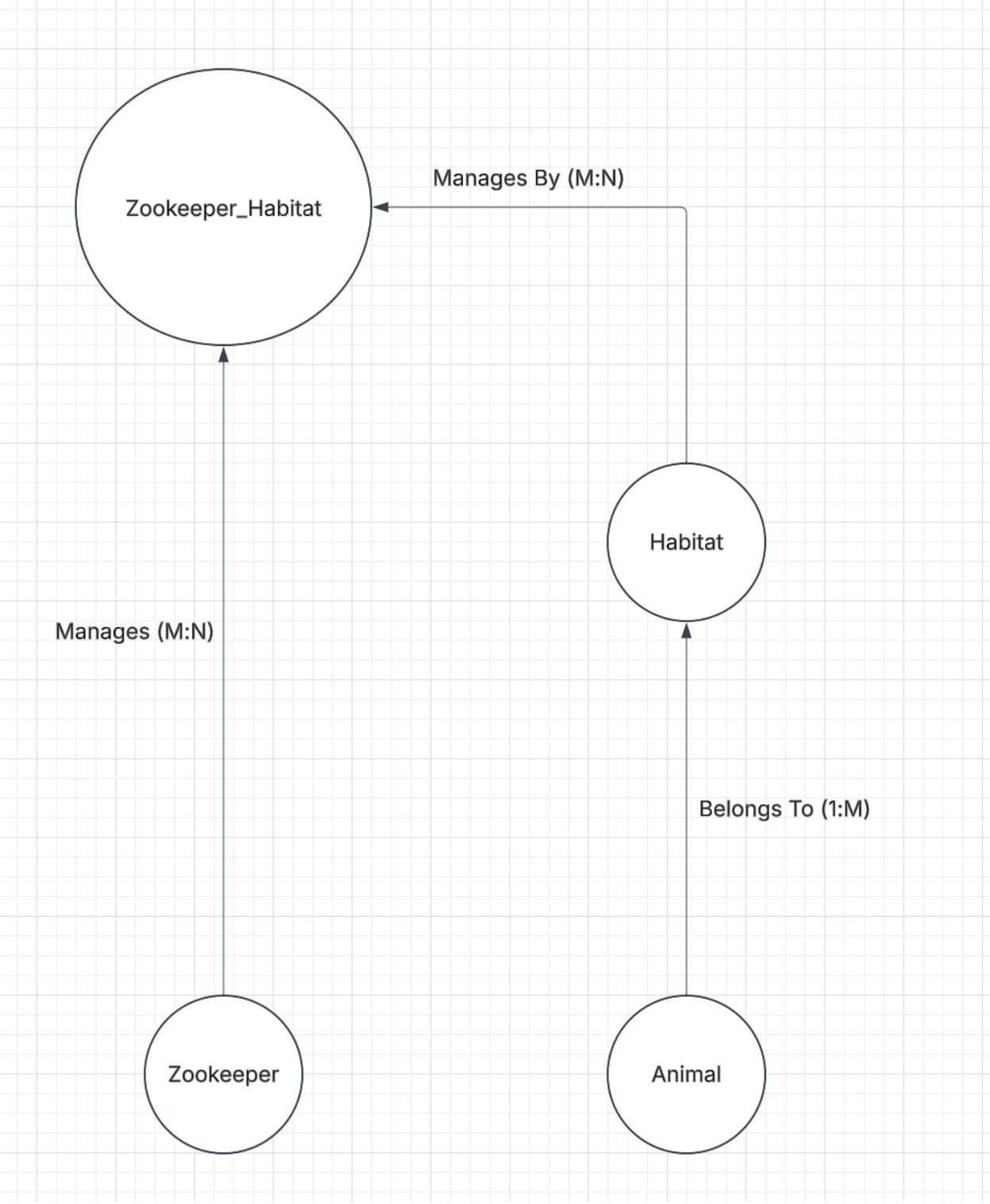
HW4 Part2

ER Diagram for Zoo Database

Muyang Cheng



1. Entities & Attributes

This zoo database has three main parts: Animal, Habitat, and Zookeeper. There is also a Zookeeper_Habitat table to connect zookeepers and habitats since they have a many-to-many relationship.

Entities:

1. Animal
    1. AnimalID (Primary Key, Unique)
    2. Name (Animal's name)
    3. Species (Type of animal)
    4. Age (Must be 0 or more)
    5. HealthStatus ("Healthy" or "Sick")
    6. HabitatID (Foreign Key → Habitat)
2. Habitat
    1. HabitatID (Primary Key, Unique)
    2. HabitatName (Name of the habitat)
    3. Climate (Hot, Cold, etc.)
    4. Size (Area size)
3. Zookeeper
    1. ZookeeperID (Primary Key, Unique)
    2. Name (Zookeeper's name)
    3. PhoneNumber (Unique contact number)
4. Zookeeper_Habitat (Link Table for Many-to-Many Relationship)
    1. ZookeeperID (Foreign Key → Zookeeper)
    2. HabitatID (Foreign Key → Habitat)

2. Relationships

1. Animal and Habitat (One-to-Many)
    1. Each Animal belongs to one Habitat, but a Habitat can have many Animals.
    2. (1:M) Relationship.
    3. HabitatID in Animal is a Foreign Key.
2. Zookeeper and Habitat (Many-to-Many)
    1. A Zookeeper manages multiple Habitats, and a Habitat can be managed by multiple Zookeepers.
    2. (M:N) Relationship.
    3. The Zookeeper_Habitat table records which zookeeper manages which habitat.

3. Constraints

1. Primary Keys: AnimalID, HabitatID, ZookeeperID must be unique.
2. Foreign Keys: HabitatID in Animal, ZookeeperID and HabitatID in Zookeeper_Habitat.
3. Data Rules:
    1. Age must be 0 or more.
    2. PhoneNumber must be unique.
    3. HealthStatus can only be "Healthy" or "Sick".

4. Why This Deisgn?

1. Animal-Habitat (1:M)
    1. Each animal has a habitat.
    2. A habitat can hold many animals.
2. Zookeeper-Habitat (M:N)
    1. One zookeeper can manage multiple habitats.
    2. One habitat can have multiple zookeepers.
3. Data Rules
    1. Makes sure values are correct (e.g., no negative age, no duplicate phone numbers).
    2. Ensures relationships stay valid using foreign keys.