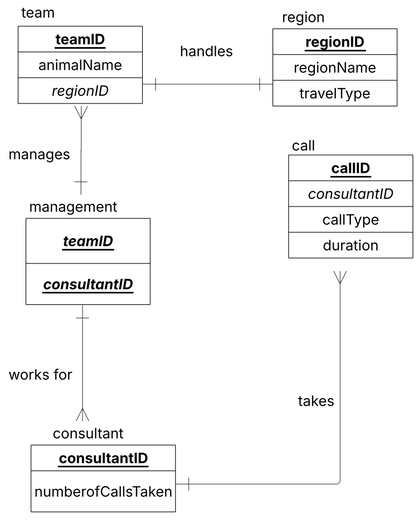
1. Travel Consulting Company

Assumptions:
1. I'm assuming that the entities "team" and "consultant" have a many-to-many relationship since one team has many consultants and a consultant can be on many teams as indicated by the prompt. Thus, I created a bridging table "management" to account for that.
2. I'm assuming that each customer phone call is taken by only one consultant, thus the one-to-many relationship between the two entities (a consultant can take many calls). I also gave each call a unique identifier in the attribute callID.



2. DVD Rental Company

Assumptions:
1. I made "manager" and "staff" separate entities because I wasn't sure how to showcase the one-on-one relationship between managers and branches otherwise. I gave the manager entity a unique managerID as a primary key to identify each manager. The manager entity also has a one-on-one relationship with the staff entity since a manager is just another version of a staff member.
2. I'm also combining the entities "movie" and "DVD" into "dvd_in_stock" itself since I can show all the necessary relationships without having to create separate entities.
3. Because the entities "customer" and "branch" have a many-to-many relationship (a member can be registered at multiple branches and a branch has many members), I created a bridging table "registeration". This is why the attribute "dateRegistered" which represents the date of a customer's registeration at a branch is in the registeration entity and not in the customer one since that wouldn't be possible.