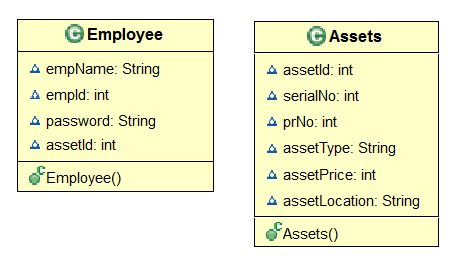
**Problem Description**

Refer to the below class diagram:



**Note:** Since mapping is not provided among classes, assetId has been used in the Employee class to show the relationship between employee and associated asset details. We need to use the same assetId to provide the relation between employee and asset details.

* Create the below entities as part of the Asset Management System
  + Assets class with an appropriate mapping to table Assets\_Table
  + Employee class with an appropriate mapping to table Employee\_Table
* Implement CRUD Operations:
  + Create:
    - Create a new Employee object and add the details into the database as empId as the id field
    - Create a new Assets object and add the Asset details into the database with assetId as the id of the newly persisted record
  + Read:
    - Retrieve the Assets details using get() or load() for the appropriate id given
    - Retrieve the Assets details with maximum price.
    - Count the total no. of employees from the Employee table.
    - Retrieve all the Employee details using HQL and order by name in ascending order.
    - Retrieve the Employee details and Assets details for given employeeId.
    - Find average assetPrice of each assetLocation using NamedQuery.
  + Update:
    - For the given empId, update the password with the new value.
  + Delete:
    - Delete the Employee details of the given empId and the associated Asset details from the Asset table
* Create a Java class user interface to demonstrate the above CRUD operations. Maintain the database tables appropriately.