1. Map the following EER models to relational model
   a. **Bank Database**
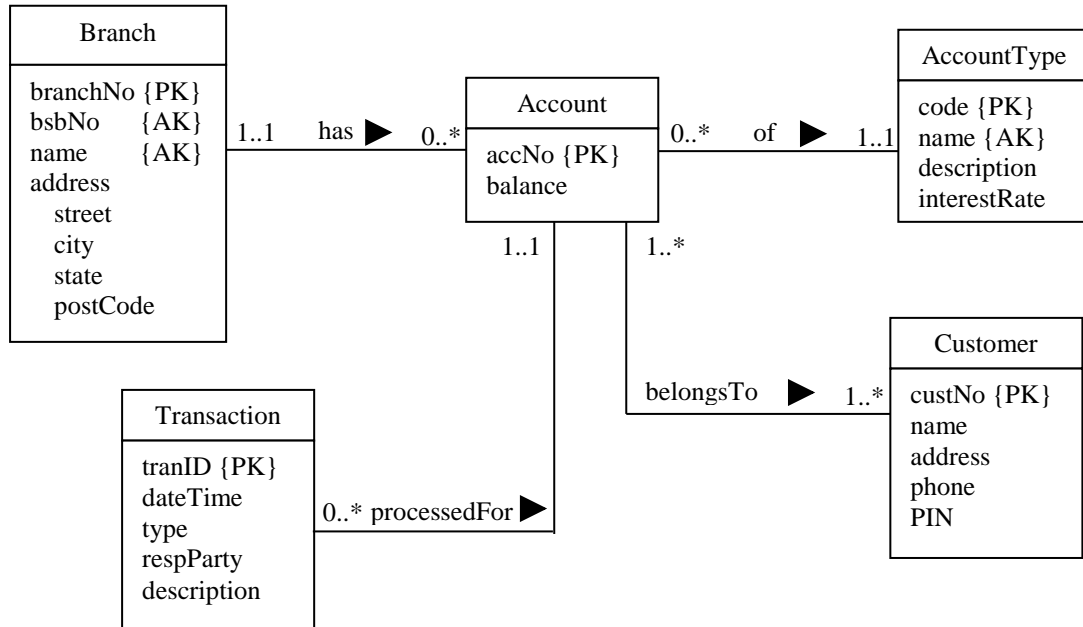
Answer: (basically use rules for binary relationship –
1 to *: for Branch & Account, and Account & AccountType, and Account & Transaction; --- add Foreign Key at * side
* to *: for Account & Customer --- create a "Relationship" relation, add 2 foreign keys)

Branch (branchNo, bsbNo, name, street, city, state, postcode)
Primary Key branchNo

Account (accNo, balance, branchNo, code)
Primary Key accNo
Foreign Key branchNo references Branch(branchNo)
Foreign Key code references AccountType (code)

AccountType (code, name, description, interestRate)
Primary Key code

Customer (custNo, name, address, phone, PIN)
Primary Key custNo

Transaction (tranID, dateTime, type, respParty, description, accNo)
Primary Key tranID
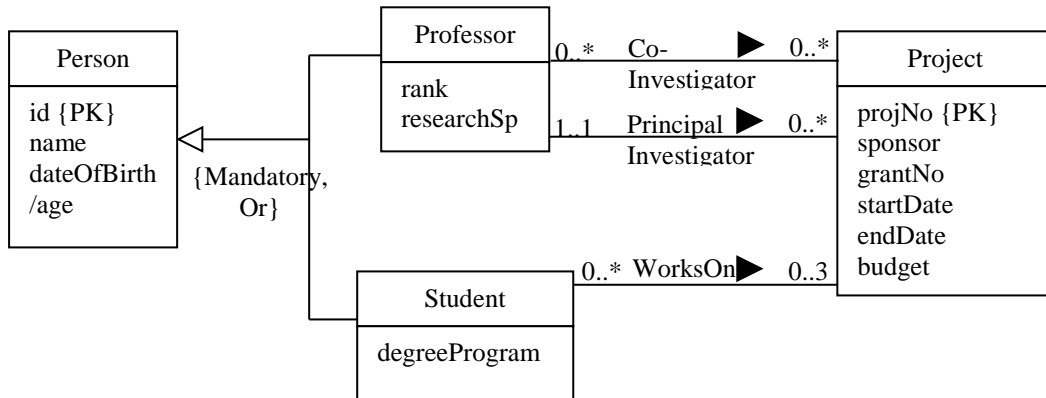Foreign Key accNo references Account (accNo)

RelationAccountCustomer (accNo, custNo)
Primary Key accNo, custNo
Foreign Key accNo references Account (accNo)
Foreign Key custNo references Customer (custNo)

b. **University Database**

Answer: (basically use combination of rules:
Rule 1 is: for super/subclass, Mandatory & Disjoint: create multi- relations, with 1 relation for each combined superclass/subclass;
Rule 2 is: * to * relation type (for Professor coinvestigates Project, and Student – Project): create "Relationship" relation and two foreign keys
Rule 3 is: 1 to * relation type (for Professor principalinvestigates Project): create a foreign key at project side
)

ProfessorPerson (id, name, dateOfBirth, rank, researchSp)
Primary Key id

StudentPerson (id, name, dateOfBirth, degreeProgram)
Primary Key id

Project (projNo, sponsor, grantNo, startDate, endDate, budget, id)
Primary Key projNo
Foreign Key id references ProfessorPerson (id)

CoInvestigate (id, projNo)
Primary Key id, projNo
Foreign Key id references ProfessorPerson (id)
Foreign Key projNo references Project (projNo)

WorksOn (id, projNo)
Primary Key id, projNo
Foreign Key id references StudentPerson (id)
Foreign Key projNo references Project (projNo)