Name \_\_\_\_\_\_\_Mao, Keying Linxia Li\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_Nov, 8, 2022\_\_\_\_\_\_\_

(last name, first name)

NYU ID: \_\_\_\_\_\_ km5580 ll4764\_\_\_\_\_\_\_\_

Course Section Number: \_\_DS-GA-2433-001\_\_\_\_\_

**Project Part 1**

**Total in points** (100 points total): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Professor’s Comments:**

Affirmation of my independent effort:

Keying Mao

Linxia Li

Account has a M:N relationship with BillingAccount. For Account entity, it includes all necessary information, such as account name, company code and location, and we assume that account name is the primary key . Since each account could have multiple billing addresses(FSA, Life, A&H and only use 1), an attribute “ business type” is used in their relationship to connect them. Account has a M:N relation with itself, in their relation, which is AccountRelation in UML table, RelationType is used to identify the type of relationship, such as “participate” or “belong to”.Account has a M:N relationship with AccountAdmin, which means that each account has administrators for each address(FSA, Life, A&H). Some accounts only have one administrator, since multiple businesses share the same address. Account can have multiple workers, and each administrator could work in different places. The AccountAdmin’s primary key is composed of LastName, FirstName, and MiddleName. Other personal information is also included in that entity. Account connects to AccountAdmin through BillingAdmin. In BillingAdmin, the attribute of AdminRole is used to identify the type of billing address. For example, if an administrator controls FSA, he’s AdminRole is “FSA”. Account has a 1: N relationship with AccountMember, which means multiple members belong to one account. In AccountMember, the attribute StartDate will record the date that the member starts to work in that account. It will include all past work experience even if the member changes the workplace. Account has a 1:1 relationship with AccountAlias. Overall, AccountAlias is a deep copy and a combination of details in account. The records of FSA, Life and A&H invoices exist in the AccountAlias table with each original record. In AccountAlias, the attribute of AliasSource is used to identify the type(FSA, Life and A&H invoices). The Account has a n:1 relationship with the Company. And CompanyCode is the primary key, the Account also uses CompanyCode to refer to Company.

From an Associate case perspective, firstly, the Account has M to N relationship with ManagerContract. Associate has 1:N relationship between Associate and ManagerContract.

Associate has the name, official, and date. ManagerContract has attributes such as State, Sitcode, writing number, and issueDate. Account has attributes such as AccountName, Address, etc. Thus, we can verify the case that a company associate can use a Sitcode of ManagerContract to service many accounts. And we have Type in the relation between Account and ManageContract, which helps show that Account has different types of associates in the area of original servicing, servicing, and assisting. Besides, we can also confirm that each associate can have many ManagerContract with different WritingNumber. For example, associate Keith can have many ManagerContracts like Cliff, Rob, David, and Gernald. ManagerContract such as David has state of GA and Sitcode0. Since the case illustrates that there will be two manners Associate maintains contract premium: commission or production credits. Therefore, we add an attribute of manners attached by the relationship of sold/maintained between Associate and ContractPremium. Finally, Associate has M:N relationship with itself. And it has the attribute of AssociateType for associate broker and associate recruiter in the self-connected relationship.

Contract has a 1: N relationship with ContractBenefit. The Contract has ContractNumber and PolicyPayer attributes. ContractNumber is the primary key. Each contract has multiple benefits. The attributes in ContractBenefits includes BenefitID, PlanName, BenefitPolity, Seriesame and Rider. In that case, the details about policy, publish year and riders are clear. ContractBenefit has a 1:N relationship with ContractPremium. Specifically, the ContractPremium has PremiumCode and PremiumType attributes, PremiumCode is the primary key.

From Customer perspective, it firstly has M:N relationship with itself. Thus, it satisfies the example that Daddy buy the insurance for other customers such as his family. Besides, Customer has M:N relationship with Contract. Specifically, since the Contract has ContractNumber and Policypayer attributes. And there is a relation of ContractingPartyRole between Customer and Contract. Thus, for instance, Daddy is the policy owner of the health insurance contract. Its policy payer is the trust fund. Third, Customer has M:N relationship with ContractBenefit. We regard ContractBenefit as weak entity since it doesn’t have any primary key here. ContractBenefit has attributes such as BenefitName, BenefitPolicy, etc. Thus, Daddy’s health insurance can benefit his son and family. His son has an accident plan to benefit his family. Then, Customer has M:N relationship with Account as well. So each account such as Dana’s dry cleaning can have multiple employees like Keith, and Keith can work Scott’s garage. Lastly, customer has M:N relationship with Associate. For example, Walt can be the associate but he is also the customer of the policy plan. After he passed away, his wife and son will be beneficiaries.

