

## არსების (სუბიექტების) და ატრიბუტების მოდელირება

E-R მოდელის ძირითადი კონსტრუქციებია არსები ((სუბიექტები), კავშირები და ატრიბუტები. როგორც ნაჩვენებია ნახაზზე 2-2, მოდელი მრავალი ვარიაციის შექმნის საშუალებას იძლევა თითოეული ამ კონსტრუქციისთვის. E-R მოდელის სიმდიდრე საშუალებას აძლევს დამპროექტებლებს ზუსტად და გამომსახველობით მოახდინონ რეალური სიტუაციების მოდელირება, რაც მოდელის პოპულარობას განმარტავს.

### არსები

არსი (სუბიექტი) არის პირი, ადგილი, ობიექტი, ღონისძიება (მოვლენა) ან კონცეფცია მომხმარებლის გარემოში, რომლის შესახებ ორგანიზაციას სურს შეინარჩუნოს/შეინახოს/დაამუშაოს მონაცემები. ამრიგად, არსი არის არსებითი სახელი.

ქვემოთ მოყვანილია ტიპის არსების (სუბიექტების) მაგალითები:

**პირი:** თანამშრომელი, სტუდენტი, პაციენტი

**ადგილი:** მაღაზია, საწყობი, სახელმწიფო

**ობიექტი:** მანქანა, შენობა, ავტომობილი

**ღონისძიება:** რეალიზაცია, რეგისტრაცია, განახლება

**კონცეფცია:** ანგარიში, კურსი, სამუშაო ცენტრი

*Person:* EMPLOYEE, STUDENT, PATIENT

*Place:* STORE, WAREHOUSE, STATE

*Object:* MACHINE, BUILDING, AUTOMOBILE

*Event:* SALE, REGISTRATION, RENEWAL

*Concept:* ACCOUNT, COURSE, WORK CENTER

### არსის (სუბიექტის) ტიპი - არსის (სუბიექტის) ეგზემპლარი

არსებობს მნიშვნელოვანი განსხვავება არსის ტიპებსა და არსის ერთეულებს (ეგზემპლარს) შორის. არსის ტიპი არის არსების ერთობლიობა, რომლებსაც აქვთ საერთო მახასიათებლები ან თვისებები. E-R მოდელის თითოეულ ტიპის არსს ენიჭება სახელი. იმის გამო, რომ სახელი წარმოადგენს ნივთების კრებულს (ან ერთობლიობას), ის ყოველთვის მხოლოდ ერთ ფორმაშია. საყოველთაოდ მიღებულია არსების ტიპების სახელების გამოსახვა ლათინური ანბანის (ინგლისური) დიდი ასოებით. E-R დიაგრამაში არსის სახელი მოთავსებულია უჯრის შიგნით, რომელიც წარმოადგენს არსის ტიპს (იხ. ნახ. 2-1). **არსის ეგზემპლარი** არის არსის ტიპის ერთი შემთხვევა. ნახ. 2-3 ასახავს განსხვავებას არსის ტიპსა და მის ორ ეგზემპლარს შორის. არსის ტიპი აღწერილია მონაცემთა ბაზაში მხოლოდ ერთხელ (მეტამონაცემების გამოყენებით), ხოლო ამ ტიპის ობიექტის მრავალი შემთხვევა შეიძლება წარმოდგენილი იყოს მონაცემთა ბაზაში შენახული მონაცემებით. მაგალითად, უმეტეს ორგანიზაციებში არის ერთი EMPLOYEE არსის ტიპი, მაგრამ მონაცემთა ბაზაში შეიძლება ინახებოდეს ასობით (ან თუნდაც ათასობით)

ამ არსის ტიპის შემთხვევა. ჩვენ ხშირად ვიყენებთ ტერმინს „არსი“, ვიდრე „არსის ეგზემპლარს“, როდესაც მნიშვნელობა ნათელია ჩვენი განხილვის კონტექსტიდან.

Entity type: EMPLOYEE			
Attributes	Attribute Data Type	Example Instance	Example Instance
Employee Number	CHAR (10)	642-17-8360	534-10-1971
Name	CHAR (25)	Michelle Brady	David Johnson
Address	CHAR (30)	100 Pacific Avenue	450 Redwood Drive
City	CHAR (20)	San Francisco	Redwood City
State	CHAR (2)	CA	CA
Zip Code	CHAR (9)	98173	97142
Date Hired	DATE	03-21-1992	08-16-1994
Birth Date	DATE	06-19-1968	09-04-1975

**FIGURE 2-3** Entity type EMPLOYEE with two instances

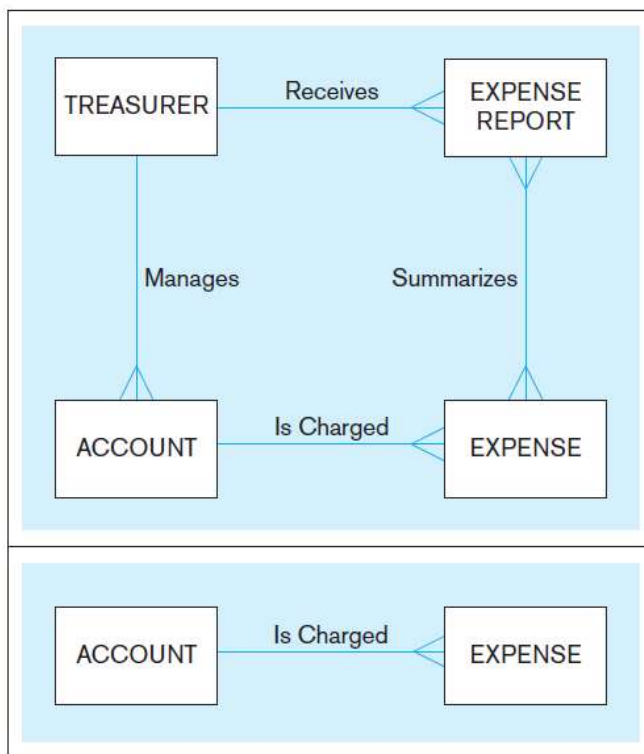
*არსის ტიპი სისტემაში მონაცემების შეტანის, გამოტანის ან მომხმარებლის წარმოდგენის მიმართებაში*

E-R დიაგრამების ხატვის პროცესში გავრცელებულ შეცდომს წარმოადგენს მონაცემთა არსების (სუბიექტების) აღრევა საერთო საინფორმაციო სისტემების მოდელის სხვა ელემენტებთან. ეს ხშირია მაშინ თუ ისინი უკვე იცნობენ მონაცემთა პროცესის მოდელირებას (მაგალითად, მონაცემთა ნაკადის დიაგრამა). ასეთი დაბნეულობის თავიდან ასაცილებლად მარტივი წესი არსებობს - რომ მონაცემთა ჭეშმარიტ არსს აქვს მრავალი შესაძლო ეგზემპლარი, რომელთაგან თითოეულს აქვს განმასხვავებელი მახასიათებლები, ასევე ერთი ან რამდენიმე სხვა აღწერითი მონაცემი.

განვიხილოთ ნახ. 2-4 ა, რომელიც შეიძლება შედგენილი იყოს რომელიმე კლუბის ხარჯების აღრიცხვის სისტემისთვის საჭირო მონაცემთა ბაზაში. ამ სიტუაციაში, კლუბის ხაზინადარი მართავს ანგარიშებს, იღებს ხარჯების ანგარიშებს და აფიქსირებს ხარჯების ტრანზაქციებს თითოეული ანგარიშის შესაბამისად. ამასთან უნდა განვსაზღვროთ საჭიროა თუ არა თვალყური ვადევნოთ ხაზინადარის (TREASURER არსის ტიპი) მონაცემებს და მის ზედამხედველობას ანგარიშებზე (კავშირი Manages) და ანგარიშების მიღებას (კავშირი Receives). ხაზინადარი არის პირი, რომელიც შეიყვანს მონაცემებს ანგარიშებისა და ხარჯების შესახებ და იღებს ხარჯების ანგარიშებს. ეს არის მონაცემთა ბაზის მომხმარებელი. იმის გამო, რომ არსებობს მხოლოდ ერთი ხაზინადარი, TREASURER მონაცემების შენახვა საჭირო არ არის. გარდა ამისა უნდა განისაზღვროს საჭიროა თუ არა არსი EXPENSE REPORT (ხარჯების ანგარიში). იმის გამო, რომ ხარჯების ანგარიში გამოითვლება ხარჯების გარიგებებიდან და ანგარიშის ნაშთებიდან, ეს არის მონაცემთა ბაზაში მონაცემების მოპოვებისა და ხაზინადარის მიერ მიღებული ინფორმაციის შედეგი. მიუხედავად იმისა, რომ დროთა განმავლობაში ხაზინადარს გადაეცემა ხარჯების შესახებ მრავალი შემთხვევის აღმწერი მონაცემები,

რომლებიც საჭიროა ანგარიშის შინაარსის გამოსათვლელად ყოველ ჯერზე, ეს ინფორმაცია უკვე წარმოდგენილია ACCOUNT და EXPENSE არსების ტიპებით.

**FIGURE 2-4** Example of inappropriate entities  
(a) System user (Treasurer) and output (Expense Report) shown as entities



(b) E-R diagram with only the necessary entities

კიდევ ერთი განმარტება იმის გასაგებად, თუ რატომ შეიძლება იყოს შეცდომა ERD ნახაზზე 2-4 ა, არის კავშირების სახელების ხასიათი, „Receives“ (მიღება) და „Summarizes“ (შეჯამება). კავშირების ეს სახელები განეკუთვნება ბიზნეს საქმიანობას, რომელიც გადასცემს ან გარდაქმნის მონაცემებს, და არა უბრალოდ ერთი სახის მონაცემების სხვა სახის მონაცემებთან ასოცირებას. ნახ. 2-4 ბ-ში მოცემული მარტივი E-R დიაგრამა გვიჩვენებს არსებს და კავშირს, რომელიც საკმარისი იქნება კლუბის ხარჯების სისტემის მოსაგვარებლად, როგორც აღწერილია აქ.

### ძლიერი და სუსტი სახეობის არსის ტიპები

ორგანიზაციაში იდენტიფიცირებული ძირითადი არსების უმეტესობა კლასიფიცირდება როგორც **ძლიერი სახეობის არსის ტიპები**. ძლიერი არსი (სუბიექტი) არის ის არსი, რომელიც არსებობს სხვა ტიპის არსისაგან (სუბიექტისაგან) დამოუკიდებლად (მონაცემთა მოდელირების ზოგი პროგრამული უზრუნველყოფა, სინამდვილეში, იყენებს ტერმინს დამოუკიდებელი არსი/სუბიექტი). მაგალითებად გამოდგება STUDENT (სტუდენტი), EMPLOYEE (თანამშრომელი), AUTOMOBILE (ავტომობილი). ძლიერი ტიპის არსებს ყოველთვის აქვთ უნიკალური მახასიათებელი (ე.წ. იდენტიფიკატორი) - ეს არის ატრიბუტი ან ატრიბუტების კომბინაცია, რომელიც ცალსახად გამოყოფს ამ არსის თითოეულ მოვლენას (ეგზემპლარს).

ამის საპირისპიროდ, **სუსტი სახეობის არსის ტიპი** არის ისეთი არსის ტიპი, რომლის არსებობა დამოკიდებულია სხვა არსზე (სუბიექტზე). (მონაცემთა მოდელირების ზოგი პროგრამა, სინამდვილეში, იყენებს ტერმინს დამოკიდებული არსი/სუბიექტი.) E-R დიაგრამაში სუსტი ტიპის არსს (სუბიექტს) არ აქვს ბიზნესისთვის მნიშვნელობა იმ არსის (სუბიექტის გარეშე), რომელზეც ის დამოკიდებულია. არსს, რომელზეც დამოკიდებულია სუსტი არსი ეწოდება **საიდენტიფიკაციო მფლობელი** (ან მოკლედ უბრალოდ მფლობელი). სუსტი არსს, როგორც წესი, არ აქვს საკუთარი იდენტიფიკატორი. საერთოდ, E-R დიაგრამაზე, სუსტ არსს აქვს ატრიბუტი, რომელიც ნაწილობრივ იდენტიფიკატორს წარმოადგენს. შემდგომი დაპროექტების ეტაპზე სუსტი არსისთვის იდენტიფიკატორის ჩამოყალიბდება ნაწილობრივი იდენტიფიკატორისა და მისი მფლობელის იდენტიფიკატორის შერწყმით ან სუროგატი იდენტიფიკატორის ატრიბუტის შექმნით ხდება.

სუსტი არსის მაგალითი, საიდენტიფიკაციო კავშირით, ნაჩვენებია ნახ. 2-5. EMPLOYEE არის ძლიერი ტიპის არსი, იდენტიფიკატორით Employee ID (თანამშრომელი ID) (იდენტიფიკატორი აღნიშნულია ატრიბუტის ხაზგასმით). DEPENDENT არის სუსტი ტიპის არსი, რასაც ორმაგ ხაზოვანი მართკუთხედი მიუთითებს. სუსტი არსის ტიპსა და მის მფლობელს შორის კავშირს საიდენტიფიკაციო კავშირი ეწოდება. ნახაზზე 2-5, Carries არის საიდენტიფიკაციო კავშირი (მითითებულია ორმაგი ხაზით). ატრიბუტი Dependent Name ნაწილობრივ იდენტიფიკატორს წარმოადგენს. Dependent Name (დამოკიდებული სახელი) არის კომპოზიციური ატრიბუტი, რომელიც შეიძლება დაიყოს კომპონენტებად (ამ საკითხს მოგვიანებით განვიხილავთ). ნაწილობრივ იდენტიფიკატორის აღსანიშნად გამოიყენება ორმაგი ხაზგასმა. შემდგომი დაპროექტების ეტაპზე, Dependent Name (დამოკიდებული სახელი) შეერწყმება Employee ID (თანამშრომელი ID) (მესაკუთრის იდენტიფიკატორი) და შექმნის სრულ იდენტიფიკატორს DEPENDENT-ისთვის. ძლიერი და სუსტი არსების წყვილების რამდენიმე დამატებითი მაგალითია: BOOK – BOOK COPY, PRODUCT – SERIAL PRODUCT.

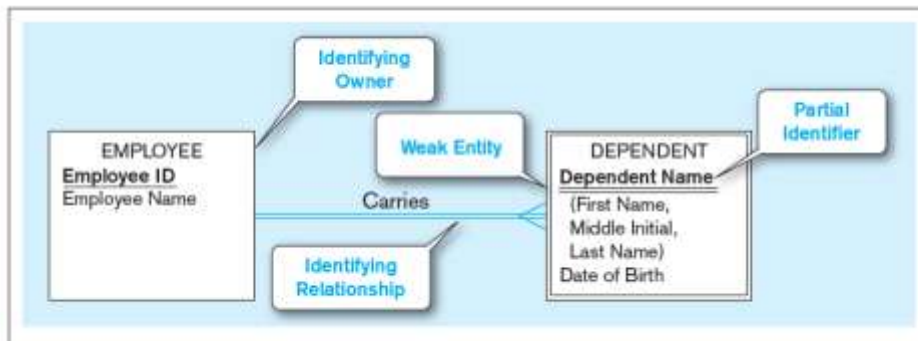


FIGURE 2-5 Example of a weak entity and its identifying relationship

### არსების (სუბიექტების) ტიპების სახელდებაა და განმარტება

მონაცემთა ობიექტების სახელდებისა და განსაზღვრის ზოგადი სახელმძღვანელო მითითებების გარდა, არსებობს რამდენიმე სპეციალური სახელმძღვანელო მითითება არსების ტიპების სახელდებისათვის, რომლებიც შემდეგში მდგომარეობს:

- **არსის (სუბიექტის) ტიპის სახელი** არის მხოლოდითი არსებითი სახელი (როგორიცაა CUSTOMER, STUDENT ან AUTOMOBILE); არსი არის პიროვნება, ადგილი, ობიექტი,

მოვლენა ან კონცეფცია, ხოლო სახელი არის არსის ტიპისთვის, რომელიც წარმოადგენს არსების ეგზემპლარების ერთობლიობას (მაგ., STUDENT წარმოადგენს სტუდენტებს ნანა ხაჩიძე, დავით დიდუბელი და ა.შ. ) ხშირია მრავლობითი ფორმის მითითება (შესაძლებელია E-R დიაგრამის თანმხლები CASE ხელსაწყო საცავში), რადგან ზოგჯერ E-R დიაგრამა უკეთესად იკითხება მრავლობითის გამოყენებით. მაგალითად, ნახ. 2-1, ჩვენ ვიტყვით, რომ მიმწოდებელმა შეიძლება უზრუნველყოს ITEMS (ნივთების) მოწოდება. იმის გამო, რომ მრავლობითი რიცხვი ყოველთვის არ იქმნება მხოლოდით არსებითი სახელზე s –ის დამატებით, უმჯობესია ზუსტი მრავლობითი ფორმის დოკუმენტირება.

- **არსის ტიპის სახელი უნდა იყოს სპეციფიკური** ორგანიზაციისთვის. ამრიგად, ერთმა ორგანიზაციამ შეიძლება გამოიყენოს არსის ტიპის სახელი CUSTOMER, ხოლო მეორე ორგანიზაციამ შეიძლება გამოიყენოს არსის ტიპის სახელი CLIENT (ეს ერთ-ერთი ამოცანა, მაგალითად, შეძენილი მონაცემების მოდელის გამართვისას). სახელი უნდა იყოს აღწერითი ყველასათვის ორგანიზაციაში და განსხვავებული ორგანიზაციის ყველა სხვა ტიპის არსების სახელისგან. მაგალითად, PURCHASE ORDER (შეკვეთის შესყიდვა) შეკვეთებისთვის განსხვავდება CUSTOMER ORDER (მომხმარებლის შესყიდვა) შესყიდვების შეკვეთისაგან. არსის ორივე ამ ტიპს არ შეიძლება დაერქვას ORDER.
- **არსის ტიპის სახელი უნდა იყოს ლაკონური და რაც შეიძლება ნაკლებსიტყვიანი.** მაგალითად, უნივერსიტეტის მონაცემთა ბაზაში, არსის ტიპის სახელი REGISTRATION (რეგისტრაცია) საკმარისი იქნება, თუ ის აღწერს სტუდენტის დარეგისტრირებას საგანზე. ხოლო სახელი STUDENT REGISTRATION FOR CLASS (სტუდენტების რეგისტრაცია კლასში), ზუსტია, მაგრამ ძალიან მრავალსიტყვიანია.
- შესაძლებელია **თითოეული ტიპის არსის სახელისთვის მიეთითოს აბრევიატურა, ან მოკლე სახელი**, და ეს საკმარისია E-R დიაგრამაში მისი გამოყენებისათვის; აბრევიატურები უნდა ემორჩილებოდეს ყველა იმავე წესს, როგორც არსების სრული სახელები.
- **ლონისძიების (მოვლენების) ტიპის არსებს (სუბიექტებს) უნდა დაერქვას ლონისძიების შედეგის და არა ლონისძიების აქტივობის ან პროცესის შესაბამისი სახელი.** მაგალითად, პროექტის მენეჯერის მიერ თანამშრომლის პროექტზე სამუშაოდ დანიშვნის შემთხვევაში შედეგად მიიღება „დანიშვნა“ (ASSIGNMENT), ხოლო თუ სტუდენტი დაუკავშირდა თავის ფაკულტეტის მრჩეველს გარკვეული ინფორმაციის მისაღებად არის „კონტაქტი“ (CONTACT).
- **ერთი და იგივე ტიპის არსებისათვის გამოყენებული სახელი უნდა იყოს იგივე ყველა E-R დიაგრამებზე, რომელზეც ჩანს ეს არსის ტიპი.** ამრიგად, სახელი რომელიც გამოიყენება არსის ტიპისათვის ორგანიზაციისთვის უნდა იყოს სტანდარტი ყველასათვის. ამასთან, არსის ზოგიერთ ტიპს ექნება მეტსახელი, ან ალტერნატიული სახელი, რომელიც სინონიმებია, რომლებიც გამოიყენება ორგანიზაციის სხვადასხვა ნაწილში. მაგალითად, არსის ტიპის ITEM–ს შეიძლება ჰქონდეს მეტსახელი MATERIAL (წარმოებისთვის) და DRAWING (პროექტირებისათვის). მეტსახელები მითითებულია მონაცემთა ბაზის შესახებ დოკუმენტაციაში, როგორიცაა CASE ინსტრუმენტის საცავში.

ასევე არსებობს არსების (სუბიექტების) ტიპების განმარტების გარკვეული კონკრეტული სახელმძღვანელო მითითებები, რომლებიც შემდეგია:

- არსის (სუბიექტის) ტიპის განმარტება ჩვეულებრივ იწყება ” *X არის. . .* ” ეს არის არსის ტიპის მნიშვნელობის ასახვის ყველაზე პირდაპირი და ცხადი გზა.

- არსის (სუბიექტის) ტიპის განმარტება უნდა შეიცავდეს დებულებას იმის შესახებ, თუ რა არის უნიკალური მახასიათებელი არსის (სუბიექტის) ტიპის თითოეული ეგზემპლარისათვის. ხშირ შემთხვევაში, იდენტიფიკატორის მითითება იურიდიული პირის ტიპისთვის ხელს უწყობს პირის იდენტიფიცირებას. ნახ. 2-4 ბ-ის მაგალითი: *”ზარჯი (expense) არის გადახდა ზოგიერთი საქონლის ან მომსახურების შეძენისთვის. ზარჯის დადგენა ხდება ჟურნალიში შემოსავლის ნომრით”*.
- არსის (სუბიექტის) ტიპის განსაზღვრებაში ნათლად უნდა იყოს განმარტებული, თუ არსის რომელი ეგზემპლარები შედიან და არ შედიან ამ ტიპის არსში; ხშირად, საჭიროა ჩამოვთვალოთ იმ არსების ტიპები, რომლებიც გამორიცხულია. მაგალითად, *”მომხმარებელი არის ფიზიკური პირი ან ორგანიზაცია, რომელმაც შეუკვეთა ჩვენი პროდუქტი ან ის, რომელსაც ჩვენ დავუკავშირდით ჩვენი პროდუქციის რეკლამირებისთვის ან პოპულარიზაციისთვის. მომხმარებელი არ მოიცავს პირებს ან ორგანიზაციებს, რომლებიც ყიდულობენ ჩვენს პროდუქტებს მხოლოდ ჩვენი მომხმარებლების, დისტრიბუტორების ან აგენტების საშუალებით”*.
- არსის (სუბიექტის) ტიპის განმარტება ხშირად მოიცავს აღწერილობას, თუ როდის იქმნება და იშლება კონკრეტული არსის ტიპის ეგზემპლარი. მაგალითად, წინა პუნქტში, მაგალითიდან შეგვიძლია დავასკვნათ მომხმარებელი აშკარად იქმნება, როდესაც პირი ან ორგანიზაცია აკეთებს თავის პირველ შეკვეთას; რადგან ამ განსაზღვრებაში სხვა რამ არ არის მითითებული, მომხმარებლის მონაცემები არასდროს არ წაიშლება, ან ის წაიშლება ზოგადი წესების საფუძველზე, რომლებიც მიეთითება მონაცემთა ბაზიდან მონაცემთა გასუფთავების შესახებ. განცხადებას იმაზე, თუ როდის უნდა წაიშალოს იურიდიული პირის ჩანაწერი (ეგზემპლარი) ზოგჯერ მოიხსენიება, როგორც არსის ტიპის შენარჩუნება. მომხმარებლის არსის ტიპის განმარტებისთვის შესაძლებელია წაშლის შესახებ განცხადების შექმნა: *”მომხმარებელი არარაა მომხმარებელი, თუ სამ წელზე მეტი ხნის განმავლობაში არ გააკეთა შეკვეთა.”*
- ზოგიერთი არსის ტიპისთვის, განმარტებაში უნდა იყოს მითითებული, როდის შეიძლება გარდაიქმნას ეგზემპლარი სხვა ტიპის არსის ეგზემპლარად. მაგალითად, განვიხილოთ სამშენებლო კომპანია, რომლისთვისაც პოტენციური მომხმარებლების მიერ მიღებული წინადადებები ხდება კონტრაქტი. ამ შემთხვევაში, სატენდერო წინადადება შეიძლება განისაზღვროს შემდეგნაირად: *”სატენდერო წინადადება არის ჩვენი ორგანიზაციის იურიდიული შეთავაზება კლიენტისთვის სამუშაოს შესრულების შესახებ. წინადადება იქმნება მაშინ, როდესაც ჩვენი კომპანიის ოფიცერი ხელს აწერს სატენდერო დოკუმენტს; სატენდერო წინადადება ხდება ხელშეკრულების ნიმუში, როდესაც მივიღებთ შეთავაზების ასლს, რომელსაც ხელს აწერს მომხმარებლის ოფიცერი.”*
- ზოგიერთი ტიპის არსისათვის, განმარტებამ უნდა მიუთითოს, თუ რა ისტორიის დაცვა ხდება ამ ტიპის არსის ეგზემპლართან მიმართებაში. მაგალითად, ნახაზი 2-1-ში ITEM- ის მახასიათებლები შეიძლება შეიცვალოს დროთა განმავლობაში და შეიძლება დაგვჭირდეს ინდივიდუალური ღირებულებების სრული ისტორიის და მათი მოქმედების დროის შენახვა. ისტორიის შენახვის შესახებ ასეთ განცხადებებს შეიძლება ჰქონდეს გავლენა იმაზე, თუ როგორ წარმოვადგენთ არსის ტიპს E-R დიაგრამაზე და საბოლოოდ, თუ როგორ ვინახავთ მონაცემებს არსის ეგზემპლარებისათვის.



## ატრიბუტები

არსის (სუბიექტის) თითოეულ ტიპს აქვს მასთან დაკავშირებული ატრიბუტების ნაკრები. **ატრიბუტი** არის არსის ტიპის თვისება ან მახასიათებელი, რომელიც ორგანიზაციის ინტერესს წარმოადგენს. ასევე შესაძლებელია ატრიბუტი ჰქონდეს ზოგიერთ ტიპის კავშირებისაც. ამრიგად, ატრიბუტი არის არსებითი სახელი. ქვემოთ მოცემულია რამდენიმე ტიპური არსი და მათთან დაკავშირებული ატრიბუტები:

STUDENT (სტუდენტი)      Student ID, Student Name, Home Address, Phone Number  
(სტუდენტის ID, სტუდენტის სახელი, სახლის მისამართი, ტელეფონის ნომერი)

AUTOMOBILE (ავტომობილი)      Vehicle ID, Color, Weight, Horsepower (ავტომობილის ID, ფერი, წონა, ცხენის ძალა)

ატრიბუტების დასახელების აღნიშვნისას პირველ ასოდ იყენებენ დიდ ასოებს, რასაც მცირე ასოები მოყვება. თუ ატრიბუტის სახელი ერთზე მეტი სიტყვისგან შედგება, სიტყვებს შორის იყენებენ ინტერვალს და თითოეულ სიტყვას იწყება დიდი ასოთი, მაგალითად, Employee Name (თანამშრომლის სახელი) ან Student Home Address (სტუდენტის სახლის მისამართი). E-R დიაგრამებში (უმეტესში) ატრიბუტი წარმოდგება მისი სახელის განთავსებით მის მიერ აღწერილ არსში. ატრიბუტები ასევე შეიძლება დაკავშირებულნი იყვნენ იყოს კავშირებთანაც. უნდა გავითვალისწინოთ, რომ ატრიბუტი დაკავშირებულია ზუსტად ერთ არსთან ან კავშირთან.

ყურადღება მივაქციეთ ნახ. 2-5, იმ ფაქტს რომ DEPENDENT-ის ყველა ატრიბუტი მხოლოდ თანამშრომლის დამოკიდებულების მახასიათებლებია და არა უშუალოს თანამშრომლის. ტრადიციული E-R აღნიშვნით, არსის ტიპი (არა მხოლოდ სუსტი არსები, არამედ ნებისმიერი არსები) არ შეიცავს იმ არსების ატრიბუტებს, რომლებთანაც ისინი დაკავშირებულნი არიან (რასაც შეიძლება გარე ატრიბუტები ეწოდოს). მაგალითად, DEPENDENT არ შეიცავს რაიმე ატრიბუტს, რომელიც მიუთითებს, თუ რომელ თანამშრომელს უკავშირდება ეს დამოკიდებული თანამშრომელი. E-R მონაცემთა მოდელის ეს არასასურველი ფუნქცია შეესაბამება მონაცემთა ბაზის მონაცემთა გაზიარებულ თვისებას.

## სავალდებულო და არასავალდებულო ატრიბუტები

პოტენციურად თითოეულ არსს (ან არსის ტიპის ეგზემპლარს) გააჩნია მნიშვნელობა, რომელიც ასოცირდება ამ ტიპის არსის თითოეულ ატრიბუტთან. ატრიბუტს, რომელიც უნდა არსებობდეს თითოეული არსის ეგზემპლარისათვის ეწოდება **სავალდებულო** ატრიბუტი, ხოლო ატრიბუტს, რომელსაც შეიძლება არ ჰქონდეს მნიშვნელობა, **არასავალდებულო** ატრიბუტი ეწოდება. მაგალითად, ნახ. 2-6 გვიჩვენებს არსის STUDENT ეგზემპლარები მათი ატრიბუტის მნიშვნელობებით. ამ არსის ერთადერთი არასავალდებულო ატრიბუტი არის Major (ზოგიერთ სტუდენტს, ჯერ არ აურჩევიათ სპეციალობა); ამასთან, ორგანიზაციის წესებით, სტუდენტისათვის უნდა ჰქონდეს მნიშვნელობა ყველა დანარჩენ ატრიბუტს; ეს ნიშნავს იმას, რომ ჩვენ არ შეგვიძლია სტუდენტის შესახებ რაიმე მონაცემების შენახვა არსი STUDENT ეგზემპლარში, თუ არ არსებობს მნიშვნელობები ყველა სავალდებულო

ატრიბუტისთვის. E-R დიაგრამის სხვადასხვა აღნიშვნებში, თითოეული ატრიბუტის წინ შეიძლება გამოჩნდეს სიმბოლო, რომელიც მიუთითებს, სავალდებულოა ის (მაგ., \*) თუ არასავალდებულო (მაგ., O), ან სავალდებულო ატრიბუტები ჩაიწერება სქელი შრიფტით, ხოლო არასავალდებულო ატრიბუტები ჩვეულებრივი შრიფტით (ფორმატი, რომელსაც ამ ტექსტში ვიყენებთ); ხშირ შემთხვევაში, სავალდებულო ან არასავალდებულო მითითებულია დამატებით დოკუმენტაციაში. მნიშვნელობის არმქონე ატრიბუტს ნულოვან ატრიბუტს უწოდებენ. გარდა ამისა უნდა აღინიშნოს, რომ თითოეულ არსს აუცილებლად აქვს საიდენტიფიკაციო ატრიბუტი, რომელსაც შემდეგ განვიხილავთ, და დამატებით კიდევ ერთი ან მეტი ატრიბუტი. თუ შევცდებით შევქმნათ არსი, რომელსაც აქვს მხოლოდ იდენტიფიკატორი, ეს არსი სავარაუდოდ არ იქნება ლეგიტიმური. მონაცემთა ასეთ სტრუქტურაში შეიძლება უბრალოდ ინახებოდეს იურიდიული მნიშვნელობების ჩამონათვალი ზოგიერთი ატრიბუტისთვის, რომელიც უკეთ ინახება მონაცემთა ბაზის გარეთ.

Entity type: STUDENT				
Attributes	Attribute Data Type	Required or Optional	Example Instance	Example Instance
Student ID	CHAR (10)	Required	876-24-8217	822-24-4456
Student Name	CHAR (40)	Required	Michael Grant	Melissa Kraft
Home Address	CHAR (30)	Required	314 Baker St.	1422 Heft Ave
Home City	CHAR (20)	Required	Centerville	Miami
Home State	CHAR (2)	Required	OH	FL
Home Zip Code	CHAR (9)	Required	45459	33321
Major	CHAR (3)	Optional	MIS	

**FIGURE 2-6** Entity type STUDENT with required and optional attributes

### მარტივი და კომპოზიტიური ატრიბუტები

ზოგიერთი ატრიბუტი შეიძლება დაიყოს მნიშვნელოვან შემადგენელ ნაწილებად (დეტალური ატრიბუტები). გავრცელებული მაგალითია Name (სახელი), რომელიც ნაჩვენებია 2-5 ნახ.-ზე; სხვა მაგალითია Address (მისამართი), რომელიც ჩვეულებრივ შეიძლება დაიყოს შემდეგ კომპოზიტიურ ატრიბუტებად: Street Address, City, State, and Postal Code (ქუჩა, ქალაქი, სახელმწიფო და საფოსტო კოდი). კომპოზიტიური ატრიბუტი არის ატრიბუტი, რომელიც შედგება სხვადასხვა კომპონენტებისაგან, რომლებიც უფრო დეტალური ატრიბუტებია. ნახ. 2-7 გვიჩვენებს აღნიშვნას, რომელსაც ვიყენებთ ამ მაგალითში მოცემული კომპოზიტიური ატრიბუტებისთვის. ხატვის ინსტრუმენტების უმეტესობას არ აქვს კომპოზიტიური ატრიბუტების აღნიშვნის საშუალება, ასე რომ უბრალოდ ხდება ყველა კომპონენტის ნაწილის ჩამოთვლა. კომპოზიტიური ატრიბუტები უზრუნველყოფს მნიშვნელოვან მოქნილობას მომხმარებლებისთვის, რომლებსაც შეუძლიათ კომპოზიტიური ატრიბუტის მითითება როგორც ერთიანი ერთეული, ანდა ამ ატრიბუტის ცალკეული კომპონენტების მითითება. მაგალითად, მომხმარებელს შეუძლია მიმართოს მისამართს ან მიმართოს მის რომელიმე კომპონენტს, მაგალითად ქუჩის მისამართს. გადაწყვეტილება ატრიბუტის მის შემადგენელ



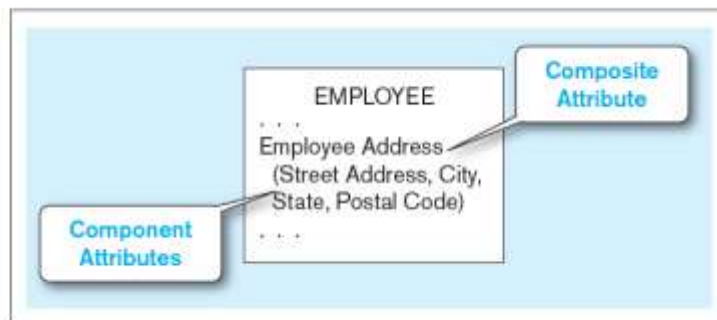
ნაწილებად დაყოფის შესახებ დამოკიდებულია იმაზე სურთ თუ არა მომხმარებლებს ამ ცალკეულ კომპონენტებზე მიმართვის საშუალება, ამიტომ მათ აქვთ ორგანიზაციული მნიშვნელობა. რა თქმა უნდა, დიზაინერი ყოველთვის უნდა შეეცადოს გაითვალისწინოს მონაცემთა ბაზის შესაძლო სამომავლო ვარიანტები.

**მარტივი** (ან ატომური) ატრიბუტი არის ატრიბუტი, რომლის დაყოფა შეუძლებელია ორგანიზაციისთვის მნიშვნელოვან მცირე კომპონენტებად. მაგალითად, AUTOMOBILE- სთან დაკავშირებული ყველა ატრიბუტი მარტივია: Vehicle ID, Color, Weight, Horsepower (ავტომობილის ID, ფერი, წონა და ცხენის ძალა).

### *ერთმნიშვნელოვანი და მრავალმნიშვნელოვანი ატრიბუტები*

ნახ. 2-6 გვიჩვენებს არსის ორი ეგზემპლარს მათი ატრიბუტის მნიშვნელობებით. თითოეული არსის ეგზემპლარისათვის ფიგურაში თითოეულ ატრიბუტს აქვს ერთი მნიშვნელობა. ხშირად ხდება, რომ არსებობს ატრიბუტი, რომელსაც შეიძლება ჰქონდეს ერთზე მეტი მნიშვნელობა მოცემული ეგზემპლარისათვის. მაგალითად, 2-8-ე EMPLOYEE არსის ტიპს აქვს ატრიბუტი სახელად Skill (უნარი), რომლის მნიშვნელობებით აღწერილია თანამშრომელის უნარი (ან უნარები). რა თქმა უნდა, ზოგიერთ თანამშრომელს შეიძლება ჰქონდეს ერთზე მეტი უნარი, როგორცაა PHP Programmer და C ++ Programmer. **მრავალმნიშვნელოვანი ატრიბუტი არის ატრიბუტი, რომელსაც შეუძლია ერთზე მეტი მნიშვნელობა მიიღოს მოცემული ერთეულის (ან კავშირის) ეგზემპლარისათვის.** ამ ტექსტში ჩვენ მიუთითებთ ატრიბუტის მრავალმნიშვნელოვნობაზე სახელწოდების ფიგურულ ფრჩხილებში ჩასმით, როგორც ეს ნაჩვენებია Skill ატრიბუტისთვის EMPLOYEE მაგალითზე 2-8 ნახ.-ზე. Microsoft Visio-ში, ასეთი ატრიბუტის არსში განთავსების შემდეგ, შეიძლება შეიცვალოს ეს ატრიბუტი (სვეტი), არჩეულ იქნას კოლექციის ჩანართი და აირჩეს ერთ – ერთი ვარიანტი (როგორც წესი არჩევანი იქნება MultiSet, მაგრამ სხვა ვარიანტი შეიძლება უფრო შესაფერისი იყოს კონკრეტული სიტუაციისთვის) E-R დიაგრამის აგების სხვა საშუალებებში ატრიბუტის სახელის შემდეგ გამოიყენება ვარსკვლავი (\*), ან გამოყენება დამატებითი დოკუმენტაცია მრავალმნიშვნელოვანი ატრიბუტის მისათითებლად.

**FIGURE 2-7** A composite attribute



მრავალმნიშვნელოვანობა და კომპოზიტიურობა განსხვავებული ცნებებია, თუმცა დამწყებთა მიერ ხშირად ხდება ამ ტერმინებს არევა. ზემოთ მოყვანილ მაგალითში მრავალმნიშვნელოვანი ატრიბუტი Skill (უნარი), შეიძლება მრავალჯერ შეგვხვდეს თითოეული თანამშრომლისთვის; თანამშრომლის სახელი და სახელფასო მისამართი ორივე

სავარაუდო კომპოზიციური ატრიბუტია, თითოეული მათგანი ერთხელ გვხვდება თითოეული თანამშრომლისთვის, მაგრამ მათ აქვთ კომპონენტი, უფრო ატომური ატრიბუტები, რომლებიც სიმარტივისთვის ნაჩვენებია 2-8 ნახაზზე.



**FIGURE 2-8** Entity with multivalued attribute (Skill) and derived attribute (Years Employed)

### შენახული და ნაწარმოები ატრიბუტები

ზოგიერთი ატრიბუტის მნიშვნელობა, რომელიც მომხმარებელს აინტერესებს, შეიძლება გამოითვალოს ან გამომდინარეობდეს სხვა მასთან დაკავშირებული ატრიბუტის მნიშვნელობიდან, რომელიც ინახება მონაცემთა ბაზაში. მაგალითად, დავუშვათ რომ ორგანიზაციისთვის, EMPLOYEE არსის ტიპს აქვს Date Employment ატრიბუტი. თუ მომხმარებლებმა უნდა იცოდნენ რამდენი წელია ადამიანია დასაქმებული, ეს მნიშვნელობა შეიძლება გამოითვალოს დასაქმების თარიღისა და დღევანდელი თარიღის გამოყენებით. მიღებული ატრიბუტი არის ატრიბუტი, რომლის მნიშვნელობების გამოთვლა შესაძლებელია დაკავშირებული ატრიბუტის მნიშვნელობებით (გარდა ამისა, მონაცემები არ არის მონაცემთა ბაზაში, მაგალითად, დღევანდელი თარიღი, მიმდინარე დრო ან სისტემის მომხმარებლის მიერ მოწოდებული უსაფრთხოების კოდი). ნაწარმოები ატრიბუტი E-R დიაგრამაში ატრიბუტის სახელის გარშემო კვადრატული ფრჩხილების გამოყენებით აღინიშნება, როგორც ეს ნაჩვენებია ნახ. 2-8- ში Years Employment ატრიბუტისთვის. ზოგიერთი E-R დიაგრამის ინსტრუმენტი იყენებს ატრიბუტის სახელის წინ დახრილი ხაზის (/) აღნიშვნას, რომ მიუთითოს, რომ იგი ნაწარმოებია.

ზოგიერთ სიტუაციაში, ატრიბუტის მნიშვნელობა შეიძლება გამომდინარეობდეს დაკავშირებული არსის ატრიბუტებიდან. მაგალითად, გაითვალისწინეთ Pine Valley Furniture Company-ში თითოეული მომხმარებლისთვის შექმნილი ინვოისი. შეკვეთა Total იქნება INVOICE არსის ატრიბუტი, რომელიც მიუთითებს დოლარის მთლიანი თანხით, რომელიც დარიცხულია მომხმარებლისთვის. შეკვეთის ჯამური ღირებულების გამოთვლა შესაძლებელია გაფართოებული ფასის მნიშვნელობების (ერთეულის ფასი გამრავლებული გაყიდული რაოდენობის) ჯამით სხვადასხვა ხაზის ერთეულებისათვის, რომლებიც ასახულია ინვოისზე. ისეთი მნიშვნელობების გამოთვლის ფორმულები, როგორიცაა ბიზნესის ერთ – ერთი წესი.

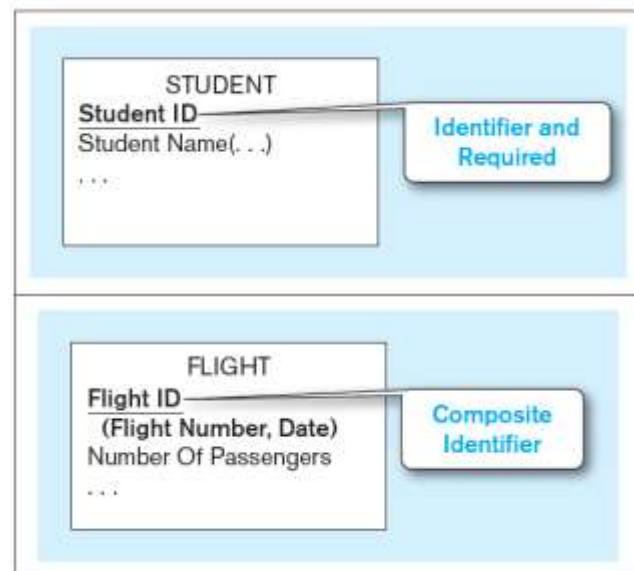
### იდენტიფიკატორი ატრიბუტი

იდენტიფიკატორი არის ატრიბუტი (ან ატრიბუტების კომბინაცია), რომლის მნიშვნელობა განასხვავებს არსის ტიპის ინდივიდუალურ შემთხვევებს. არსის ტიპის ერთიდაიგივე არცერთ ორ ეგზემპლარს არ შეიძლება ჰქონდეს ერთიდაიგივე მნიშვნელობის იდენტიფიკატორი ატრიბუტი. ადრე მოყვანილ მაგალითსი STUDENT (სტუდენტი) ტიპის არსის იდენტიფიკატორი არის Student ID, ხოლო AUTOMOBILE- ის იდენტიფიკატორი Vehicle ID. გავითვალისწინოთ, რომ ატრიბუტი, როგორცაა Student Name, არ არის იდენტიფიკატორი, რადგან ბევრ სტუდენტს შეიძლება ჰქონდეს იგივე სახელი და სტუდენტებს, ისევე როგორც ყველა ადამიანს, შეუძლიათ შეცვალონ თავიანთი სახელები. არსებობს კანდიდატი ინედიტიფიკატორის ცნებაც. ატრიბუტი **კანდიდატი იდენტიფიკატორი** რომ იყოს, არსის თითოეული მნიშვნელობისათვის უნდა ჰქონდეს ერთი მნიშვნელობა. E-R დიაგრამაზე იდენტიფიკატორის სახელი აღინიშნება ხაზის გასმით, როგორც ეს ნაჩვენებია STUDENT არსის ტიპის მაგალითზე, ნახაზი 2-9 ა. ატრიბუტი იდენტიფიკატორი რომ იყოს, ატრიბუტი უნდა იყოს აუცილებელი (ასე რომ განმასხვავებელი მნიშვნელობა უნდა არსებობდეს), ამიტომ იდენტიფიკატორი ასევე არის სქელი შრიფტით გამოსახული. E-R ნახაზის ზოგიერთი პროგრამა იდენტიფიკატორის წინ განათავსებს სიმბოლოს, რომელსაც ეწოდება სტერეოტიპი (მაგ., <<ID>> ან <<PK>>).

ზოგიერთი არსის ტიპისთვის არ არსებობს ერთი (ან ატომური) ატრიბუტი, რომელიც იდენტიფიკატორის ფუნქციას შეასრულებს (ან უზრუნველყოფს უნიკალურობას). ამასთან, ორი (ან მეტი) ატრიბუტი, რომელიც გამოიყენება კომბინირებულად, შეიძლება გამოდგეს როგორც იდენტიფიკატორი. **კომპოზიტური იდენტიფიკატორი** არის იდენტიფიკატორი, რომელიც შედგება კომპოზიტური ატრიბუტისგან. დიაგრამა 2-9 ბ აჩვენებს ობიექტს FLIGHT კომპოზიტური იდენტიფიკატორით Flight ID. ფრენის ID-ს თავის მხრივ აქვს კომპონენტის ატრიბუტები ფრენის ნომერი და თარიღი. ეს კომბინაცია საჭიროა FLIGHT ცალსახად დასადგენად.

**FIGURE 2-9** Simple and composite identifier attributes  
(a) Simple identifier attribute

(b) Composite identifier attribute



ზოგიერთ არსს შეიძლება ჰქონდეს ერთზე მეტი კანდიდატი იდენტიფიკატორი. თუ არსებობს ერთზე მეტი კანდიდატი იდენტიფიკატორი, დიზაინერმა იდენტიფიკატორად უნდა აირჩიოს ერთი მათგანი. არსებობს კრიტერიუმები, რომლებიც გამოიყენება იდენტიფიკატორების შერჩევისას:

1. იდენტიფიკატორად უნდა შეირჩეს ისეთი ატრიბუტი, რომელიც არ შეცვლის მნიშვნელობას არსის ტიპის თითოეული ეგზემპლარის სასიცოცხლო ციკლის განმავლობაში. მაგალითად, თანამშრომლის სახელისა და ხელფასის კომბინაცია (თუნდაც უნიკალური) ცუდი არჩევანი იქნება, როგორც EMPLOYEE- ს იდენტიფიკატორი, რადგან თანამშრომლის სახელისა და ხელფასის მნიშვნელობები შეიძლება ადვილად შეიცვალოს თანამშრომლის დასაქმების პერიოდში.
2. იდენტიფიკატორი უნდა აირჩეს ისე, რომ არსის თითოეული ეგზემპლარისათვის ამ ატრიბუტს გარანტირებული ჰქონდეს სწორი მნიშვნელობის მინიჭება და არ იყოს ნულოვანი (ან უცნობი). თუ იდენტიფიკატორი კომპოზიტური ატრიბუტია, მაგალითად, ფრენის ID 2-9 ზ სურათზე, დარწმუნებულები უნდა ვიყოთ, რომ იდენტიფიკატორის ყველა ნაწილს ექნება სწორი მნიშვნელობები.
3. უნდა მოვერიდოთ ე.წ. ინტელექტუალური იდენტიფიკატორის (ან გასაღების) გამოყენებას, რომელთა სტრუქტურა მიუთითებს კლასიფიკაციას, ადგილმდებარეობას და ა.შ. მაგალითად, იდენტიფიკატორის მნიშვნელობის პირველი ორი ციფრი შეიძლება მიუთითებდეს საწყობის მდებარეობას. ასეთი კოდები ხშირად იცვლება პირობების შეცვლისთანავე, რაც იდენტიფიკატორის მნიშვნელობებს აბათილებს.
4. გასათვალისწინებელია ერთი ატრიბუტიანი სუროგატი იდენტიფიკატორების ჩანაცვლების შესაძლებლობა დიდი კომპოზიტური იდენტიფიკატორით. მაგალითად, ატრიბუტი სახელწოდებით Game Number („თამაშის ნომერი“) შეიძლება გამოყენებულ იქნას არსის ტიპის GAME-სთვის, ნაცვლად მასპინძელი გუნდისა და სტუმრების გუნდის კომბინაციისა.

### *ატრიბუტების სახელდება და განმარტება*

მონაცემთა არსების სახელდების ზოგადი სახელმძღვანელო მითითებების გარდა, არსებობს ატრიბუტების სახელდების რამდენიმე სპეციალური სახელმძღვანელო, რომლებიც შემდეგნაირად გამოიყურება:

- *ატრიბუტის სახელი არის მხოლოდითი არსებითი სახელი ან არსებითი ფრაზა* (მაგალითად, Customer ID, Age, Product Minimum Price, or Major), რომლებიც წარმოიქმნება როგორც მონაცემთა მნიშვნელობები, და წარმოადგენს არსების ცნებებს ან ფიზიკურ მახასიათებლებს. ცნებები და ფიზიკური მახასიათებლები აღწერილია სახელებით.
- *ატრიბუტის სახელი უნდა იყოს უნიკალური.* ერთი და იმავე არსის ტიპის ორ ატრიბუტს არ შეიძლება ჰქონდეს ერთი და იგივე სახელი და სასურველია, სიცხადისათვის არსის ყველა ტიპისათვის არ არსებობდეს ორი ატრიბუტი ერთი და იგივე სახელით.
- *იმისათვის, რომ ატრიბუტის სახელი გახდეს უნიკალური და უფრო მკაფიო, თითოეული ატრიბუტის სახელი უნდა შეესაბამებოდეს სტანდარტულ ფორმატს.* მაგალითად, თქვენს უნივერსიტეტს შეუძლია შექმნას სტუდენტური GPA, სტუდენტის GPA-სგან განსხვავებით, როგორც ატრიბუტების დასახელების სტანდარტული ფორმატის მაგალითი. გამოსაყენებელ ფორმატს დაადგენს თითოეული ორგანიზაცია. ფორმატის წარმოდგენის გავრცელებული სახეა **[არსის ტიპის სახელი [[კვალიფიკატორი]]]** კლასი, სადაც [. . .] არასავალდებულო პუნქტია და {. . .} მიუთითებს იმაზე, რომ წინადადება შეიძლება განმეორდეს. **არსის ტიპის სახელი** არის იმ არსის (სუბიექტის) სახელი, რომელთანაც დაკავშირებულია ატრიბუტი. არსის ტიპის სახელი შეიძლება გამოყენებულ იქნას ატრიბუტის სახელის განსამარტად. ის თითქმის ყოველთვის გამოიყენება თითოეული არსის იდენტიფიკატორი ატრიბუტისთვის (მაგალითად, Customer ID - მომხმარებლის ID). **კლასი** არის ფრაზა ორგანიზაციის მიერ განსაზღვრული ფრაზების სიიდან, რომლებიც წარმოადგენს არსის დასაშვებ მახასიათებლებს ან თვისებებს (ან ამ მახასიათებლების აბრევიატურა). მაგალითად, კლასისთვის დასაშვები მნიშვნელობები (და მასთან დაკავშირებული დამტკიცებული აბრევიატურა) შეიძლება იყოს სახელი (Nm), იდენტიფიკატორი (ID), თარიღი (Dt) ან ოდენობა (Amt). ცხადია კლასის არსებობა აუცილებელია. **კვალიფიკატორი** არის ფრაზა ორგანიზაციის მიერ განსაზღვრული ფრაზების სიიდან, რომლებიც გამოიყენება კლასებზე შეზღუდვების დასადებად. არსის ტიპის თითოეული ატრიბუტის უნიკალურობის უზრუნველსაყოფად შეიძლება საჭირო გახდეს ერთი ან მეტი კვალიფიკატორი. მაგალითად, კვალიფიკატორი შეიძლება იყოს Maximum (Max), Hourly (Hrly), ან State (St). ასევე შეიძლება საჭირო არ იყოს კვალიფიკატორი: Employee Age (თანამშრომელთა ასაკი) და Student Major (სტუდენტის სპეციალობა) ორივე სრულად გამოკვეთილი ატრიბუტის სახელებია. ზოგჯერ შეიძლება აუცილებელი იყოს კვალიფიკატორი. მაგალითად, Employee Birth Date (თანამშრომლის დაბადების თარიღი) და Employee Hire Date (თანამშრომლის დაქირავების თარიღი) არის თანამშრომლის ორი ატრიბუტი, რომლებიც საჭიროებენ ერთ კვალიფიკატორს. ზოგიერთ შემთხვევაში კი ერთზე მეტი კვალიფიკატორი შეიძლება იყოს საჭირო. მაგალითად, Employee Residence City Name (ან Emp Res Cty Nm) არის თანამშრომლის საცხოვრებელი ქალაქის სახელი, ხოლო Employee Tax City სახელი (ან Emp Tax Cty Nm) არის ქალაქის სახელი, რომელშიც თანამშრომელი რეგისტრირებულია.



- *სხვადასხვა არსის ტიპების მსგავსი ატრიბუტები უნდა იყენებდეს ერთნაირ კვალიფიკატორს და კლასებს*, მხოლოდ იმ შემთხვევაში, თუ ეს არის ორგანიზაციაში გამოყენებული სახელები. მაგალითად, „ფაკულტეტებისა და სტუდენტების საცხოვრებელი ქალაქი უნდა იყოს, შესაბამისად, ფაკულტეტის რეზიდენციის ქალაქის სახელი და სტუდენტური საცხოვრებელი ქალაქის სახელი“. მსგავსი სახელების გამოყენება მომხმარებლებს უადვილებს იმის გაგებას, რომ ამ ატრიბუტების მნიშვნელობები მოდის იმავე მნიშვნელობათაგან, რასაც დომენებს ვეძახით. მომხმარებლებს შეიძლება სურთ ისარგებლონ მოთხოვნების საერთო დომენებით (მაგ., იპოვნ სტუდენტები, რომლებიც ცხოვრობენ იმავე ქალაქში, როგორც მათი მრჩეველი) და მომხმარებლებისთვის უფრო ადვილი იქნება იმის გაცნობიერება, რომ ასეთი შესატყვისი შეიძლება იყოს, თუ გამოყენებული იქნება იგივე კვალიფიკატორი და კლასის ფრაზები.

გარდა ატრიბუტების სახელების მითითებებისა ასევე არსებობს ატრიბუტების განმარტების გარკვეული კონკრეტული სახელმძღვანელო მითითებები, რომლებიც შემდეგნაირად გამოიყურება:

- *ატრიბუტის განმარტებაში მიეთითება თუ რა ატრიბუტია ეს და რატომ არის მნიშვნელოვანი*. განმარტება ხშირად ატრიბუტის სახელის თანმდევაა. მაგალითად, სტუდენტის რეზიდენციის ქალაქის სახელი შეიძლება განისაზღვროს, როგორც „ქალაქის სახელი, რომელიც სტუდენტის მუდმივი საცხოვრებელი ადგილია“.
- *ატრიბუტის განმარტებამ მკვეთრად უნდა განისაზღვროს თუ რა არის შეტანილი და რა არ არის შეტანილი მის მნიშვნელობაში*; მაგალითად, „თანამშრომლის ყოველთვიური ხელფასის თანხა არის ყოველთვიურად გადახდილი თანხის ოდენობა დასაქმებულის საცხოვრებელი ქვეყნის ვალუტაში, ყოველგვარი შეღავათების, პრემიებისა, ანაზღაურების ან სპეციალური გადასახადების გარდა.“
- *ატრიბუტის ნებისმიერი მეტსახელი, ან ალტერნატიული სახელი შეიძლება მითითებული იყოს განმარტებაში ან შეიძლება სხვა ადგილას ატრიბუტის შესახებ დოკუმენტაციაში*, რომელიც შესაძლოა ინახებოდეს CASE ინსტრუმენტის საცავში, რომელიც გამოიყენება მონაცემთა განმარტებების შესანარჩუნებლად.
- *ასევე შეიძლება სასურველი იყოს განმარტებაში ატრიბუტის მნიშვნელობების წყაროს მითითება*. წყაროს მითითებამ შეიძლება უფრო ნათელი გახადოს მონაცემთა მნიშვნელობა. მაგალითად, „მომხმარებლის სტანდარტული ინდუსტრიული კოდი“ არის მომხმარებლის ბიზნესის ტიპის მითითება.
- *ატრიბუტის განმარტებამ (ან სხვა დაზუსტებამ CASE ხელსაწყო საცავში) ასევე უნდა მიუთითოს, სავალდებულოა ატრიბუტის მნიშვნელობა თუ არასავალდებულოა*. ეს ბიზნეს წესი ატრიბუტის შესახებ მნიშვნელოვანია მონაცემთა მთლიანობის შესანარჩუნებლად. განმარტებით, არსის ტიპის იდენტიფიკატორის შესაბამისი ატრიბუტი სავალდებულოა. თუ ატრიბუტის მნიშვნელობა სავალდებულოა, მაშინ ობიექტის ტიპის ეგზემპლარის შესაქმნელად, უნდა იყოს მითითებული ამ ატრიბუტის მნიშვნელობა. აუცილებელი ნიშნავს, რომ არსის ეგზემპლარს ყოველთვის უნდა ჰქონდეს მნიშვნელობა ამ ატრიბუტისთვის, არა მხოლოდ ეგზემპლარის შექმნის დროს. არასავალდებულო ნიშნავს, რომ ამ ატრიბუტის მნიშვნელობა შეიძლება არ არსებობდეს არსის ტიპის ეგზემპლარის შესაძენად. ატრიბუტის არასავალდებულოებისას შესაძლებელია შემდგომში მიეთითოს, უნდა არსებობდეს თუ არა ყოველთვის მონაცემების შეყვანის მერე. მაგალითად, „თანამშრომელთა დეპარტამენტის იდენტიფიკატორი არის განყოფილების

იდენტიფიკატორი, რომელსაც მიმაგრებულია თანამშრომელი. თანამშრომელი შეიძლება არ იქნეს მიმაგრებული განყოფილებაზე დაქირავებისას (ამიტომ ეს ატრიბუტი თავდაპირველად არასავალდებულოა), მაგრამ მას შემდეგ, რაც თანამშრომელი მიმაგრდება განყოფილებაზე, თანამშრომელი ყოველთვის უნდა დაინიშნოს რომელიმე განყოფილებაში. ”

- ატრიბუტის განმარტებამ (ან სხვა დაზუსტებამ CASE ხელსაწყოთა სავსეში) შეიძლება ასევე მიუთითოს, *შეიძლება თუ არა შეიცვალოს ატრიბუტის მნიშვნელობა* მას შემდეგ, რაც მიენიჭება მნიშვნელობა და სანამ წაიშლება არსის ეგზემპლარი. ეს ბიზნესის წესი ასევე აკონტროლებს მონაცემთა მთლიანობას. არაიტელექტუალურმა იდენტიფიკატორებმა შეიძლება დროთა განმავლობაში არ შეცვალონ მნიშვნელობები. არსის ეგზემპლარისათვის ახალი არაიტელექტუალური იდენტიფიკატორის მინიჭების მიზნით, ეს ეგზემპლარი ჯერ უნდა წაიშალოს და შემდეგ ხელახლა შეიქმნას.
- *მრავალმნიშვნელოვანი ატრიბუტისთვის, ატრიბუტის განსაზღვრებაში უნდა მიეთითოს ატრიბუტის მნიშვნელობის მაქსიმალური და მინიმალური რაოდენობა* არსის ეგზემპლარისათვის მაგალითად, "Employee Skill Name" არის უნარი, რომელსაც თანამშრომელი ფლობს. თითოეული თანამშრომელი უნდა ფლობდეს მინიმუმ ერთ უნარს, ხოლო თანამშრომელს შეუძლია ჩამოთვლოს ყველაზე მეტი 10 უნარი". ატრიბუტის მრავალმნიშვნელოვანობის მიზეზი შეიძლება იყოს ატრიბუტის „ისტორია“ შენახვის აუცილებლობა. მაგალითად, "თანამშრომლის ყოველწლიური გაცდენის დღეების რიცხვი არის კალენდარული წლის დღეების რაოდენობა, როდესაც თანამშრომელი არ იმყოფებოდა სამსახურში". თანამშრომელი განიხილება გამცდენად იმ შემთხვევაში, თუ ის მუშაობს დღის დაგეგმილი საათების 50 პროცენტზე ნაკლებს. ამ ატრიბუტის მნიშვნელობა უნდა იყოს დაცული ყოველი წლის განმავლობაში, როდესაც თანამშრომელი მუშაობს კომპანიაში. ”
- ატრიბუტის განმარტებაში შეიძლება *მიეთითოს ნებისმიერი დამოკიდებულება რომელიც ამ ატრიბუტს აქვს სხვა ატრიბუტებთან*. მაგალითად, "თანამშრომელთა შვებულების დღეების რაოდენობა" არის თანამშრომლის ანაზღაურებადი შვებულების დღეების რაოდენობა. თუ თანამშრომელის არტიბუტს „თანამშრომლის ტიპი“, აქვს მნიშვნელობა - "გათავისუფლებული", მაშინ თანამშრომლის შვებულების დღეების მაქსიმალური მნიშვნელობა განისაზღვრება ფორმულის მიხედვით, რომელიც მოიცავს თანამშრომლის წლების მუშაობას».