რელაციური მონაცემთა ბაზები პრაქტიკაში

მონაცემთა ბაზების რელაციური მოდელის შესაძლებლობების დემონსტრირების მიზნით პროექტში სიმულირებული იქნება ბიზნეს მოდელი, რომლის ფარგლებშიც დავაგენერირებთ საშუალო მოცულობის მონაცემებს და გავმართავთ მონაცემთა საცავს. მონაცემთა საცავი აგებულ იქნება ინმონისეული პრინციპით.

საბაზისო მონაცემები აღებულია პლატფორმა Kaggle-ის უფასო მონაცემთა კატალოგიდან. შემდგომ, საბაზისო მონაცემი დამუშავებულია პითონზე დაწერილი კოდით, რომელიც მონაცემებს გადაამუშავებს და ბიზნეს მოდელს მოარგებს, რის შედეგადაც გვექნება სიმულირებული მონაცემები, რომელიც იდეაში დაგროვებულია ბიზნესპროცესების საშუალებით.

ბიზნესის აღწერა

ავტომანქანა ბევრი ადამიანის ცხოვრების აუცილებელი ელემენტია. კავკასიის ქვეყნებში მომხმარებლები ძირითადად ირჩევენ მეორადი მანქნის ყიდვას დაბალი მშპ-ს გამო, ამიტომ ჩვენი ბიზნესი ბაზარს აშშ-ს სხვადასხვა შტატებიდან ჩამოყვანილ მეორად მანქნებს სთავაზობს.

მონაცემთა საცავის შექმნის მიზეზი

ხსენებული ტიპის ბიზნესი ძალიან კონკურენტუნარიანია, ამიტომ თუ გვსურს ვიყოთ წარმატებული ამ სფეროში, პასუხისმგებლობით უნდა მიუდგეთ ამ საქმეს და გავიგოთ იმ ფაქტორებზე, რომლებიც გავლენას ახდენენ მომხმარებლის არჩევანზე. უპირველეს ყოვლისა, ეს შეიძლება გაკეთდეს გაყიდვების შესახებ მოგროვილი ინფორმაციის შეგროვებით და ანალიზით.

მონაცემთა საცავის შექმნის სარგებელი

მონაცემთა საცავში არსებულ მონაცემთა დამუშავებით შეგვეძლება პასუხი გავცეთ შემდეგ კითხვებს:

რომელ მახასიათებლებს აქვთ მანქნის ფასზე დიდი გავლენა?

რომელ ბრენდებს ენიჭებათ უპირატესობა?

და ა. შ.

მონაცემთა აღწერა

ინფორმაცია პროდუქტზე

* ბრენდი
* მოდელი
* გადაცემათა კოლოფი
* ვინკოდი
* მდგომარეობა (1-დან 50-მდე შეფასების შკალით)
* ოდომეტრი
* ინფორმაცია შესყიდვაზე
* შესყიდვის დრო
* ფასი
* MMR (ახალი მანქნის საბაზრო ფასი)

ინფორმაცია კონსულტანტზე

* სახელი, გვარი
* ელ-ფოსტა
* თანამშრომლის ბარათი

მომხმარებლის ინფორმაცია

* სახელი, გვარი
* ელ-ფოსტა

დამატებითი ატრიბუტები:

* ფერი
* ინტერიერის ფერი
* პირველი მფლობელის შტატი

მონაცემთა სტრუქტურა

მომხმარებლები:

|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Customer\_id | მომხმარებლის იდენტიფიკატორი | Bigint |
| Customer\_firstname | სახელი | Varchar |
| Customer\_lastname | გვარი | Varchar |
| Customer\_mail | საკონტაქტო ელ-ფოსტა | Varchar |

მაგალითი:

|  |  |  |  |
| --- | --- | --- | --- |
| Customer\_id | Customer\_firstname | Customer\_lastname | Customer\_mail |
| 1 | Amiran | Menabde | amiran@gmail.com |

პროდუქტები (ავტომანქანები)

|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Product\_id | ავტომანქნის იდენტიფიკატორი | Bigint |
| Vin | ვინ კოდი | Varchar |
| State | შტატი (ორი სიმბოლო) | Varchar |
| Condition | კონდიციის 0-50 შკალით შეფასება | Tinyint |
| Odometer | გავლილი მანძილი | Int |
| Color | Color of car | Varchar |
| Interior | Color of interior | Varchar |

მაგალითი

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Product\_id | Odometer | Color | interior | Vin | State | Condition |
| 1 | 10000 | Black | Black | 5xyktca69fg566472 | CA | 23 |

კონსულტანტები

|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Employee\_id | თანამშრომლის იდენტიფიკატორი | Bigint |
| firstname | სახელი | Varchar |
| lastname | გვარი | Varchar |
| mail | ელ-ფოსტა | Varchar |
| badge | თანამშრომლის კოდი | Varchar |

მაგალითი

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Employee\_id | firstname | lastname | mail | badge |
| 1 | Amiran | Menabde | amiran@gmail.com | Er45 |

მოდელები

|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Brand\_id | მდელის იდენტიფიკატორი | Bigint |
| Brand | ბრენდი | Varchar |
| Model | მოდელი | Varchar |
| Transmission | გადატემათა კოლოფი | Varchar |
| Body | კორპუსის ტიპი (სედანი, ჯიპი...) | Varchar |
| year | გამოშვების წელი | smallint |

მაგალითი

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Brand\_id | Make | Model | Trim | Year | Body | Transmission |
| 1 | BMW | F30 | LX | 2013 | sedan | automatic |

თარიღები

|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Date\_id | თარიღის იდენტიფიკატორი | Bigint |
| Year | წელიწადი | Tinyint |
| Month | თვე | Tinyint |
| Day | რიცხვი | Tinyint |

მაგალითი

|  |  |  |  |
| --- | --- | --- | --- |
| Date\_id | Year | Month | Day |
| 1 | 2015 | 02 | 20 |

ფაქტი

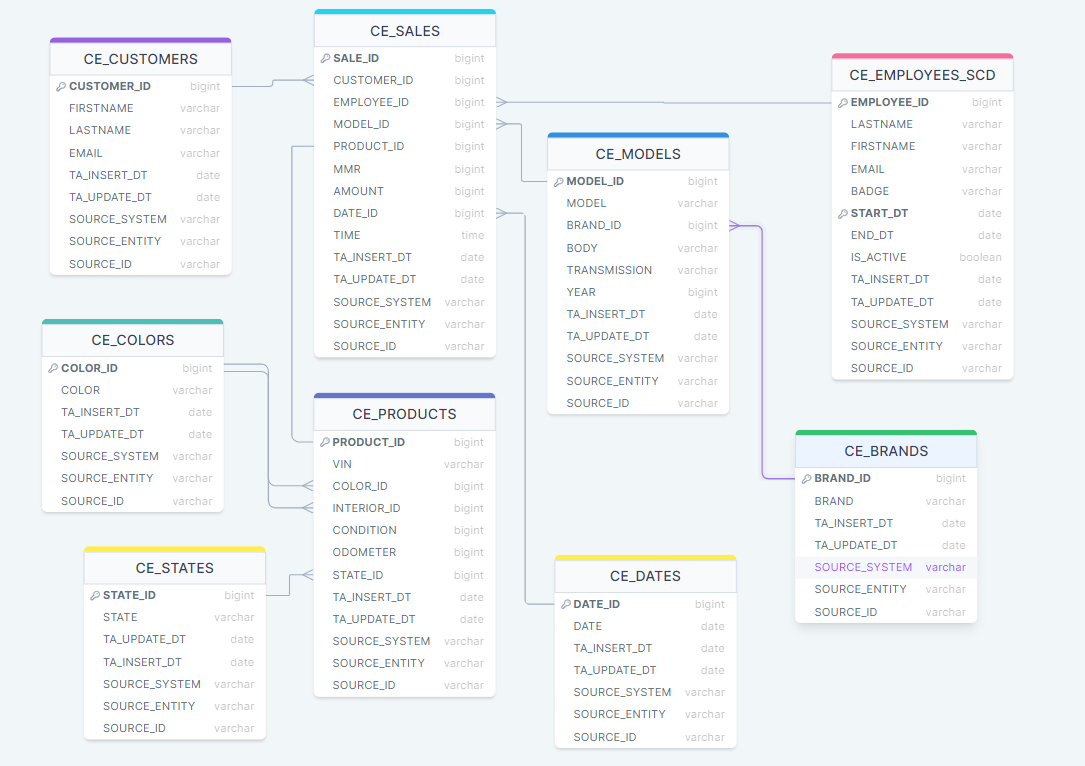
|  |  |  |
| --- | --- | --- |
| სვეტის სახელი | აღწერა | მონაცემის ტიპი |
| Sale\_id | შესყიდვის იდენტიფიკატორი | Bigint |
| Product\_id | პროდუქტის იდენტიფიკატორი | Bigint |
| Customer\_id | მომხმარებლის იდენტიფიკატორი | Bigint |
| Employee\_id | თანამშრომლის იდენტიფიკატორი | Bigint |
| Model\_id | მოდელის იდენტიფიკატორი | Bigint |
| Date\_id | თარიღის იდენტიფიკატორი | Bigint |
| Amount | მოცულობა | Int |
| Profit | მოგება (Amount – MMR) | Int |
| Time | ზუსტი დრო | Time |
| Mmr | ახალი მანქნის საბაზრო ფასი | Int |

მაგალითი

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sale\_id | Product\_id | Customer\_id | Employee\_id | Mmr |
| 1 | 1 | 2 | 20 | 10000 |
| Model\_id | Date\_id | amount | profit | Time |
| 1 | 1 | 15000 | 5000 | 16:00 |

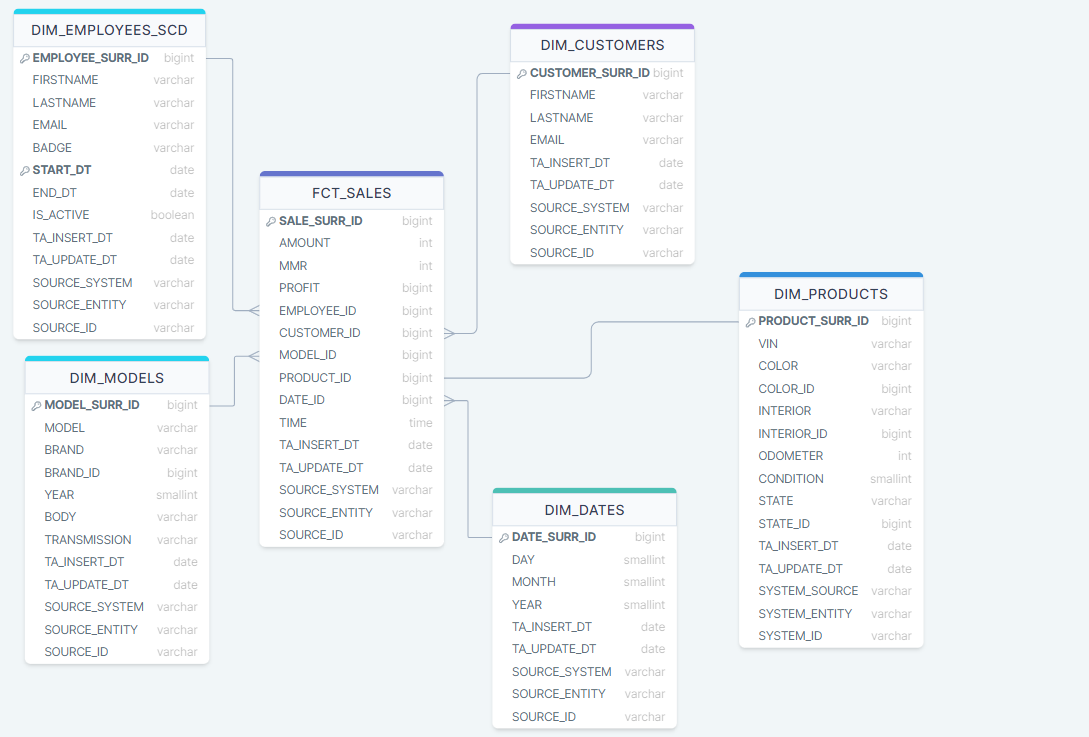
მონაცემთა ბაზის ნორმალიზებული ფორმა

სისტემის ეს ნაწილი უზრუნველყოფს, რომ მონაცემები ინახებოდეს დედუპლიცირებული და შესაბამისი ტიპების ფორმით, რომლებსაც ერთმანეთთან საიმედო კავშირი აქვთ. მონაცემების ამ ფენაზე გატარება გვეხმარება, რომ საბოლოო მოდელს მაღალი ხარისხის მონაცემები მივაწოდოთ.



მონაცემთა ბაზის დენორმალიზებული ფორმა

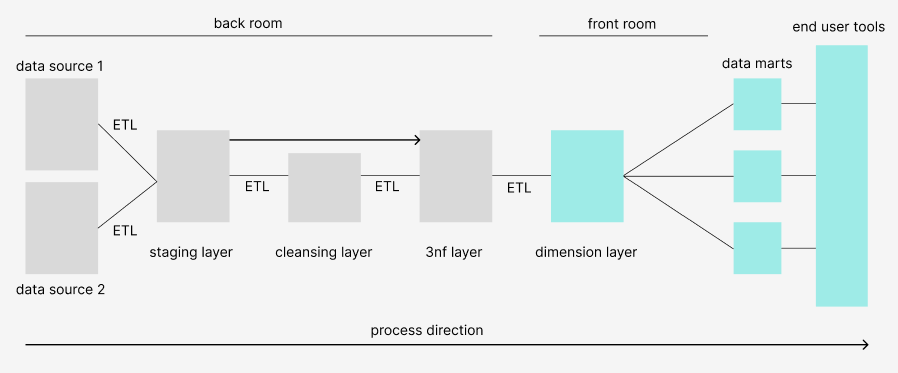
ყველა ზემოთხსენებული პროცესის შედეგად, ჩვენ ვიღებთ მონაცემთა დენორმალიზებულ სტრუქტურას, რომელიც ოპტიმიზირებულია სისტემის მიერ მონაცემთა სწრაფად გამოსატანად. სწორედ ამ სტრუქტურას უკავშირდება ყველა ის ანალიტიკური აპლიკაცია, რომელიც ბიზნეს რეპორტის საშუალებას იძლევა.



მონაცემთა საწყობის სისტემა

სურათზე ნაჩვენებია სქემა, რომელიც მონაცემთა საწყობის სისტემას ზედაპირულად ასახავს.

* მონაცემების ამოღება ხდება 2 სხვადასხვა ბიზნეს წყაროდან
* მონაცემები იტვირთება staging ფენაზე და გადის საჭირო ტრანსფორმაციას
* მონაცემთა ნაწილი დედუპლიცირდება cleansing ფენაზე
* მონაცემები გადადიან მე-3 ხარისხის ნორმალიზებულ ფენაზე
* მონაცემები გადადიან დენორამლიზებულ ფენაზე
* დენორმალიზებული ფორმიდან იქმნებიან data mart-ები
* Data mart-ს უკავშირდებიან აპლიკაციები



ინფორმაციის შენახვის პროცესი

სურათზე გვაქვს ინფორმაციის შენახვის დეტალური სისტემა, სადაც სტრუქტურის ობიექტებია აღნიშნული.

