Bir resim etiketleme uygulaması oluşturduğumuzu varsayalım. Bu uygulama ile kullanıcılar resimlere etiket atayabilir , en sevdikleri resimleri oylayabilir ve resimlere etiketlere göre göz atabilir

Kullanacağımız tablolar

İmage table()

Image : her bir görselin url’sini ve görsel için verilen oyların sayısını saklar

Tag : her etiketi ve o etikete sahip resim sayısını saklar

ImageTag : Resimler ve Etiketler arasındaki ilişkiyi saklar

**Tables(tablo)** Diğer veritabanı sistemlerine benzer şekilde, DynamoDB verileri tablolarda depolar. Bir tablo verilerinin topluluğudur. Örneğin, arkadaşlarınız, aileniz veya ilgilendiğiniz herhangi biri hakkındaki kişisel iletişim bilgilerini saklamak için kullanabileceğiniz Kişiler adlı örnek tabloya bakın . İnsanların kullandığı araçlar hakkında bilgi depolamak için Arabalar tablonuz da olabilir .

**Items(öğe)** Her tablo sıfır veya daha fazla öğe içerir. Bir öğe , diğer tüm öğeler arasında benzersiz bir şekilde tanımlanabilen bir öznitelikler grubudur. Bir de insanlar masanın, her öğe bir kişiyi temsil eder. Bir İçin Otomobiltablo, her öğe bir aracı temsil eder. DynamoDB'deki öğeler, diğer veritabanı sistemlerindeki satırlara, kayıtlara veya kayıtlara birçok yönden benzerdir. **DynamoDB'de, bir tabloda saklayabileceğiniz öğe sayısında bir sınır yoktur.**

**Attributes(öznitelikler)** Her öğe bir veya daha fazla öznitelikten oluşur. Bir öznitelik , daha fazla ayrıştırılması gerekmeyen temel bir veri öğesidir. Örneğin, Kişiler tablosundaki bir öğe, KişiKimliği , Soyadı , Ad , vb. Adlı öznitelikleri içerir . Bir Departman tablosu için, bir öğenin Departman Kimliği gibi öznitelikleri olabilir, Ad , Yönetici vb. DynamoDB'deki özellikler, diğer veritabanı sistemlerindeki alanlara veya sütunlara birçok yönden benzerdir.

**PEOPLE(TABLE)**

{ //item\_head//

“**PersonID**”:101, //partition key

“LastName:”:”smith”,

“FirstName”:”Fred”,

“Phone”:”555-4321”

} //item\_end//

{

“**PersonID**” : 102, //partition key

“LastName”: ”Jones”,

“Address” : {

“Street”: 102,

“City”: “Anytown” ,

“State”: “OH”,

“ZIPCode”: 12345

}

}

“**PersonID**” : 103, //partition key

“LastName”: “Stephens”,

“FirstName”: ”Howard”,

“Address”: {

“Street”:”123 Main”,

“City”:”London”,

“PostalCode”: ”46050”

}

“FavoriteColor”: ”Blue”

}

PEOPLE TABLOSU İLE İLGİLİ

Ne özniteliklerin ne de veri türlerinin önceden tanımlanması gerekmez. Her öğenin kendine özgü özellikleri olabilir.

Bazı öğelerin iç içe bir özniteliği ( Adres kısmında olduğu gibi ) vardır. DynamoDB, 32 düzey derinliğe kadar yuvalanmış öznitelikleri destekler

Aşağıda, müzik koleksiyonunuzu takip etmek için kullanabileceğiniz Müzik adlı başka bir örnek tablodur .

Music

{

“Artist” : “No One You Know”,

“SongTitle” : “ My Dog Spot”,

“AlbumTitle” : “Hey Now” ,

“Price”: 1.98,

“Genre”: “Country”

“CriticRaiting” : 8.4,

}

{

“Artist” : “No One You Know”,

“SongTitle” : “ My Dog Spot”,

“AlbumTitle” : “Hey Now” ,

“Price”: 1.98,

“Genre”: “Country”

“CriticRaiting” : 8.4,

}

}

“Artist” : “The Acme Band”,

“SongTitle” : “ still in Love”,

“AlbumTitle” : “The Buck Starts Here” ,

“Price”: 4.98,

“Genre”: “Rock”

“CriticRaiting” : 8.4,

}

.

.

.