

## Logical database Design

By Afroz Ahmad, Mob. No. 9006658740

### डेटाबेस डिजाइन (Database design)

डेटाबेस डिजाइन वह है जो अनावश्यक डेटा को कम करने के लिए हमारी सूचना को विषय-आधारित तालिकाओं (Table) में विभाजित करता है। आवश्यक सूचना के साथ पहुँच प्रदान करता है, जिसकी आवश्यकता तालिकाओं में सूचना को एक साथ जोड़ने के लिए होती है। हमारी सूचना की सटीकता और अखंडता को समर्थन और सुनिश्चित करने में मदद करता है। लॉजिकल डेटाबेस डिजाइन में डेटा का एक दूसरे से सम्बन्ध और डेटा को किस प्रकार संग्रहीत किया जाना है यह बताया जाता है। तार्किक डेटाबेस डिजाइन डेटा तत्वों (data element) और उनके संबंधों (Relationship) को परिभाषित करने में मदद करता है। भौतिक डेटाबेस डिजाइन तार्किक डेटा डिजाइन के दौरान एकत्रित आवश्यकताओं के आधार पर वास्तविक डेटाबेस को डिजाइन करने में मदद करता है।

**Logical database design:** इस डेटाबेस डिजाइन में जितना जितना डेटा होता है उससे सम्बंधित सभी विवरण इसमें मौजूद होते हैं। अगर हम डेटाबेस में कोई नियम लागू करना चाहते हैं तो इसमें सभी नियम परिभाषित करने होते हैं और जितना भी डेटा हमारे डेटाबेस के अन्दर होता है उसके संरचना (Structure) से सम्बंधित सूचना भी लॉजिकल डेटाबेस डिजाइन के अन्दर उपलब्ध होती है। इसके अतिरिक्त डेटाबेस के अन्दर जितनी भी इकाइयाँ (entities) और इस इकाइयों के अन्दर जितने भी गुण (attributes) होते हैं उदाहरण के लिए स्टूडेंट्स हमारे पास इकाइयाँ (entities) है तो उसके गुण (attributes) स्टूडेंट्स का नाम, उसका एड्रेस, उसका मोबाइल नंबर हो सकता है यह सभी सूचना या डेटा गुण (attributes) होती है। इकाइयाँ (entities) और गुण (attributes) के बीच सम्बन्ध (Relationship) होता है। यह सभी चीज़ें लॉजिकल डेटाबेस का अन्दर वर्णन (Describe) किया जाता है। लॉजिकल डेटाबेस के अन्दर निम्नलिखित प्रक्रिया (process) का पालन किया जाता है।

- **Represent entities:** इसके अंतर्गत डेटाबेस में जितनी भी इकाइयाँ (entities) होंगी उसे ER (Entities relationship) diagram के रूप में दर्शाया जाता है। इस इकाइयाँ (entities) को दर्शाने के लिए rectangle symbol का प्रयोग किया जाता है। इकाइयाँ (entities) का नाम हमेशा बड़े अक्षर (Capital letter) में लिखा जाता है। गुण (attributes) को दर्शाने के लिए Oval symbol का प्रयोग किया जाता है।

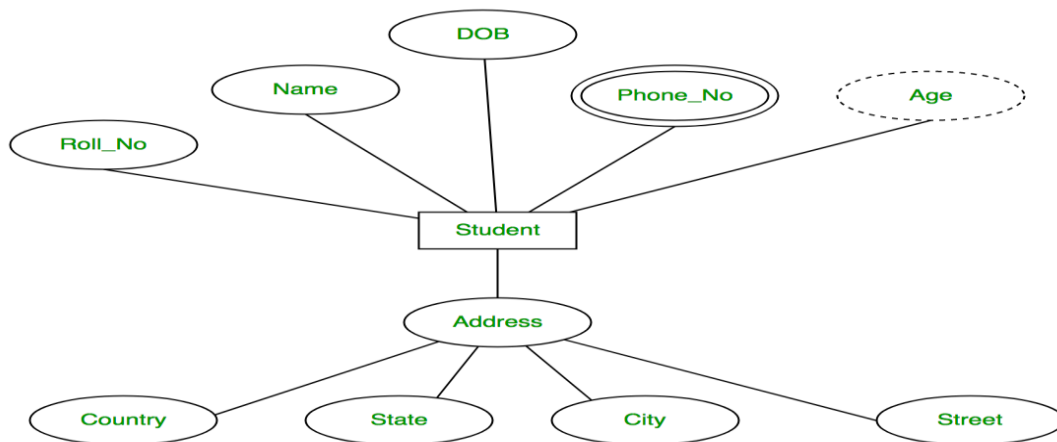


Image of ER diagram

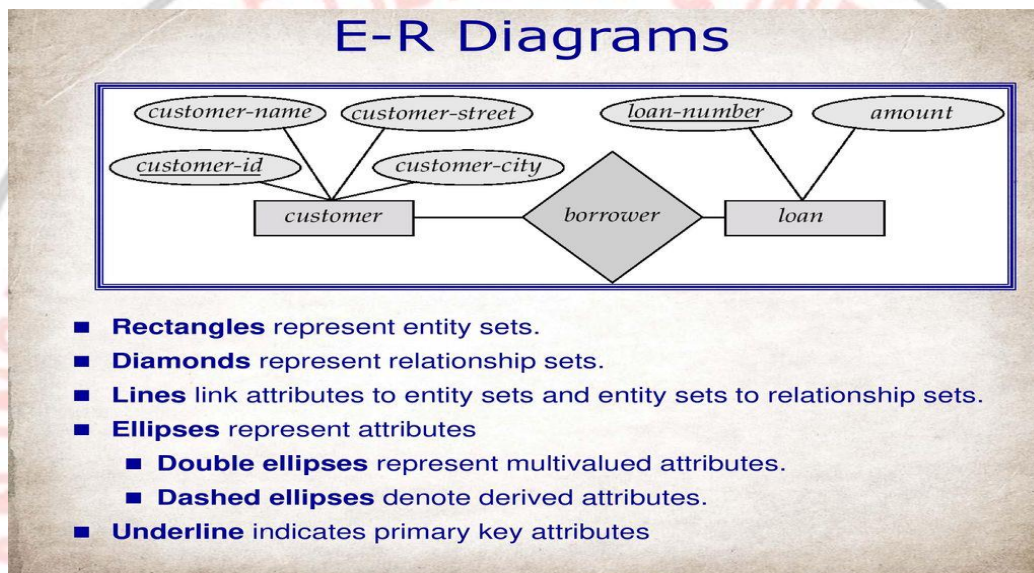


Image of ER diagram

ऊपर दिए गए चित्र में छात्र इकाई (entity) है और Roll no., Name, DOB, Phone no. Age, Address, Country, State, City, Street यह सभी गुण (attributes) हैं। इसमें एक ऐसा गुण (attributes) होता है जो प्राइमरी की (Primary key) या Identifier होता है जो कि रोल नंबर या ID हो सकता है। यह सभी इकाइयां (entities) को एक दूसरे से अलग करता है उदाहरण के लिए किसी कंपनी में बहुत सारे कर्मचारी काम करते हैं सभी कर्मचारी ID के आधार पर एक दूसरे से अलग होते हैं। दो कर्मचारियों का नाम या पता एक हो सकता है मगर ID एक नहीं हो सकता है यह एक दूसरे से अलग ही होता है। इसी प्रकार किसी कॉलेज में हर छात्र का रोल नंबर अलग अलग होता है।

- **Represent relationship:** इसमें विभिन्न इकाइयां (entities) के बीच जो भी सम्बन्ध होता है उसे प्रदर्शित (Represent) किया जाता है। इसमें हम यह देखते हैं कि हमारे डेटाबेस में अलग अलग जो इकाइयां (entities) हैं वह एक दूसरे के साथ किस प्रकार से जुड़े हुए या लिंक्ड (Linked) हैं। इसे दर्शाने के लिए प्राइमरी की और फॉरेन की (Foreign key) का उपयोग कर सकते हैं।
- **Merge relation:** इसके द्वारा अगर टेबल डुप्लीकेट होता है तो इसे एक साथ जोड़ (merge) कर दिया जाता है। इसकी आवश्यकता तब होती है जब एक ही इकाइयां (entities) से सम्बंधित एक से अधिक टेबल बन जाये तो उस स्थिति में हम सम्बन्ध को एक दूसरे के साथ मिला देते यानि merge कर देते हैं।

उदाहरण के लिए दो कर्मचारी का टेबल है।

**EMP1** (Emp. No, Name, Address, Phone)

**EMP2** (Emp. No., Name, address, Job, DOB)

दोनों टेबल के अन्दर कर्मचारी से सम्बंधित सूचना उपलब्ध है। इसमें गुण (attributes) कुछ तो एक ही हैं और कुछ अलग हैं मगर सूचना एक ही चीज़ के बारे में है। इस प्रकार के टेबल में दोनों टेबल को मिला कर एक नया टेबल बना दिया जाता है और जिसका नाम अलग रखा जाता है इसे ही हम Merge relation कहते हैं।

**After merging**

**EMP** (Emp. No., Name, Address, Phone, Job, DOB)

जिससे वह सभी गुण (attributes) इस टेबल में आ जाते हैं जो दोनों टेबल में उपलब्ध थे इसमें कोई भी गुण (attributes) नहीं दोहराएगा।

- **Normalize relation:** अगर टेबल के अन्दर डुप्लीकेशन आ रही है तो उसे normalize करके सम्बन्ध (relation) के द्वारा हटा कर साधारण रूप में बदला जाता है। जब हम इकाइयां (entities) को दर्शाते हैं और मिलाते (Merge) करते हैं तो ऐसी स्थिति में हमारे इकाइयां (entities) जो डेटा के अन्दर उपलब्ध रहता है वहाँ पर कुछ न कुछ दोहराव (Repetition) आ जाती है। उस दोहराव का परिणाम यह होता है कि जब हम डेटा को दर्शाते हैं या प्रविष्ट (Insert) करते हैं तो समस्या उत्पन्न होती है उसे हम anomalies कहते हैं। यह अनामोलिज़ किसी डेटा को इन्सर्ट, अपडेट या डिलीट करते समय आ सकती है। Normalization में डेटा को आसान रूप में बनाते हैं ताकि इस प्रकार की समस्या anomalies को हटाया जा सके।

**By Afroz Ahmad**

**Khuda Bakhsh O. P. Library, Patna**

**Mob. 9006658740**

