

## इकाई-1

### 1.1 डेटाबेस कांसेप्ट का परिचय:

आम तौर पर एक डेटाबेस संबंधित जानकारी का एक संगठित संग्रह होता है। संगठित जानकारी या डेटाबेस एक आधार के रूप में कार्य करता है जिसमें से वांछित जानकारी प्राप्त की जा सकती है। लोग अपने दिन-प्रतिदिन के जीवन में कई डेटाबेस का उपयोग करते हैं। शब्दकोश, टेलीफोन निर्देशिका, लाइब्रेरी कैटलॉग, आदि डेटाबेस के लिए उदाहरण हैं जहां वर्णमाला या वर्गीकृत क्रम के अनुसार प्रविष्टियों की व्यवस्था की जाती है।

एक डेटाबेस डेटा का एक संग्रह है, एक या एक से अधिक संबंधित संगठनों की गतिविधियों का वर्णन करता है। 'DATA' शब्द को एक इकाई की विशेषता के मूल्य के रूप में परिभाषित किया जा सकता है। समान विशेषताओं वाली संस्थाओं के संबंधित डेटा आइटम के किसी भी संग्रह को 'DATABASE' के रूप में संदर्भित किया जा सकता है। डेटा का मात्र संग्रह इसे डेटाबेस नहीं बनाता है। जिस तरह से यह प्रभावी और कुशल उपयोग के लिए आयोजित किया जाता है वह इसे एक डेटाबेस बनाता है।

एक इकाई व्यक्ति या पुस्तक के रूप में ठोस हो सकती है, या यह अमूर्त हो सकती है जैसे ऋण या छुट्टी या एक अवधारणा। इकाइयाँ वस्तुओं की मूल इकाइयाँ हैं जिनका ठोस अस्तित्व हो सकता है या विचार या अवधारणाएँ बन सकती हैं। एक इकाई सेट उसी प्रकार की संस्थाओं का एक सेट है जो समान गुणों या विशेषताओं को साझा करते हैं।

एक इकाई को विशेषताओं के सेट द्वारा दर्शाया जाता है। एक विशेषता को डेटा आइटम, डेटा तत्व, डेटा फ़ील्ड, आदि के रूप में भी संदर्भित किया जाता है। विशेषताएँ एक इकाई सेट के प्रत्येक सदस्य के पास वर्णनात्मक गुण हैं। संबंधित संस्थाओं की ग्रोपिंग एक इकाई सेट बन जाती है।

उदाहरण के लिए: पुस्तकालय के माहौल में,

इकाई सेट - कैतेलोग

इकाई-किताबें, पत्रिकाओं, एवी-सामग्री, आदि

विशेषताएँ- इसमें लेखक, शीर्षक, छाप, अकन शामिल हैं। नहीं, आईएसबीएन, आदि..

'DATA' शब्द का अर्थ एक तथ्य या अधिक विशेष रूप से एक इकाई की विशेषता का एक मूल्य है। डेटा सूचना का कच्चा माल है। 'बेस' शब्द का अर्थ किसी भी चीज के समर्थन, नींव या प्रमुख घटक से है। इसलिए आधार डेटा का समर्थन करता है।

एक 'DATABASE' की परिकल्पना एक ऐसे सिस्टम के रूप में की जा सकती है जिसका आधार, जिसकी मुख्य अवधारणा, डेटा को संभालने का एक विशेष तरीका है। दूसरे शब्दों में, एक डेटाबेस कंप्यूटर आधारित रिकॉर्ड रखने से ज्यादा कुछ नहीं है। डेटाबेस का उद्देश्य जानकारी को रिकॉर्ड करना और बनाए रखना है।

संक्षेप में, एक डेटाबेस-प्रबंधन प्रणाली (DBMS) अंतरसंबंधित डेटा का एक संग्रह है और उन डेटा तक पहुंचने के लिए कार्यक्रमों का एक सेट है। यह एक निहित अर्थ के साथ संबंधित डेटा का एक संग्रह है और इसलिए एक डेटाबेस है। डेटा का संग्रह, जिसे आमतौर पर डेटाबेस के रूप में संदर्भित किया जाता है, में एक उद्यम से संबंधित जानकारी होती है। DBMS का प्राथमिक लक्ष्य डेटाबेस जानकारी को संग्रहीत करने और पुनः प्राप्त करने का एक तरीका प्रदान करना है जो सुविधाजनक और कुशल दोनों है। डेटा से हमारा मतलब ज्ञात तथ्यों से है जिन्हें रिकॉर्ड किया जा सकता है और जिसका अर्थ निहित होता है। उदाहरण के लिए, जिन लोगों को आप जानते हैं उनके नाम, टेलीफोन नंबर और पते पर विचार करें। आपने इस डेटा को अनुक्रमित पता पुस्तिका में दर्ज

किया हो सकता है, या आपने पर्सनल कंप्यूटर और सॉफ्टवेयर जैसे DBASE IV या V, Microsoft ACCESS, या EXCEL का उपयोग करके इसे डिस्कट पर संग्रहीत किया हो। एक डेटा - डेटा की एक इकाई - एक प्रतीक या प्रतीकों का एक सेट है जिसका उपयोग किसी चीज़ का प्रतिनिधित्व करने के लिए किया जाता है। प्रतीकों और वे जो प्रतिनिधित्व करते हैं, उसके बीच का यह संबंध है कि हम जानकारी से क्या मतलब रखते हैं। इसलिए, सूचना की व्याख्या आंकड़ों से की जाती है - शब्दार्थ के साथ दिया गया डेटा। ज्ञान का तात्पर्य सूचना के व्यावहारिक उपयोग से है। जबकि जानकारी को कई कठिनाइयों के बिना परिवहन, संग्रहीत या साझा किया जा सकता है, वही ज्ञान के बारे में नहीं कहा जा सकता है। ज्ञान में एक व्यक्तिगत अनुभव शामिल है। वैज्ञानिक प्रयोग पर चर्चा करते हुए, परिणामों को पढ़ने वाले तीसरे व्यक्ति को इसके बारे में जानकारी होगी, जबकि व्यक्तिगत रूप से प्रयोग करने वाले व्यक्ति को इसके बारे में जानकारी होगी।

डेटाबेस सिस्टम को सूचना के बड़े निकायों के प्रबंधन के लिए डिज़ाइन किया गया है। डेटा के प्रबंधन में सूचनाओं के भंडारण के लिए परिभाषित संरचनाएं और सूचनाओं के हेरफेर के लिए तंत्र प्रदान करना शामिल है। इसके अलावा, डेटाबेस सिस्टम को अनधिकृत पहुंच पर सिस्टम क्रैश या प्रयासों के बावजूद संग्रहीत जानकारी की सुरक्षा सुनिश्चित करनी चाहिए। यदि डेटा को कई उपयोगकर्ताओं के बीच साझा किया जाना है, तो सिस्टम को संभावित विषम परिणामों से बचना चाहिए।

## 1.2. डेटाबेस की आवश्यकता:

एक डेटाबेस डेटा के विभिन्न रूपों का एक संगठित संग्रह है। इसे डेटा के संरचित सेट के रूप में भी जाना जाता है जो आपके कंप्यूटर के माध्यम से कई तरह से सुलभ है। जिन विभिन्न कारणों से हमें डेटाबेस की आवश्यकता होती है वे हैं:-

- डेटा की बड़ी मात्रा को प्रबंधित करने के लिए: हां, आप डेटा को एक स्प्रेडशीट में संग्रहीत कर सकते हैं, लेकिन यदि आप शीट में डेटा के बड़े हिस्से जोड़ते हैं, तो यह बस काम नहीं करेगा। उदाहरण के लिए: यदि आपके डेटा का आकार हजारों रिकॉर्ड में बढ़ता है, तो यह बस गति की समस्या पैदा करेगा।

- सटीकता: एक स्प्रेडशीट में डेटा एंट्री फाइल करते समय, सटीकता को प्रबंधित करना मुश्किल हो जाता है क्योंकि इसमें कोई सत्यापन मौजूद नहीं है।
- डेटा अपडेट करने में आसानी: डेटाबेस के साथ, आप अपनी सुविधा के अनुसार डेटा को लचीले ढंग से अपडेट कर सकते हैं। इसके अलावा, कई लोग एक ही समय में डेटा संपादित भी कर सकते हैं।
- डेटा की सुरक्षा: इस तथ्य से इनकार नहीं किया जाता है कि स्प्रेडशीट में आपका डेटा कम सुरक्षित है। कोई भी आसानी से फ़ाइल तक पहुँच सकता है और उसमें परिवर्तन कर सकता है। डेटाबेस के साथ आपके पास सुरक्षा समूह और विशेषाधिकार हैं जो आपने पहुँच को प्रतिबंधित करने के लिए निर्धारित किए हैं।
- डेटा अखंडता: डेटा अखंडता भी एक सवाल बन जाता है जब स्प्रेडशीट में डेटा संग्रहीत किया जाता है। डेटाबेस में, आप अखंडता की जाँच और अभिगम नियंत्रण में निर्मित होने के कारण सटीकता और डेटा की स्थिरता का आश्वासन दिया जा सकता है।

### 1.3. फ्लैट डेटाबेस:

एक फ्लैट-फ़ाइल डेटाबेस एक डेटाबेस है जिसे एक फ़ाइल में संग्रहीत किया जाता है जिसे एक फ्लैट फ़ाइल कहा जाता है। रिकॉर्ड एक समान प्रारूप का पालन करते हैं, और रिकॉर्ड के बीच संबंधों को अनुक्रमित करने या पहचानने के लिए कोई संरचना नहीं है। फ़ाइल सरल है। एक फ्लैट फ़ाइल एक सादा पाठ फ़ाइल, या एक बाइनरी फ़ाइल हो सकती है। डेटाबेस में डेटा से संबंधों का अनुमान लगाया जा सकता है, लेकिन डेटाबेस प्रारूप ही उन रिश्तों को स्पष्ट नहीं करता है।

#### फ्लैट फ़ाइल मॉडल:

	Route No.	Miles	Activity
Record 1	1-95	12	Overlay
Record 2	1-495	05	Patching
Record 3	SR-301	33	Crack seal

सादा टेक्स्ट फ़ाइलों में आमतौर पर प्रति पंक्ति एक रिकॉर्ड होता है। डेटा को दर्शाने के लिए अलग-अलग परिपाटी होती हैं। अल्पविराम से अलग किए गए मान और सीमांकित-पृथक मान फ़ाइलों में फ़ील्ड को अल्पविराम जैसे अल्पविराम या टैब वर्णों द्वारा अलग किया जा सकता है। अन्य मामलों में, प्रत्येक फ़ील्ड की एक निश्चित लंबाई हो सकती है; लघु मान अंतरिक्ष वर्णों के साथ गद्देदार हो सकते हैं। सीमांकक टकराव से बचने के लिए अतिरिक्त स्वरूपण की आवश्यकता हो सकती है।

फ्लैट फ़ाइलों के उदाहरण फ़ील्ड नाम, पता और फोन नंबर के साथ एक नाम-और-पता सूची है। कागज की शीट पर हाथ से लिखे गए नाम, पते और फोन नंबर की एक सूची एक फ्लैट-फ़ाइल डेटाबेस है। यह किसी भी टाइपराइटर या वर्ड प्रोसेसर के साथ भी किया जा सकता है। एक स्प्रेडशीट

या टेक्स्ट एडिटर प्रोग्राम का उपयोग एक फ्लैट-फाइल डेटाबेस को लागू करने के लिए किया जा सकता है, जो तब बेहतर खोज क्षमताओं के लिए ऑनलाइन मुद्रित या उपयोग किया जा सकता है।

निम्न उदाहरण एक फ्लैट फ़ाइल डेटाबेस के विशिष्ट तत्वों को दिखाता है। डेटा व्यवस्था में स्तंभ और पंक्तियों की एक श्रृंखला होती है जो एक सारणीबद्ध प्रारूप में व्यवस्थित होती है। यह विशिष्ट उदाहरण केवल एक तालिका का उपयोग करता है।

कॉलम में शामिल हैं: नाम (एक व्यक्ति का नाम, दूसरा स्तंभ); टीम (व्यक्ति, तीसरे स्तंभ द्वारा समर्थित एक एथलेटिक टीम का नाम); और एक संख्यात्मक विशिष्ट आईडी, (विशिष्ट रूप से रिकॉर्ड, पहले कॉलम की पहचान करने के लिए उपयोग किया जाता है)।

यहाँ वर्णित डेटा का एक टेक्स्ट बनावट प्रतिनिधित्व है:

Id	NAME	Team
1	Aarti	Blues
2	Boby	Reds
3	Chinki	Blues
4	Ryan	Blues
5	Ela	Reds
6	Franky	Blues
7	Gagan	Blues
8	ari	Reds
9	ari	Blues

एक फ्लैट-फाइल डेटाबेस के लिए इस प्रकार का डेटा प्रतिनिधित्व काफी मानक है, हालांकि कुछ अतिरिक्त विचार हैं जो टेक्स्ट से स्पष्ट रूप से स्पष्ट नहीं हैं:

- डेटा प्रकार: एक डेटाबेस तालिका में प्रत्येक स्तंभ जैसे कि ऊपर एक विशेष डेटा प्रकार के लिए सीमित है। इस तरह के प्रतिबंध आमतौर पर सम्मेलन द्वारा स्थापित किए जाते हैं, लेकिन

औपचारिक रूप से संकेत नहीं दिया जाता है जब तक कि डेटा एक रिलेशनल डेटाबेस सिस्टम में स्थानांतरित नहीं होता है।

- अलग कॉलम: उपरोक्त उदाहरण में, व्हाट्सएप पात्रों का उपयोग करके अलग-अलग कॉलम को अलग किया जाता है। इसे इंडेंटेशन या "निश्चित-चौड़ाई" डेटा स्वरूपण भी कहा जाता है। एक और आम सम्मेलन एक या एक से अधिक सीमांत वर्णों का उपयोग करके स्तंभों को अलग करना है, जैसे कि टैब या अल्पविराम।
- संबंधपरक बीजगणित: उपर्युक्त तालिका में प्रत्येक पंक्ति या रिकॉर्ड संबंधपरक बीजगणित के तहत एक टपल की मानक परिभाषा को पूरा करता है (उपरोक्त उदाहरण में 3- टपल की एक श्रृंखला को दर्शाया गया है)। साथ ही, पहली पंक्ति फ़ील्ड नामों को निर्दिष्ट करती है जो प्रत्येक पंक्ति के मानों से जुड़े होते हैं।
- डेटाबेस प्रबंधन प्रणाली: चूंकि एक पाठ फ़ाइल के साथ औपचारिक संचालन आमतौर पर वांछित से अधिक सीमित होता है, इसलिए उपरोक्त उदाहरण में पाठ आमतौर पर डेटाबेस प्रबंधन प्रणाली में स्थानांतरित होने से पहले डेटा के एक मध्यस्थ राज्य का प्रतिनिधित्व करेगा।

#### 1.4. डेटाबेस प्रबंधन प्रणाली

DBMS का अर्थ है डेटाबेस मैनेजमेंट सिस्टम। हम इसे इस तरह तोड़ सकते हैं: DBMS = Database + Management System. डेटाबेस डेटा का एक संग्रह है और प्रबंधन प्रणाली उन डेटा को संग्रहीत और पुनर्प्राप्त करने के लिए कार्यक्रमों का एक समूह है। इसके आधार पर हम DBMS को इस तरह परिभाषित कर सकते हैं: DBMS अंतर-संबंधित डेटा का संग्रह है और उन डेटा को आसान और प्रभावी तरीके से संग्रहीत और एक्सेस करने के लिए कार्यक्रमों का सेट है।

डेटाबेस सिस्टम मूल रूप से बड़ी मात्रा में डेटा के लिए विकसित किए जाते हैं। भारी मात्रा में डेटा के साथ काम करते समय, दो चीजें हैं जो अनुकूलन की आवश्यकता होती हैं: डेटा का भंडारण और डेटा की पुनर्प्राप्ति।

भंडारण: डेटाबेस सिस्टम के सिद्धांतों के अनुसार, डेटा को इस तरह से संग्रहीत किया जाता है कि यह बहुत कम जगह प्राप्त करता है क्योंकि भंडारण से पहले अनावश्यक डेटा (डुप्लिकेट डेटा) को हटा देते हैं।

डेटा की फास्ट पुनर्प्राप्ति: डेटा को अनुकूलित और व्यवस्थित तरीके से संग्रहीत करने के साथ-साथ यह भी महत्वपूर्ण है कि हम डेटा को जरूरत पड़ने पर जल्दी से पुनर्प्राप्त करें। डेटाबेस सिस्टम यह सुनिश्चित करता है कि डेटा को जल्द से जल्द पुनर्प्राप्त किया जाए।

डेटाबेस सिस्टम का मुख्य उद्देश्य डेटा का प्रबंधन करना है। एक विश्वविद्यालय पर विचार करें जो छात्रों, शिक्षकों, पाठ्यक्रमों, पुस्तकों आदि का डेटा रखता है। इस डेटा को प्रबंधित करने के लिए, हमें इस डेटा को कहीं स्टोर करना होगा जहां हम नया डेटा जोड़ सकते हैं, अप्रयुक्त डेटा को हटा सकते हैं, पुराने डेटा को अपडेट कर सकते हैं, डेटा पुनः प्राप्त कर सकते हैं। डेटा पर इन ऑपरेशनों को करने के लिए हमें एक डेटाबेस प्रबंधन प्रणाली की आवश्यकता होती है जो हमें डेटा को इस तरह संग्रहीत करने की अनुमति देता है ताकि इन सभी ऑपरेशनों को कुशलतापूर्वक डेटा पर किया जा सके।

#### 1.4.1. डेटाबेस अनुप्रयोग - DBMS

अनुप्रयोग जहाँ हम डेटाबेस प्रबंधन प्रणाली का उपयोग करते हैं:

- टेलीकॉम: एक कॉल, नेटवर्क उपयोग, ग्राहक विवरण आदि के बारे में जानकारी का ट्रैक रखने के लिए एक डेटाबेस है। डेटाबेस सिस्टम के बिना डेटा की उस विशाल मात्रा को बनाए रखना कठिन है जो हर मिलीसेकंड को अपडेट करता रहता है।

- उद्योग: जहां यह एक विनिर्माण इकाई, गोदाम या वितरण केंद्र है, प्रत्येक को इन्स और आउट्स के रिकॉर्ड रखने के लिए डेटाबेस की आवश्यकता होती है
- बैंकिंग प्रणाली: ग्राहक जानकारी संग्रहीत करने के लिए, दिन-प्रतिदिन के क्रेडिट और डेबिट लेनदेन पर नज़र रखने, बैंक स्टेटमेंट बनाने आदि के लिए। यह सब काम डेटाबेस प्रबंधन प्रणालियों की मदद से किया गया है।
- बिक्री: ग्राहक की जानकारी, उत्पादन जानकारी और चालान विवरण संग्रहीत करने के लिए
- एयरलाइंस: एयरलाइंस की यात्रा करने के लिए, हम शुरुआती आरक्षण करते हैं, फ्लाइट शेड्यूल के साथ आरक्षण की जानकारी डेटाबेस में संग्रहीत की जाती है।
- शिक्षा क्षेत्र: डेटाबेस सिस्टम अक्सर स्कूल और कॉलेजों में छात्र के विवरण, स्टाफ विवरण, पाठ्यक्रम विवरण, परीक्षा विवरण, पेरोल डेटा, उपस्थिति विवरण, फीस विवरण आदि के बारे में डेटा संग्रहीत और पुनर्प्राप्त करने के लिए उपयोग किया जाता है।
- ऑनलाइन शॉपिंग: आपको ऑनलाइन शॉपिंग वेबसाइट जैसे अमेज़न, फ्लिपकार्ट आदि के बारे में पता होना चाहिए। ये साइटें उत्पाद जानकारी, आपके पते और वरीयताओं, क्रेडिट विवरणों को संग्रहीत करती हैं और आपको आपकी क्वेरी के आधार पर उत्पादों की प्रासंगिक सूची प्रदान करती हैं। यह सब एक डेटाबेस प्रबंधन प्रणाली शामिल है।

#### 1.4.2. फाइल सिस्टम पर डीबीएमएस का लाभ

फाइल सिस्टम पर डेटाबेस प्रबंधन प्रणाली के कई फायदे हैं। उनमें से कुछ इस प्रकार हैं:

- कोई अनुपयोगी डेटा नहीं: अनुपयोगी डेटा सामान्यीकरण द्वारा हटाए। कोई डेटा दोहराव भंडारण को बचाता है और पहुंच के समय में सुधार करता है।

- डेटा संगति और अखंडता: डेटा असंगति का मूल कारण डेटा अनुपयोगिता है। चूंकि डेटा सामान्यीकरण अनुपयोगी डेटा रखता है, इसलिए डेटा असंगति को भी इसके भाग के रूप में ध्यान रखा गया है।
- डेटा सुरक्षा: डेटाबेस सिस्टम में एक्सेस बाधाओं को लागू करना आसान है ताकि केवल अधिकृत उपयोगकर्ता ही डेटा का उपयोग करने में सक्षम हो। प्रत्येक उपयोगकर्ता की पहुंच अलग-अलग होती है। इस प्रकार डेटा को पहचान की चोरी, डेटा लीक और डेटा के दुरुपयोग जैसे मुद्दों से सुरक्षित किया जाता है।
- गोपनीयता: सीमित पहुंच का मतलब डेटा की गोपनीयता है।
- डेटा तक आसान पहुंच - डेटाबेस सिस्टम डेटा को इस तरह से प्रबंधित करता है ताकि डेटा तेजी से प्रतिक्रिया समय के साथ आसानी से सुलभ हो।
- आसान पुनर्प्राप्ति: चूंकि डेटाबेस सिस्टम डेटा का बैकअप रखता है, इसलिए विफलता की स्थिति में डेटा की पूर्ण वसूली करना आसान होता है।
- लचीले: फ़ाइल प्रोसेसिंग सिस्टम की तुलना में डेटाबेस सिस्टम अधिक लचीले होते हैं।

#### 1.4.3. DBMS के नुकसान:

- फ़ाइल सिस्टम की तुलना में DBMS कार्यान्वयन लागत अधिक है
- जटिलता: डेटाबेस सिस्टम समझने के लिए जटिल हैं
- प्रदर्शन: डेटाबेस सिस्टम सामान्य हैं, जिससे वे विभिन्न अनुप्रयोगों के लिए उपयुक्त हैं। हालाँकि यह सुविधा कुछ अनुप्रयोगों के लिए उनके प्रदर्शन को प्रभावित करती है।

#### 1.4.4. DBMS आर्किटेक्चर

डेटाबेस मैनेजमेंट सिस्टम आर्किटेक्चर हमें डेटाबेस सिस्टम के घटकों और उनके बीच संबंध को समझने में मदद करेगा।

डीबीएमएस का आर्किटेक्चर उस कंप्यूटर प्रणाली पर निर्भर करती है जिस पर वह चलता है। उदाहरण के लिए, क्लाइंट-सर्वर DBMS आर्किटेक्चर में, सर्वर मशीन पर डेटाबेस सिस्टम क्लाइंट मशीन द्वारा किए गए कई अनुरोधों को चला सकते हैं। हम इस संचार को आरेखों की सहायता से समझेंगे।

#### DBMS आर्किटेक्चर के प्रकार

DBMS आर्किटेक्चर के तीन प्रकार हैं:

1. एकल स्तरीय आर्किटेक्चर
2. दो स्तरीय आर्किटेक्चर
3. तीन स्तरीय आर्किटेक्चर

#### 1. एकल स्तरीय आर्किटेक्चर

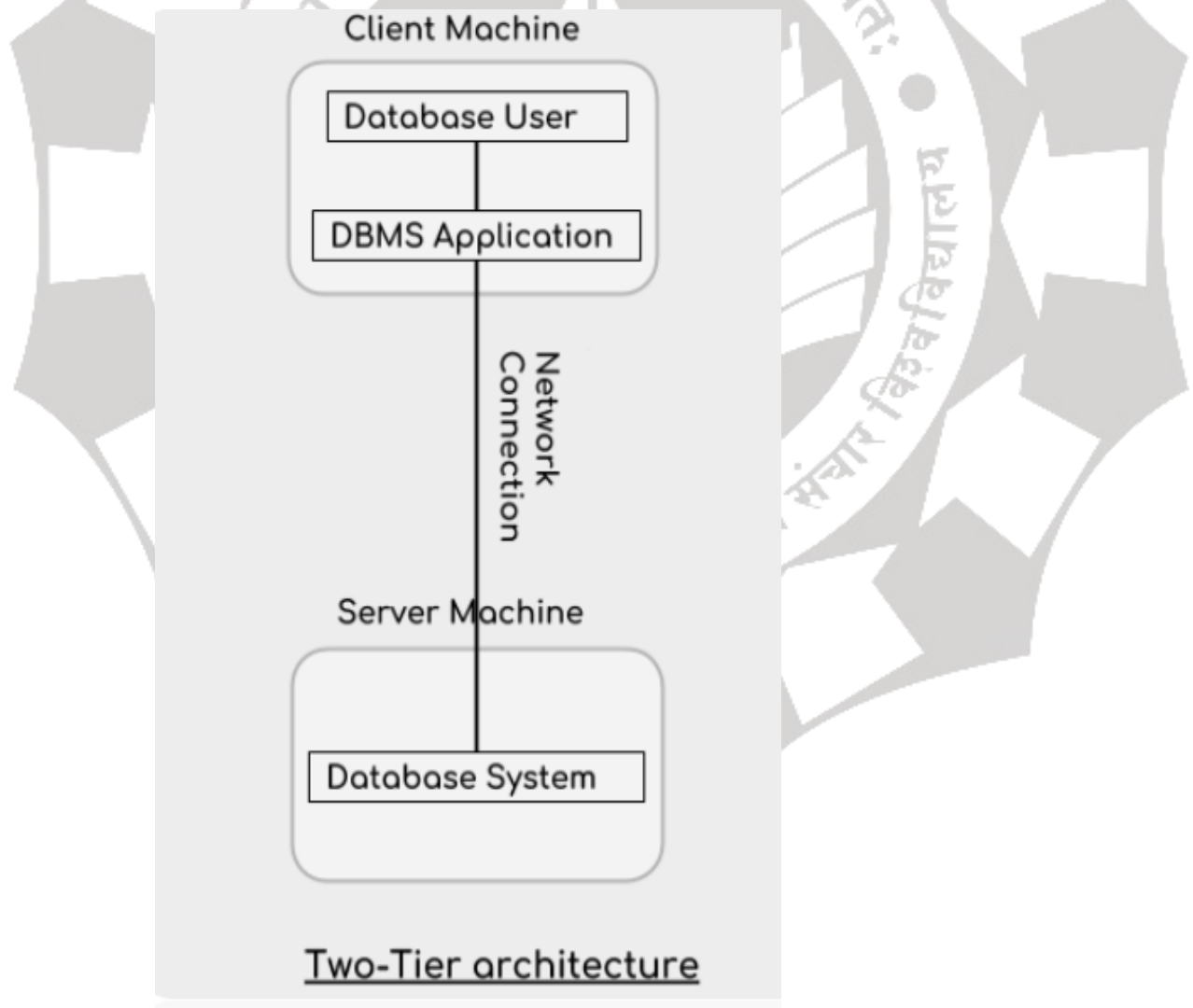
इस प्रकार की वास्तुकला में, डेटाबेस ग्राहक मशीन पर आसानी से उपलब्ध है। क्लाइंट द्वारा किए गए किसी भी अनुरोध के लिए डेटाबेस पर कार्रवाई करने के लिए नेटवर्क कनेक्शन की आवश्यकता नहीं होती है।

उदाहरण के लिए, आप डेटाबेस से कर्मचारी के रिकॉर्ड प्राप्त करना चाहते हैं और डेटाबेस आपके कंप्यूटर सिस्टम पर उपलब्ध है। तो कर्मचारी विवरण प्राप्त करने का अनुरोध आपके कंप्यूटर द्वारा किया जाएगा और रिकॉर्ड आपके कंप्यूटर द्वारा डेटाबेस से प्राप्त किया जाएगा। इस प्रकार की प्रणाली को आमतौर पर स्थानीय डेटाबेस प्रणाली के रूप में जाना जाता है।

## 2. दो स्तरीय आर्किटेक्चर

द्वि-स्तरीय वास्तुकला में, डेटाबेस सिस्टम सर्वर मशीन पर मौजूद है और डीबीएमएस एप्लिकेशन क्लाइंट मशीन में मौजूद है। इन दो मशीनों को एक विश्वसनीय नेटवर्क के माध्यम से एक दूसरे के साथ जोड़ा जाता है जैसा कि नीचे आरेख में दिखाया गया है।

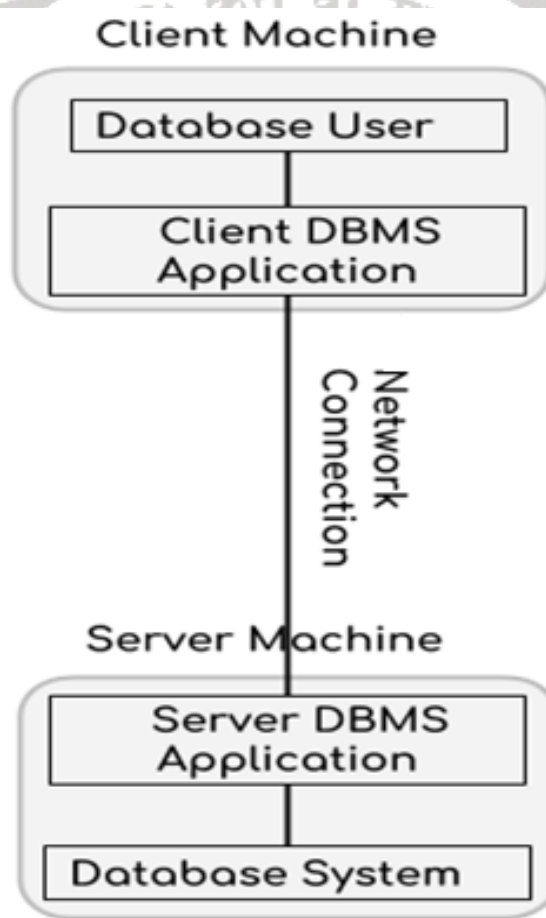
जब भी क्लाइंट मशीन SQL भाषा की तरह क्वेरी भाषा का उपयोग करके सर्वर पर मौजूद डेटाबेस तक पहुंचने का अनुरोध करती है, तो सर्वर डेटाबेस पर अनुरोध करता है और परिणाम को क्लाइंट को वापस लौटा देता है। JDBC, ODBC जैसे एप्लिकेशन कनेक्शन इंटरफ़ेस का उपयोग सर्वर और क्लाइंट के बीच बातचीत के लिए किया जाता है।



### 3. तीन स्तरीय आर्किटेक्चर:

श्री-टियर आर्किटेक्चर में, क्लाइंट मशीन और सर्वर मशीन के बीच एक और परत मौजूद होती है। इस आर्किटेक्चर में, क्लाइंट एप्लिकेशन सर्वर मशीन में मौजूद डेटाबेस सिस्टम के साथ सीधे संवाद नहीं करता है। बल्कि क्लाइंट एप्लिकेशन सर्वर एप्लिकेशन के साथ संचार करता है और सर्वर एप्लिकेशन आंतरिक रूप से सर्वर पर मौजूद डेटाबेस सिस्टम के साथ संचार करता है।

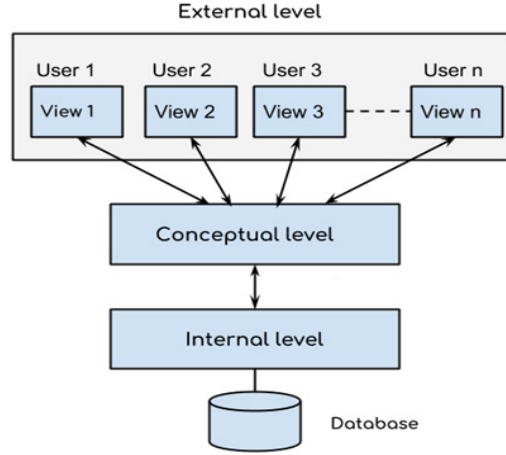
#### डीबीएमएस श्री लेवल आर्किटेक्चर डायग्राम



Three-Tier architecture

इस आर्किटेक्चर के तीन स्तर हैं:

1. बाहरी स्तर
2. वैचारिक स्तर
3. आंतरिक स्तर



### 1. बाहरी स्तर

इसे व्यू लेवल भी कहा जाता है। इस स्तर को "दृश्य" कहा जाता है क्योंकि कई उपयोगकर्ता इस स्तर से अपना वांछित डेटा देख सकते हैं। इस वांछित डेटा को वैचारिक और आंतरिक स्तर मानचित्रण की मदद से डेटाबेस से आंतरिक रूप से प्राप्त किया जाता है।

उपयोगकर्ता को डेटाबेस स्कीमा विवरण जैसे डेटा संरचना, तालिका परिभाषा आदि जानने की आवश्यकता नहीं है। उपयोगकर्ता केवल डेटा के बारे में चिंतित है, जो डेटाबेस से प्राप्त होने के बाद (आंतरिक स्तर पर मौजूद) दृश्य स्तर पर वापस आ गया है।

बाह्य स्तर तीन स्तर DBMS आर्किटेक्चर का "शीर्ष स्तर" है।

### 2. वैचारिक स्तर

इसे तार्किक स्तर भी कहा जाता है। डेटाबेस का पूरा डिज़ाइन जैसे डेटा, डेटा का स्कीमा आदि के बीच संबंध इस स्तर में वर्णित हैं।

इस स्तर की आर्किटेक्चर में डेटाबेस की कमी और सुरक्षा को भी लागू किया जाता है। यह स्तर डीबीए (डेटाबेस प्रशासक) द्वारा बनाए रखा जाता है।

### 3. आंतरिक स्तर

इस स्तर को भौतिक स्तर के रूप में भी जाना जाता है। यह स्तर बताता है कि डेटा वास्तव में भंडारण उपकरणों में कैसे संग्रहीत किया जाता है। यह स्तर डेटा को स्थान आवंटित करने के लिए भी जिम्मेदार है। यह आर्किटेक्चर का सबसे निचला स्तर है।

## 1.5 डेटाबेस प्रबंधन प्रणाली के लक्षण

परंपरागत रूप से, डेटा फ़ाइल स्वरूपों में आयोजित किया गया था। डीबीएमएस (DBMS) तब एक नई अवधारणा थी, और यह सभी अनुसंधान डेटा प्रबंधन की पारंपरिक शैली में कमियों को दूर करने के लिए किया गया था। एक आधुनिक डीबीएमएस (DBMS) की निम्नलिखित विशेषताएं हैं -

- वास्तविक दुनिया की इकाई - एक आधुनिक डीबीएमएस (DBMS) अधिक यथार्थवादी है और इसकी आर्किटेक्चर को डिजाइन करने के लिए वास्तविक दुनिया की संस्थाओं का उपयोग करता है। यह व्यवहार और विशेषताओं का भी उपयोग करता है। उदाहरण के लिए, एक स्कूल डेटाबेस छात्रों को एक इकाई और उनकी उम्र एक एट्रिब्यूट के रूप में उपयोग कर सकता है।
- रियल - वर्ल्ड एंटीटी - डीबीएमएस (DBMS) तालिकाओं को बनाने के लिए संस्थाओं और रिलेशनस की अनुमति देता है। एक उपयोगकर्ता डेटाबेस के आर्किटेक्चर को केवल तालिका के नामों को देखकर समझ सकता है।
- रिलेशन - बेस्ड टेबल्स - एक डेटाबेस प्रणाली इसके डेटा से पूरी तरह से अलग है। एक डेटाबेस एक सक्रिय इकाई है। जबकि डेटा को निष्क्रिय कहा जाता है, जिस पर डेटाबेस काम करता है और व्यवस्थित करता है। डीबीएमएस (DBMS) अपनी स्वयं की प्रक्रिया को आसान बनाने के लिए मेटाडेटा को संग्रहीत करता है, जो डेटा के बारे में डेटा है।
- लेस रिडनडनसी - डीबीएमएस सामान्यीकरण के नियमों का पालन करता है, जो किसी भी एट्रिब्यूट को मूल्यों में रेडूण्डनस्य होने पर संबंध को विभाजित करता है। सामान्यीकरण एक गणितीय रूप से समृद्ध और वैज्ञानिक प्रक्रिया है जो डेटा रेडूण्डनस्य को कम करती है।
- कंसिस्टेंसी - कंसिस्टेंसी एक अवस्था है जहाँ एक डेटाबेस में प्रत्येक संबंध सुसंगत रहता है। ऐसी विधियाँ और तकनीकें मौजूद हैं, जो असंगत (इन्कन्सीस्टेंट) अवस्था में डेटाबेस छोड़ने के प्रयास का पता लगा सकती हैं। एक डीबीएमएस (DBMS) फाइल-प्रोसेसिंग सिस्टम जैसे डेटा भंडारण अनुप्रयोगों के पुराने रूपों की तुलना में अधिक स्थिरता प्रदान कर सकता है।
- क्वेरी लैंग्वेज - डीबीएमएस (DBMS) क्वेरी लैंग्वेज से सुसज्जित है, जो डेटा को पुनः प्राप्त करने और हेरफेर करने के लिए अधिक कुशल बनाता है। एक उपयोगकर्ता डेटा के एक सेट को पुनः प्राप्त करने के लिए आवश्यक के रूप में कई और विभिन्न फ़िल्टरिंग विकल्पों के रूप में अप्लाइ कर सकता है। परंपरागत रूप से यह संभव नहीं था जहां फ़ाइल-प्रोसेसिंग सिस्टम का उपयोग किया गया था।
- • ACID गुण - डीबीएमएस (DBMS) एटमॉसिटी, कंसिस्टेंसी, आइसोलेशन और स्थायित्व (ड्युराबिलिटी) की अवधारणाओं का अनुसरण करता है (आमतौर पर ACID के रूप में छोटा किया जाता है)। इन अवधारणाओं को लेनदेन पर लागू किया जाता है, जो एक डेटाबेस के डेटा में हेरफेर करते हैं। ACID गुण डेटाबेस को बहु-लेन-देन के वातावरण में और विफलता के मामले में ठीक रहने में मदद करते हैं।

- मल्टीयूजर एंड कंकरेंट एक्सेस - डीबीएमएस (DBMS) मल्टीयूजर वातावरण का समर्थन करता है और उन्हें समानांतर में डेटा तक पहुंचने और हेरफेर करने की अनुमति देता है। हालांकि लेन-देन पर प्रतिबंध हैं जब उपयोगकर्ता एक ही डेटा आइटम को संभालने का प्रयास करते हैं, लेकिन उपयोगकर्ता हमेशा उनसे अनजान होते हैं।
- मल्टीप्ल व्यूज - डीबीएमएस (DBMS) विभिन्न उपयोगकर्ताओं के लिए कई विचार प्रस्तुत करता है। एक उपयोगकर्ता जो बिक्री विभाग में है, उत्पादन विभाग में काम करने वाले व्यक्ति की तुलना में डेटाबेस का एक अलग दृष्टिकोण होगा। यह सुविधा उपयोगकर्ताओं को उनकी आवश्यकताओं के अनुसार डेटाबेस का एक केंद्रित दृष्टिकोण रखने में सक्षम बनाती है।
- सुरक्षा - कई विचारों की तरह सुविधाएँ कुछ हद तक सुरक्षा प्रदान करती हैं जहां उपयोगकर्ता अन्य उपयोगकर्ताओं और विभागों के डेटा तक पहुंचने में असमर्थ हैं। डीबीएमएस डेटाबेस में डेटा दर्ज करते हुए और बाद के चरण में समान रूप में प्राप्त करने के लिए बाध्यताएं लागू करने के तरीके प्रदान करता है। डीबीएमएस (DBMS) सुरक्षा सुविधाओं के कई अलग-अलग स्तर प्रदान करता है, जो कई उपयोगकर्ताओं को विभिन्न विशेषताओं के साथ अलग-अलग विचार रखने में सक्षम बनाता है। उदाहरण के लिए, बिक्री विभाग का एक उपयोगकर्ता उस डेटा को नहीं देख सकता जो खरीद विभाग का है। इसके अतिरिक्त, यह भी प्रबंधित किया जा सकता है कि उपयोगकर्ता को बिक्री विभाग का कितना डेटा प्रदर्शित किया जाना चाहिए। चूंकि DBMS डिस्क पर पारंपरिक फ़ाइल सिस्टम के रूप में सहेजा नहीं गया है, इसलिए शरारती तत्वों के लिए कोड को तोड़ना बहुत कठिन है।

## 1.6 रिलेशनल डेटाबेस

एक रिलेशनल डेटाबेस एक डेटाबेस को संदर्भित करता है जो पंक्तियों और स्तंभों का उपयोग करके स्ट्रक्चर्ड प्रारूप में डेटा संग्रहीत करता है। यह डेटाबेस के भीतर विशिष्ट मूल्यों का पता लगाने और उन तक पहुंचने में आसान बनाता है। यह " रिलेशनल " है क्योंकि प्रत्येक तालिका के भीतर मान एक दूसरे से संबंधित हैं।

इसे रिलेशनल डेटा बेस मैनेजमेंट सिस्टम (RDBMS) कहा जाता है क्योंकि यह E.F. Codd द्वारा पेश किए गए रिलेशनल मॉडल पर आधारित है। SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL और Microsoft Access जैसे सभी आधुनिक डेटाबेस प्रबंधन सिस्टम RDBMS पर आधारित हैं। आरडीबीएमएस (RDBMS) में Tuples (पंक्तियों) के संदर्भ में डेटा का प्रतिनिधित्व किया जाता है। रिलेशनल डेटाबेस सबसे अधिक इस्तेमाल किया जाने वाला डेटाबेस है। इसमें तालिकाओं की संख्या होती है और प्रत्येक तालिका की अपनी प्राथमिक (प्राइमरी) कुंजी होती है। तालिकाओं के संगठित सेट के संग्रह के कारण, डेटा को आरडीबीएमएस में आसानी से एक्सेस किया जा सकता है।

## टेबल क्या है

RDBMS डेटाबेस डेटा स्टोर करने के लिए तालिकाओं का उपयोग करता है। एक तालिका संबंधित डेटा प्रविष्टियों का एक संग्रह है और इसमें डेटा संग्रहीत करने के लिए पंक्तियाँ और स्तंभ हैं। RDBMS में डेटा स्टोरेज का सबसे सरल उदाहरण एक टेबल है।

छात्र तालिका का उदाहरण देखते हैं।

**STUDENT TABLE**

ID	NAME	AGE	COURSE
1	Ajeet	24	B.Tech
2	aryan	20	C.A
3	Mahesh	21	BCA
4	Ratan	22	MCA
5	Vimal	26	BSC

## फ़ील्ड / पंक्ति क्या है

तालिका की एक पंक्ति को रिकॉर्ड भी कहा जाता है। इसमें तालिका में प्रत्येक व्यक्तिगत प्रविष्टि की विशिष्ट जानकारी शामिल है। यह तालिका में एक क्षैतिज (हॉरिजॉन्टल) इकाई है। उदाहरण के लिए: उपरोक्त तालिका में 5 रिकॉर्ड हैं।

आइए तालिका में एक रिकॉर्ड / पंक्ति देखें।

1	Ajeet	24	B.Tech
---	-------	----	--------

## कॉलम क्या है

एक स्तंभ तालिका में एक ऊर्ध्वाधर (वर्टिकल) इकाई है जिसमें एक तालिका में एक विशिष्ट क्षेत्र से जुड़ी सभी जानकारी होती है। उदाहरण के लिए: "नाम" उपरोक्त तालिका में एक कॉलम है जिसमें छात्र के नाम के बारे में सभी जानकारी है।

NAME
Ajeet
Aryan
Mahesh
Ratan
Vimal

## NULL मान क्या है?

तालिका में एक NULL मान एक फ़ील्ड में एक मान है जो रिक्त दिखाई देता है, जिसका अर्थ है कि NULL मान वाला फ़ील्ड बिना मान वाला फ़ील्ड है।

यह समझना बहुत महत्वपूर्ण है कि एक शून्य मान शून्य मान या एक फ़ील्ड से भिन्न होता है जिसमें रिक्त स्थान होते हैं। NULL मान वाला फ़ील्ड वह है जिसे रिकॉर्ड निर्माण के दौरान खाली छोड़ दिया गया है।

## डोमेन क्या है?

एक डोमेन तालिका में एक एट्रिब्यूट के लिए अनुमत मूल्यों का एक समूह है। उदाहरण के लिए, महीने के वर्ष का एक डोमेन जनवरी, फरवरी, ... दिसंबर को मानों के रूप में स्वीकार कर सकता है, तिथियों का एक डोमेन सभी संभावित मान्य तिथियों को स्वीकार कर सकता है। हम तालिका बनाते समय एट्रिब्यूट का डोमेन उल्लिखित करते हैं।

एक एट्रिब्यूट उन मानों को स्वीकार नहीं कर सकती है जो उनके डोमेन के बाहर हैं। उदाहरण के लिए, उपरोक्त तालिका में "छात्र" आईडी क्षेत्र में पूर्णांक डोमेन है ताकि क्षेत्र उन मूल्यों को स्वीकार न कर सके जो उदाहरण के लिए पूर्णांक नहीं हैं, आईडी में मान नहीं हो सकते हैं, जैसे "प्रथम", 10.11 आदि।

RDBMS एक DBMS है जो विशेष रूप से रिलेशनल डेटाबेस के लिए डिज़ाइन किया गया है। इसलिए RDBMSes, DBMSes का सबसेट है। चूंकि MySQL एक संबंधपरक डेटाबेस है, इसलिए यह डेटाबेस के पूरे डेटा को एक तालिका के बजाए अलग अलग तालिकाओं में स्टोर कर रहा है। इन तालिकाओं के बीच में हम अलग-अलग संबंध भी स्थापित कर सकते हैं, और अपनी जरूरत के अनुसार इन अलग-अलग प्रश्नों के अनुसार इन तालिकाओं पर डेटा को लाने, सम्मिलित करने, या अपडेट करने के लिए कर सकते हैं।

## DBMS और RDBMS के बीच अंतर

हालांकि DBMS और RDBMS दोनों का उपयोग भौतिक डेटाबेस में जानकारी संग्रहीत करने के लिए किया जाता है, लेकिन उनके बीच कुछ उल्लेखनीय अंतर हैं।

DBMS और RDBMS के बीच मुख्य अंतर नीचे दिए गए हैं:

	DBMS (डीबीएमएस)	RDBMS (आरडीबीएमएस)
1.	DBMS एप्लिकेशन डेटा को फ़ाइल के रूप में संग्रहीत करता है।	RDBMS एप्लिकेशन डेटा को सारणीबद्ध रूप में संग्रहीत करता है।
2.	DBMS में, डेटा आमतौर पर या तो एक पदानुक्रमित रूप या एक नेविगेशनल रूप में संग्रहीत किया जाता है।	RDBMS में, तालिकाओं में एक पहचानकर्ता होता है जिसे प्राथमिक (प्राइमरी) कुंजी कहा जाता है और डेटा मान को तालिकाओं के रूप में संग्रहीत किया जाता है।

3.	डीबीएमएस में सामान्यीकरण (Normalization) मौजूद नहीं है।	आरडीबीएमएस में सामान्यीकरण (Normalization) मौजूद है।
4.	DBMS डेटा हेरफेर (मैनीपुलेशन) के संबंध में कोई सुरक्षा लागू नहीं करता है।	RDBMS ACID (एटमोसिटी, कंसिस्टेंसी, आइसोलेशन और ड्यूरेबिलिटी) संपत्ति के उद्देश्य के लिए अखंडता बाधा को परिभाषित करता है।
5.	DBMS डेटा को स्टोर करने के लिए फ़ाइल सिस्टम का उपयोग करता है, इसलिए तालिकाओं के बीच कोई संबंध नहीं होगा।	RDBMS में, डेटा मान को तालिकाओं के रूप में संग्रहीत किया जाता है, इसलिए इन डेटा मानों के बीच संबंध तालिका के रूप में भी संग्रहीत किया जाएगा।
6.	DBMS को संग्रहीत जानकारी तक पहुंचने के लिए कुछ समान तरीके प्रदान करने होंगे।	RDBMS सिस्टम संग्रहीत जानकारी तक पहुंचने के लिए डेटा की एक सारणीबद्ध संरचना और उनके बीच संबंध का समर्थन करता है।
7.	DBMS वितरित डेटाबेस का समर्थन नहीं करता है।	RDBMS वितरित डेटाबेस का समर्थन करता है।
8.	DBMS छोटे संगठन के लिए और छोटे डेटा से निपटने के लिए है। यह सिंगल यूजर को सपोर्ट करता है।	RDBMS को बड़ी मात्रा में डेटा को संभालने के लिए डिज़ाइन किया गया है। यह कई (multiple) उपयोगकर्ताओं का समर्थन करता है।
9.	DBMS के उदाहरण फ़ाइल सिस्टम, xml आदि हैं।	RDBMS का उदाहरण mysql, postgre, sql server, oracle आदि हैं।

### 1.7 डेटाबेस में प्राथमिक कुंजी की अवधारणा

डेटाबेस की दुनिया में, एक प्राथमिक कुंजी एक तालिका में एक फ़ील्ड है जो डेटाबेस तालिका में प्रत्येक पंक्ति / रिकॉर्ड को विशिष्ट रूप से पहचानती है। प्राथमिक कुंजी में अद्वितीय (unique) मान होना चाहिए। प्राथमिक कुंजी कॉलम में NULL मान नहीं हो सकते।

डेटाबेस कुंजियों का उपयोग रिकॉर्ड्स की तुलना, सॉर्ट और स्टोर करने और रिकॉर्ड के बीच संबंध बनाने के लिए करता है।

एक डेटाबेस में प्राथमिक कुंजी चुनना प्रक्रिया में सबसे महत्वपूर्ण चरणों में से एक है। यह एक सामान्य विशेषता हो सकती है जो अद्वितीय होने की गारंटी है जैसे: सामाजिक सुरक्षा संख्या एक मेज पर प्रति व्यक्ति एक से अधिक रिकॉर्ड नहीं है। या विशेषतः (preferably) यह डेटाबेस प्रबंधन प्रणाली द्वारा उत्पन्न किया जा सकता है जैसे: Microsoft SQL सर्वर में विश्व स्तर पर विशिष्ट पहचानकर्ता,

या GUID। प्राथमिक कुंजियाँ संयोजन में एकल विशेषता या एकाधिक विशेषताओं से युक्त हो सकती हैं।

एक तालिका में केवल एक प्राथमिक कुंजी हो सकती है, जिसमें एकल या एकाधिक(multiple) फ़ील्ड शामिल हो सकते हैं। जब कई क्षेत्रों को प्राथमिक कुंजी के रूप में उपयोग किया जाता है, तो उन्हें समग्र(composite) कुंजी कहा जाता है।

यदि किसी तालिका में किसी भी फ़ील्ड पर कोई प्राथमिक कुंजी निर्धारित है, तो आपके पास उस फ़ील्ड के समान मूल्य वाले दो रिकॉर्ड नहीं हो सकते हैं।

प्राथमिक कुंजी अन्य तालिकाओं में संबंधित जानकारी के लिए अद्वितीय लिंक हैं जहां प्राथमिक कुंजी का उपयोग किया जाता है। रिकॉर्ड बनाते समय इसे दर्ज किया जाना चाहिए, और इसे कभी नहीं बदलना चाहिए। डेटाबेस में प्रत्येक तालिका में विशेष रूप से प्राथमिक कुंजी के लिए एक स्तंभ या दो होते हैं।

प्राथमिक कुंजी का उदाहरण:

कल्पना कीजिए कि आपके पास एक छात्र तालिका है जिसमें एक विश्वविद्यालय में प्रत्येक छात्र के लिए एक रिकॉर्ड है। छात्र की अद्वितीय छात्र आईडी संख्या छात्रों की तालिका में एक प्राथमिक कुंजी के लिए एक अच्छा विकल्प है। छात्र का पहला और अंतिम नाम अच्छे विकल्प नहीं हैं क्योंकि हमेशा यह संयोग होता है कि एक से अधिक छात्रों का नाम एक ही हो सकता है।

## 1.8 एंटीटीज (Entities)

एंटीटीज प्रमुख डेटा ऑब्जेक्ट हैं जिनके बारे में जानकारी एकत्र की जाती है। प्रविष्टियाँ आमतौर पर पहचानने योग्य अवधारणाएँ (concepts) होती हैं, या तो ठोस या अमूर्त, जैसे: व्यक्ति, स्थान, चीजें, या घटनाएँ जिनकी डेटाबेस में प्रासंगिकता होती है। संस्थाओं के कुछ विशिष्ट उदाहरण कर्मचारी, परियोजनाएँ, निवेश हैं। एक इकाई संबंधपरक मॉडल में एक तालिका के अनुरूप है।

एंटीटीज को मजबूत और कमजोर के रूप में वर्गीकृत किया गया है। एक स्वतंत्र (मजबूत) इकाई वह है जो पहचान के लिए दूसरे पर निर्भर नहीं है। एक आश्रित (कमजोर) इकाई वह है जो पहचान के लिए दूसरे पर निर्भर है। एक एंटीटी घटित होना (जिसे एक उदाहरण भी कहा जाता है) एक एंटीटी की एक अकेली घटना है। एक घटना संबंधपरक (relational) तालिका में एक पंक्ति के अनुरूप है।

विशेष एंटीटी प्रकार एसोसिएटिव इकाइयाँ (जिसे चौराहे संस्थाओं के रूप में भी जाना जाता है) कई-से-कई संबंधों को समेटने के लिए दो या दो से अधिक संस्थाओं को जोड़ने के लिए उपयोग की जाने वाली इकाइयाँ हैं। उपप्रकार इकाइयाँ का उपयोग सामान्यीकरण पदानुक्रम में उनके मूल इकाई के उदाहरणों के सबसेट का प्रतिनिधित्व करने के लिए किया जाता है, जिसे सुपरटाइप कहा जाता है। लेकिन जिनके गुण या संबंध हैं जो केवल सबसेट पर लागू होते हैं।

## 1.9 रेफरेंशियल इंटीग्रिटी (Referential Integrity)

एक फॉरेन कुंजी एक तालिका में एक पहचानकर्ता (identifier) है जो एक अलग तालिका की प्राथमिक कुंजी से मेल खाती है। फॉरेन कुंजी एक अलग तालिका के साथ संबंध बनाती है। रेफरेंशियल इंटीग्रिटी (Referential integrity) इन तालिकाओं के बीच संबंध को संदर्भित (refers) करता है।

जब एक टेबल में दूसरी टेबल के लिए एक फॉरेन कुंजी होती है। रेफरेंशियल इंटीग्रिटी (Referential Integrity) की अवधारणा में कहा गया है कि: आप उस तालिका में रिकॉर्ड नहीं जोड़ सकते हैं जिसमें फॉरेन कुंजी शामिल है जब तक कि लिंक की गई तालिका में एक समान रिकॉर्ड न हो।

- एक रेफरेंशियल इंटीग्रिटी (Referential Integrity) कन्स्ट्रेंट (constraint) दो तालिकाओं के बीच निर्दिष्ट (specified) है।
- रेफरेंशियल इंटीग्रिटी (Referential Integrity) की कमी में, यदि तालिका 1 में कोई फॉरेन कुंजी तालिका 2 की प्राथमिक कुंजी को संदर्भित करती है। तो तालिका 1 में फॉरेन कुंजी का प्रत्येक मान शून्य होना चाहिए या तालिका 2 में उपलब्ध होना चाहिए।

(Table 1)

EMP_NAME	NAME	AGE	D_No
1	Jack	20	11
2	Harry	40	24
3	John	27	18
4	Devil	38	13

Foreign key

Not allowed as D\_No 18 is not defined as a Primary key of table 2 and In table 1, D\_No is a foreign key defined

Relationships

(Table 2)

<u>D_No</u>	D_Location
11	Mumbai
24	Delhi
13	Noida

Primary Key

## रेफ़रेंशियल इंटीग्रिटी कन्सट्रेंट के लाभ

रेफ़रेंशियल इंटीग्रिटी के साथ एक संबंधपरक डेटाबेस प्रबंधन प्रणाली का उपयोग करने से कई फायदे मिलते हैं:

- डुप्लिकेट डेटा के प्रवेश को रोकता है
- एक तालिका को किसी अन्य तालिका में अस्तित्वहीन (nonexistent) फ़ील्ड को इंगित (pointing) करने से रोकता है
- " भागीदार (partnered) " तालिकाओं के बीच स्थिरता की गारंटी देता है
- किसी रिकॉर्ड को हटाने से रोकता है, जिसमें किसी अन्य कुंजी द्वारा किसी अन्य तालिका में निर्दिष्ट (referred) मान शामिल होता है
- एक तालिका के लिए रिकॉर्ड जोड़ने से रोकता है, जिसमें एक विदेशी कुंजी होती है जब तक कि लिंक की गई तालिका में एक प्राथमिक कुंजी नहीं होती है।

### 1.10 डेटाबेस एप्लीकेशन:

#### 1. बैंकिंग:

हम रोजाना बैंकों के माध्यम से हजारों लेन-देन करते हैं और हम बैंक में गए बिना ऐसा कर सकते हैं। तो कैसे बैंकिंग इतनी आसान हो गई है कि घर बैठे हम बैंकों के माध्यम से पैसा भेज या प्राप्त कर सकते हैं। यह सब डीबीएमएस के कारण ही संभव है जो सभी बैंक के लेनदेन का प्रबंधन करता है। डेटाबेस प्रबंधन प्रणाली विशेष रूप से (बड़ी मात्रा में) डेटा के भंडारण और पुनर्प्राप्ति के लिए डिज़ाइन की गई है। बैंक बड़ी मात्रा में डेटा संभालते हैं।

एक DBMS उन्हें सक्षम बनाता है:

- उस डेटा को स्टोर करें,
- डेटा पर काम करते हैं, और जरूरत पड़ने पर इसे पुनः प्राप्त करते हैं।
- ग्राहक की जानकारी संग्रहीत करने के लिए,
- क्रेडिट और डेबिट लेनदेन पर प्रतिदिन नज़र रखने,
- बैंक स्टेटमेंट आदि बनाना

यह सब काम डेटाबेस मैनेजमेंट सिस्टम की मदद से किया गया है।

## 2. रेलवे:

टिकट बुकिंग, ट्रेन के प्रस्थान और आगमन की स्थिति का रिकॉर्ड रखने के लिए डेटाबेस की आवश्यकता होती है। इसके अलावा अगर ट्रेनें लेट हो जाती हैं तो लोग डेटाबेस अपडेट के जरिए इसे जान पाते हैं। रेलवे रिजर्वेशन सिस्टम के लिए डेटाबेस बनाए रखने का मुख्य उद्देश्य बुकिंग में शामिल मैनुअल त्रुटियों को कम करना और टिकटों को रद्द करना है और ग्राहकों और प्रदाताओं को अपने ग्राहकों के बारे में डेटा बनाए रखने और उन पर उपलब्ध सीटों के बारे में भी सुविधाजनक बनाना है।

एक DBMS सक्षम करता है:

- एंड्रॉइड प्रोजेक्ट एप्लिकेशन पर एक रेलवे सिस्टम डेटाबेस प्रोजेक्ट ट्रेन टिकट खरीदने में शामिल मानव प्रयासों को कम करता है।
- यह यात्रियों को टिकट खरीदने के लिए लंबी कतारों में इंतजार करने से बचता है।
- यह एप्लिकेशन उपयोग करने के लिए विश्वसनीय है।
- आप इस एप्लिकेशन के माध्यम से आसानी से एक रेलवे सिस्टम डेटाबेस प्रोजेक्ट पर मिनी प्रोजेक्ट कर सकते हैं।
- यह एप्लिकेशन लोगों को टिकट मैनुअल रूप से ले जाने से बचता है।

## 3. स्कूल:

स्कूल अपने विद्यार्थियों के बारे में विवरण संग्रहीत करने के लिए डेटाबेस का उपयोग करते हैं, जैसे कि वे कितने दिनों से स्कूल से अवकाश पर हैं। एक अस्पताल अपने सभी रोगियों का विवरण एक डेटाबेस में संग्रहीत करेगा, जैसे कि उनके स्वास्थ्य के मुद्दों का इतिहास। सरकार लोगों के आयकर भुगतान के रिकॉर्ड को संग्रहीत करने के लिए एक डेटाबेस का उपयोग करती है।

डेटाबेस सिस्टम का उपयोग अक्सर स्कूल और कॉलेजों में छात्र के विवरण, स्टाफ विवरण, पाठ्यक्रम विवरण, परीक्षा विवरण, पेरोल डेटा, उपस्थिति विवरण, शुल्क विवरण आदि के बारे में जानकारी संग्रहीत करने और प्राप्त करने के लिए किया जाता है।

अंतर-संबंधित (inter-related) डेटा की एक बड़ी मात्रा है जिसे एक कुशल तरीके से संग्रहीत और पुनर्प्राप्त करने की आवश्यकता है। परीक्षाएं आज ऑनलाइन होती हैं और विश्वविद्यालय और कॉलेज डीबीएमएस के माध्यम से इन सभी रिकॉर्डों को बनाए रखते हैं। छात्र के पंजीकरण विवरण, परिणाम, पाठ्यक्रम और ग्रेड सभी जानकारी डेटाबेस में संग्रहीत हैं।

स्कूलों में DBMS के कुछ और फायदे:

- सरलीकृत प्रवेश प्रक्रिया
- तेज़ उपस्थिति प्रबंधन
- रिपोर्ट प्रबंधन
- स्वचालित समय सारिणी जनरेटर
- स्कूल इवेंट कैलेंडर
- अनायास असाइनमेंट प्रबंधन
- बड़ी हुई डेटा सुरक्षा

#### 4. खुदरा स्टोर (Retail store):

अपने खुदरा व्यापार की दक्षता और लाभप्रदता के लिए अपनी सूची को नियंत्रित करना आवश्यक है। इन्वेंट्री डेटाबेस आपको प्रत्येक उत्पाद के लिए स्टॉक स्तर की एक सटीक, ताज़ा तस्वीर प्रदान करता है, ताकि आपके पास ओवरस्टॉकिंग के बिना ग्राहक की मांग को पूरा करने के लिए पर्याप्त स्टॉक हो।

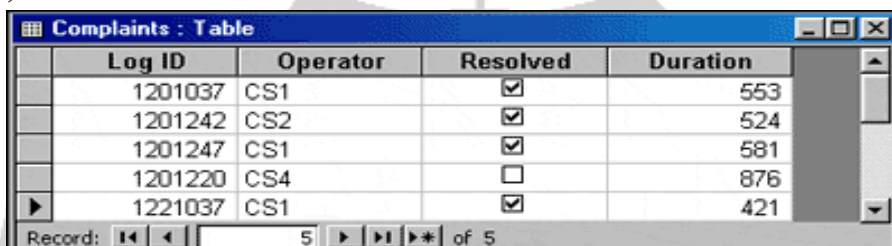
एक DBMS सक्षम करता है:

- मात्रा, बिल, खरीद, आपूर्ति श्रृंखला प्रबंधन जैसे उत्पादों के बारे में सभी विवरणों का रिकॉर्ड रखें।
- ग्राहक की जानकारी संग्रहीत करने के लिए,
- उत्पादन जानकारी और चालान विवरण संग्रहीत करने के लिए।
- व्यापार राजस्व बढ़ाता है
- यह आपके उत्पादों को प्रभावी ढंग से बाजार में लाना आसान बनाता है
- ग्राहक संतुष्टि में सुधार करता है

## 1.11 डेटाबेस एलिमेंट्स:

### 1. तालिका:

एक डेटाबेस तालिका रिकॉर्ड और फ़ील्ड से बना होता है जो डेटा को धारण करता है। तालिका को डाटाशीट भी कहा जाता है। एक डेटाबेस में प्रत्येक तालिका एक अलग, लेकिन संबंधित, विषय के बारे में डेटा रखती है।

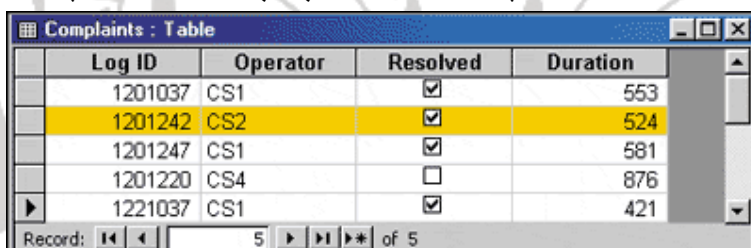


Log ID	Operator	Resolved	Duration
1201037	CS1	<input checked="" type="checkbox"/>	553
1201242	CS2	<input checked="" type="checkbox"/>	524
1201247	CS1	<input checked="" type="checkbox"/>	581
1201220	CS4	<input type="checkbox"/>	876
1221037	CS1	<input checked="" type="checkbox"/>	421

डेटाबेस तालिका

#### ए) रिकॉर्ड्स

डेटा रिकॉर्ड में संग्रहीत किया जाता है। एक रिकॉर्ड फ़ील्ड से बना होता है और इसमें एक डेटाबेस में किसी विशेष व्यक्ति, कंपनी या आइटम के बारे में सभी डेटा होते हैं। इस डेटाबेस में, एक रिकॉर्ड में एक ग्राहक से संबंधित रिपोर्ट के लिए डेटा होता है। डेटाबेस तालिका में पंक्तियों के रूप में रिकॉर्ड दिखाई देते हैं। नीचे दिए गए चित्र में लॉग आईडी 1201242 का रिकॉर्ड डाला गया है।



Log ID	Operator	Resolved	Duration
1201037	CS1	<input checked="" type="checkbox"/>	553
1201242	CS2	<input checked="" type="checkbox"/>	524
1201247	CS1	<input checked="" type="checkbox"/>	581
1201220	CS4	<input type="checkbox"/>	876
1221037	CS1	<input checked="" type="checkbox"/>	421

#### ख) फ़ील्ड्स

एक फ़ील्ड एक रिकॉर्ड का हिस्सा है और रिकॉर्ड के विषय के लिए डेटा का एक भाग है। चित्र में चित्रित डेटाबेस तालिका में, प्रत्येक रिकॉर्ड में चार फ़ील्ड शामिल हैं:

Log ID	पहचान में सहायता के उद्देश्यों के लिए ग्राहक को सौंपा गया एक नंबर
Operator	ग्राहक सहायता ऑपरेटर का कोड जिसने इस कार्य को संभाला
Resolved	घटना का समाधान किया गया था क्या, जानने के लिए एक चेक बॉक्स
Duration	ऑपरेटर द्वारा इस घटना पर बिताए गए समय, सेकंड में

डेटाबेस तालिका में फ़ील्ड स्तंभ के रूप में दिखाई देते हैं। पांच रिकॉर्ड के लिए लॉग आईडी फ़ील्ड से डेटा नीचे चित्र में हाइलाइट किया गया है।

Log ID	Operator	Resolved	Duration
1201037	CS1	<input checked="" type="checkbox"/>	553
1201242	CS2	<input checked="" type="checkbox"/>	524
1201247	CS1	<input checked="" type="checkbox"/>	581
1201220	CS4	<input type="checkbox"/>	876
1221037	CS1	<input checked="" type="checkbox"/>	421

डेटाबेस तालिका में फ़ील्ड स्तंभ के रूप में दिखाई देते हैं।

## 2. फॉर्म

एक डेटाबेस फॉर्म एक रिकॉर्ड के लिए सभी या चयनित फ़ील्ड दिखाता है। एक आकर्षक और आसानी से पढ़े जाने वाले प्रारूप में फ़ील्ड नाम और डेटा दिखाते हैं। डेटाबेस का रिकॉर्ड 1 नीचे चित्र में दिखाया गया है।

Log ID	1201037
Operator	CS1
Resolved	<input checked="" type="checkbox"/>
Duration	553

## डेटाबेस फॉर्म

आप फ़ील्ड में डेटा को केवल एक तालिका में दर्ज कर सकते हैं। नीचे दिए गए चित्र में दिखाए गए फॉर्म में डेटा को तीन क्षेत्रों में दर्ज किया गया है। एक रिकॉर्ड से दूसरे रिकॉर्ड में जाने के लिए नेविगेशन बटन फॉर्म के निचले भाग में प्रदर्शित किए जाते हैं।

Log ID	1231420
Operator	CS6
Resolved	<input checked="" type="checkbox"/>
Duration	

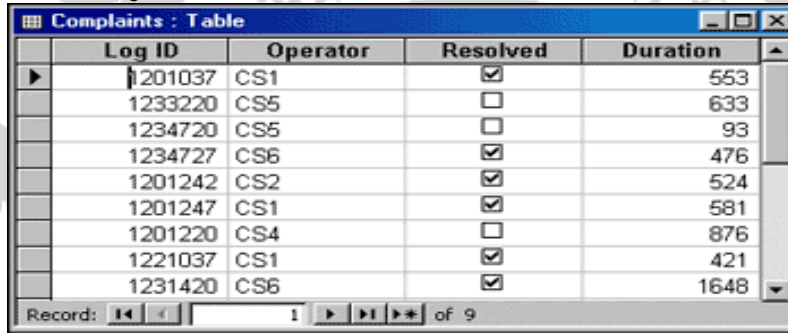
Navigation Buttons

एक रिकॉर्ड से दूसरे रिकॉर्ड में जाने लिए नेविगेशन बटन का उपयोग करें।

### 3. क्वेरी

एक क्वेरी डेटाबेस में आपके द्वारा निर्दिष्ट मानदंडों के अनुसार रिकॉर्ड जमाता है। कभी-कभी बड़े डेटाबेस में सभी डेटा के साथ काम करना व्यावहारिक नहीं होता है। आप केवल एक छोटे से हिस्से या डेटा के सबसेट के साथ काम करना चाह सकते हैं। उदाहरण के लिए, आप ग्राहक के अनसुलझे कार्यों की सहायता के लिए रिकॉर्ड ढूंढना चाह सकते हैं। इन रिकॉर्ड को प्रदर्शित करने के लिए एक क्वेरी का उपयोग किया जा सकता है। आप केवल चयनित फ़ील्ड, जैसे लॉग आईडी और रेसोल्वेड (Resolved) प्रदर्शित करने के लिए क्वेरी का निर्माण कर सकते हैं। एक क्वेरी केवल आपके द्वारा अनुरोधित डेटा को प्रदर्शित करती है।

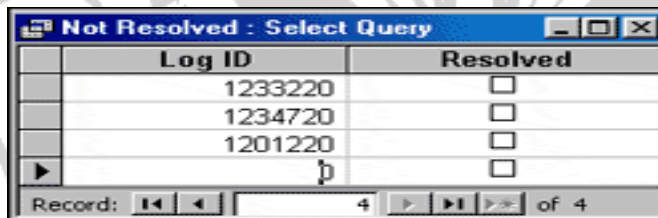
नीचे दी गई तालिका में तालिका डेटाबेस में दर्ज किए गए सभी डेटा को दिखाती है। रेसोल्वेड (Resolved) किए गए फ़ील्ड में एक खाली चेकबॉक्स इंगित करता है कि घटना रेसोल्वेड (Resolved) नहीं हुई है।



Log ID	Operator	Resolved	Duration
201037	CS1	<input checked="" type="checkbox"/>	553
1233220	CS5	<input type="checkbox"/>	633
1234720	CS5	<input type="checkbox"/>	93
1234727	CS6	<input checked="" type="checkbox"/>	476
1201242	CS2	<input checked="" type="checkbox"/>	524
1201247	CS1	<input checked="" type="checkbox"/>	581
1201220	CS4	<input type="checkbox"/>	876
1221037	CS1	<input checked="" type="checkbox"/>	421
1231420	CS6	<input checked="" type="checkbox"/>	1648

तालिका का सभी डेटा प्रदर्शित करना

नीचे दिया गया चित्र एक प्रश्न के परिणाम दिखाता है। अनसुलझे घटनाओं के लिए केवल लॉग आईडी और रेसोल्वेड (Resolved) किए गए फ़ील्ड प्रदर्शित किए जाते हैं।



Log ID	Resolved
1233220	<input type="checkbox"/>
1234720	<input type="checkbox"/>
1201220	<input type="checkbox"/>
	<input type="checkbox"/>

क्वेरी से प्राप्त डेटा

#### 4. रिपोर्ट

एक रिपोर्ट आकर्षक प्रारूप में डेटा प्रस्तुत करती है और विशेष रूप से मुद्रण के लिए उपयुक्त है। रिपोर्ट तालिका या प्रश्नों से डेटा प्रदर्शित कर सकती हैं। सभी या चयनित क्षेत्रों को एक रिपोर्ट में शामिल किया जा सकता है। डेटा को समूहीकृत या सॉर्ट किया जा सकता है और विभिन्न तरीकों से व्यवस्थित किया जा सकता है।

नीचे दी गई तस्वीर एक रिपोर्ट दिखाती है जिसमें शिकायत तालिका से डेटा है। डेटा को सुलझे और अनसुलझे घटनाओं द्वारा समूहीकृत (grouped) किया जाता है। लॉग आईडी फ़ील्ड को आरोही (ascending) क्रम में सॉर्ट किया जाता है।

<b>Complaints</b>			
<b>Resolved</b>	<b>Log ID</b>	<b>Operator</b>	<b>Duration</b>
<input type="checkbox"/> Yes	1201037	CS1	553
	1201242	CS2	524
	1201247	CS1	581
	1221037	CS1	421
	1231420	CS6	1648
	1234727	CS6	476
<input type="checkbox"/> No	1201220	CS4	876
	1233220	CS5	633
	1234720	CS5	93

डेटाबेस रिपोर्ट

इस रिपोर्ट में डेटा के प्रारूप की तुलना नीचे चित्र में दिखाई गई डेटाबेस तालिका से करें।

Log ID	Operator	Resolved	Duration
1201037	CS1	<input checked="" type="checkbox"/>	553
1233220	CS5	<input type="checkbox"/>	633
1234720	CS5	<input type="checkbox"/>	93
1234727	CS6	<input checked="" type="checkbox"/>	476
1201242	CS2	<input checked="" type="checkbox"/>	524
1201247	CS1	<input checked="" type="checkbox"/>	581
1201220	CS4	<input type="checkbox"/>	876
1221037	CS1	<input checked="" type="checkbox"/>	421
1231420	CS6	<input checked="" type="checkbox"/>	1648

डेटाबेस तालिका

## 1.12 क्लाइंट सर्वर मॉडल (Paradigm) का परिचय:

क्लाइंट-सर्वर DBMS (डेटाबेस मैनेजमेंट सिस्टम) एक आर्किटेक्चर मॉडल को संदर्भित करता है जो क्लाइंट मशीनों और सर्वर के बीच डेटाबेस कार्यक्षमता को अलग करता है।

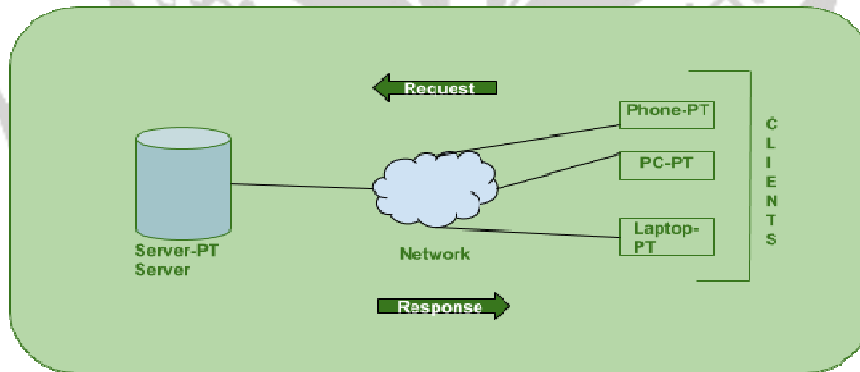
क्लाइंट-सर्वर मॉडल एक वितरित एप्लिकेशन संरचना है जो किसी सर्वर या सर्वर नामक सेवा (service) के प्रदाताओं और सेवा (service) अनुरोधकर्ताओं या क्लाइंट के बीच के कार्य एवं कार्यभार का विभाजन करती है। क्लाइंट-सर्वर आर्किटेक्चर में, जब क्लाइंट कंप्यूटर इंटरनेट के माध्यम से सर्वर पर डेटा के लिए अनुरोध भेजता है, तो सर्वर अनुरोधित प्रक्रिया को स्वीकार करता है और ग्राहक को वापस अनुरोध किए गए डेटा पैकेट वितरित करता है। ग्राहक अपने किसी भी संसाधन को साझा नहीं करते हैं। क्लाइंट-सर्वर मॉडल के उदाहरण ईमेल, वर्ल्ड वाइड वेब आदि हैं।

### 1.12.1 क्लाइंट-सर्वर मॉडल कैसे काम करता है?

यह WEB की एक ठोस नींव रखने में हमारी मदद करेगा और WEB प्रौद्योगिकियों के साथ आसानी से काम करने में मदद करेगा।

- क्लाइंट: जब हम क्लाइंट शब्द की बात करते हैं, तो इसका मतलब किसी व्यक्ति या किसी विशेष सेवा का उपयोग करने वाले संगठन से बात करना है। इसी तरह डिजिटल दुनिया में एक क्लाइंट एक कंप्यूटर (होस्ट) है, जो सेवा प्रदाताओं (सर्वर) से किसी विशेष सेवा की जानकारी प्राप्त करने या उपयोग करने में सक्षम है।
- सर्वर: इसी तरह, जब हम सर्वर शब्द की बात करते हैं, तो इसका अर्थ है एक व्यक्ति या माध्यम जो कुछ कार्य करता है। इसी तरह इस डिजिटल दुनिया में एक सर्वर एक दूरस्थ कंप्यूटर है जो विशेष सेवाओं तक जानकारी (डेटा) या पहुंच प्रदान करता है।

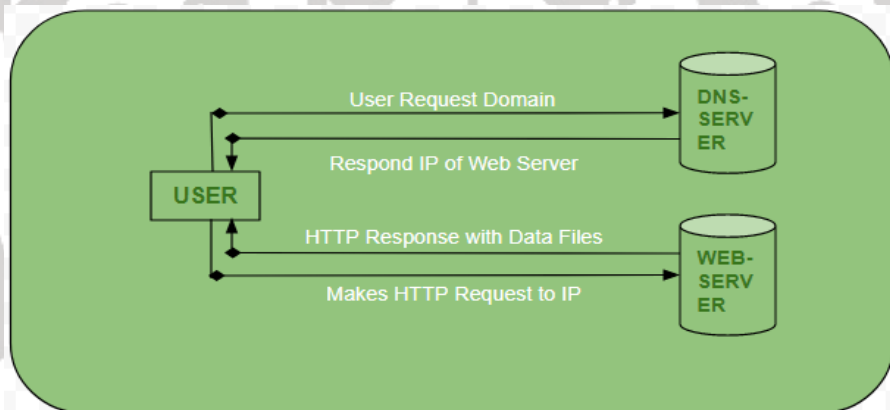
तो, इसका मूल रूप से क्लाइंट कुछ अनुरोध कर रहा है और सर्वर इसे डेटाबेस में अपने वर्तमान के रूप में लंबे समय से सेवा कर रहा है।



ब्राउज़र सर्वर के साथ कैसे इंटरैक्ट करता है?

सर्वर के क्लाइंट के साथ इंटरैक्ट करने के लिए कुछ पद हैं।

- उपयोगकर्ता वेबसाइट या फ़ाइल का URL (यूनिफ़ॉर्म रिसोर्स लोकेटर) बताता है। ब्राउज़र तब DNS (DOMAIN NAME SYSTEM) सर्वर को अनुरोध करता है।
- WEB सर्वर के पते के लिए DNS सर्वर खोजता है।
- DNS सर्वर WEB सर्वर के आईपी पते के साथ प्रतिक्रिया करता है।
- ब्राउज़र वेब सर्वर के आईपी (DNS सर्वर द्वारा प्रदान) के लिए एक  $TTP$  /  $TTPS$  अनुरोध भेजता है।
- सर्वर वेबसाइट की आवश्यक फाइलों को भेजता है।
- ब्राउज़र तब फाइलों को प्रस्तुत करता है और वेबसाइट प्रदर्शित की जाती है। यह रेंडरिंग DOM (डॉक्यूमेंट ऑब्जेक्ट मॉडल) इंटरप्रेटर, CSS इंटरप्रेटर और JS इंजन की मदद से किया जाता है जिसे सामूहिक रूप से JIT या (जस्ट इन टाइम) कंपाइलर के रूप में जाना जाता है।



**क्लाइंट-सर्वर मॉडल के लाभ:**

- एक ही स्थान पर सभी डेटा के साथ केंद्रीकृत प्रणाली।
- कम लागत तथा कम रखरखाव की आवश्यकता होती है और डेटा रिकवरी संभव है।
- क्लाइंट और सर्वर की क्षमता को अलग-अलग बदला जा सकता है।

**क्लाइंट-सर्वर मॉडल के नुकसान:**

- सर्वर में मौजूद या सर्वर में अपलोड होने पर ग्राहक वायरस, ट्रोजन और वर्म से ग्रस्त होते हैं।
- सर्वर डिनायल ऑफ़ सर्विस (DOS) का शिकार होते हैं।
- डेटा पैकेट ट्रांसमिशन के दौरान खराब हो सकते हैं या बदल सकते हैं।
- लॉगिन क्रेडेंशियल की फ़िशिंग या कैप्चरिंग या उपयोगकर्ता की अन्य उपयोगी जानकारी आम हैं और MITM (मैन इन द मिडल) हमले आम हैं।