

डेटाबेस यूजिंग एमएस एक्सेस

यूनिट - 3

एमएस एक्सेस में क्वेरी का परिचय

(Introduction of Query in MS ACCESS)

Query एक या अधिक Tables से डेटा खोजने और संकलित करने का एक तरीका है। एक क्वेरी चलाना आपके डेटाबेस का एक विस्तृत प्रश्न पूछना है। जब आप एक्सेस में कोई क्वेरी बनाते हैं, तो इसका मतलब है कि आप सटीक डेटा को ढूँढने के लिए विशिष्ट खोज स्थितियों को परिभाषित कर रहे हैं।

किसी भी डाटाबेस फाइल में उसकी संरचना के अनुकूल डाटा डाला जाता है। किसी भी व्यावसायिक कार्य में प्रयोग होने वाले डाटाबेस फाइल में बहुत बड़ा डाटा संग्रहित किया जाता है। डाटाबेस फाइल उन डाटा को क्रमवार तरीके से संग्रहित करती जाती है। डाटाबेस फाइल का मुख्य उद्देश्य वांछित डाटा को कम से कम समय में यूजर को दर्शाना है।

जो डाटा हम टेबल या डाटाबेस फाइल में संग्रहित करते हैं, उसकी आवश्यकता हमें दूसरे दिन, एक माह बाद या कुछ वर्षों बाद भी हो सकती है। सिर्फ टेबल के प्रयोग से यह कार्य मुश्किल हो सकता है लेकिन इच्छित डाटा प्राप्त करने के लिए क्वेरी (Query) बहुत अच्छा विकल्प है। जहाँ पर एक से अधिक टेबल का उपयोग हो रहा है, वहाँ क्वेरी का महत्व और अधिक बढ़ जाता है। क्वेरी की सहायता से प्रयोगकर्ता इच्छित डाटा को तेजी से प्राप्त कर सकता है। Access में क्वेरी से प्राप्त डाटा को Form, Report आदि में भी प्रयोग कर सकते हैं। दूसरे शब्दों में “क्वेरी डाटा को जानकारी में परिवर्तित करती है।”

क्वेरी का उपयोग

क्वेरीज़ सरल खोजों या फ़िल्टरों की तुलना में कहीं अधिक शक्तिशाली हैं जिनका उपयोग आप टेबल में डेटा खोजने के लिए कर सकते हैं। ऐसा इसलिए है क्योंकि क्वेरी कई जानकारी से अपनी जानकारी खींच सकती है। उदाहरण के लिए, आप अपने व्यापार में एक ग्राहक का नाम ढूँढने के लिए ग्राहक टेबल में एक खोज का उपयोग कर सकते हैं या ऑर्डर टेबल पर एक फ़िल्टर को पिछले सप्ताह के भीतर केवल ऑर्डर देखने के लिए उपयोग कर सकते हैं, हालांकि, आप पिछले हफ्ते में खरीदारी करने वाले प्रत्येक ग्राहक के नाम और फोन नंबर को खोजने के लिए आसानी से एक क्वेरी चला सकते हैं। एक अच्छी तरह से डिज़ाइन की गई क्वेरी ऐसी जानकारी दे सकती है जो आप अपने टेबल में डेटा की जांच करके नहीं ढूँढ पाएंगे।

जब आप कोई क्वेरी चलाते हैं, तो परिणाम आपको टेबल में प्रस्तुत किए जाते हैं, लेकिन जब आप एक डिजाइन करते हैं तो आप एक अलग व्यू का उपयोग करते हैं। इसे क्वेरी डिजाइन व्यू कहा जाता है, और यह आपको यह देखने देता है कि आपकी क्वेरी एक साथ कैसे रखी जाती है।

एमएस एक्सेस में रिलेशनशिप (Relationships MS Access)

डेटाबेस के संदर्भ में Relationship, एक ऐसी स्थिति है जो दो रिलेशनल डेटाबेस टेबल के बीच मौजूद होती है जब एक टेबल में एक Foreign Key होती है जो अन्य टेबल की Primary key का संदर्भ देती है। Relationship अलग-अलग डेटा आइटम्स को जोड़ने के दौरान, अलग-अलग टेबल्स में डेटा को विभाजित और स्टोर करने के लिए Relationship डेटाबेस को अनुमति देते हैं Relationships में दो टेबिल को आपस में जोड़ा जाता है। इसके लिये दोनों टेबिल में एक कामन फील्ड (Common field) होना चाहिये। जिनमें दोनों का नाम एवं डाटा टाईप एक समान होना जरूरी है। टेबिल में Relation बनाने के बाद यूजर फार्म एवं रिपोर्ट में multiple टेबल का प्रयोग करके रिपोर्ट एवं फार्म को डिजाइन कर सकता है। प्रायः Master टेबल और Transaction टेबल के बीच रिलेशन बनायीं जाती है, जिस टेबल में प्राइमरी की (Primary Key) होती है उस टेबल को मास्टर टेबल कहते हैं, Transaction टेबल में जो फील्ड कॉमन होती है उसे हम Foreign Key कहते हैं।

Types of Relationship

MS Access में रिलेशनशिप तीन प्रकार की होती हैं -

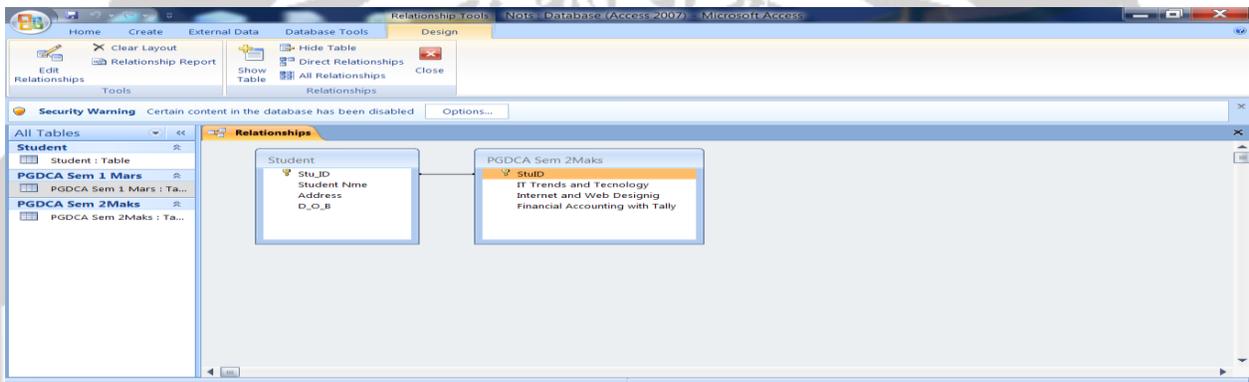
- One to One Relationship
- One to Many Relationships
- Many to Many Relationships

1. One to One Relationship

प्रत्येक टेबिल को एक entity के रूप में निरूपित करते हैं इसलिए इसे entity set कहते हैं। entity को square box में प्रदर्शित करते हैं। जब दो entities के मध्य रिलेशनशिप स्थापित करते हैं। तब इसे entity relationship कहते हैं। one to one relationship केवल दो entities के मध्य का एसोसिएशन है। जैसे एक महाविद्यालय में प्रत्येक विभाग में एक HOD

होता है। तथा एक HOD एक से अधिक विभाग का HOD नहीं हो सकता है। ऐसी रिलेशनशिप one to one relationship कहलाती है।

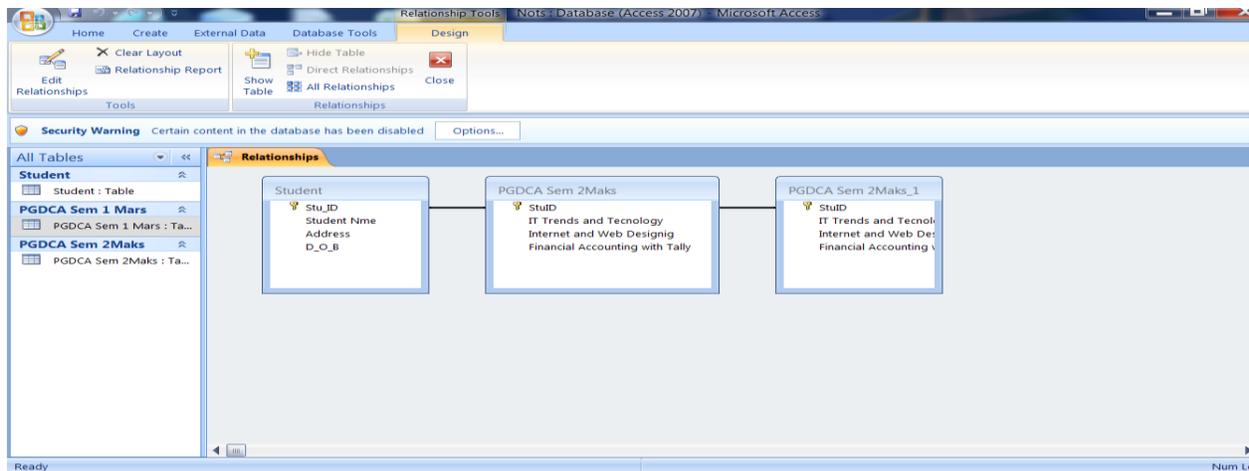
one to one relationship में, पहली टेबल में प्रत्येक रिकॉर्ड में दूसरी टेबल में केवल एक मिलान रिकॉर्ड हो सकता है, और दूसरी टेबल में प्रत्येक रिकॉर्ड में पहली टेबल में केवल एक मिलान रिकॉर्ड हो सकता है। यह संबंध आम नहीं है क्योंकि, अक्सर इस तरह से संबंधित जानकारी उसी टेबल में संग्रहीत होती है। सुरक्षा कारणों से किसी टेबल के हिस्से को अलग करने के लिए, या मुख्य टेबल के उप-समूह पर लागू होने वाली जानकारी संग्रहीत करने के लिए आप कई फ़ील्ड वाले टेबल को विभाजित करने के लिए one to one relationship का उपयोग कर सकते हैं।



2. One to Many Relationships

one to many relationship में दो entities के मध्य एक से अधिक relationship हो सकती है जैसे एक कोर्स में बहुत से छात्र पंजीकृत हो सकते हैं। one to many relationship को अक्सर “Master details” या “Parent child” relationship के रूप में जाना जाता है।

One to many relationship relationships का सबसे आम प्रकार है। एक से कई relationship में टेबल में कई मिलान रिकॉर्ड हो सकते हैं जैसे टेबल ए में एक रिकॉर्ड टेबल बी में कई मिलान रिकॉर्ड हो सकते हैं।



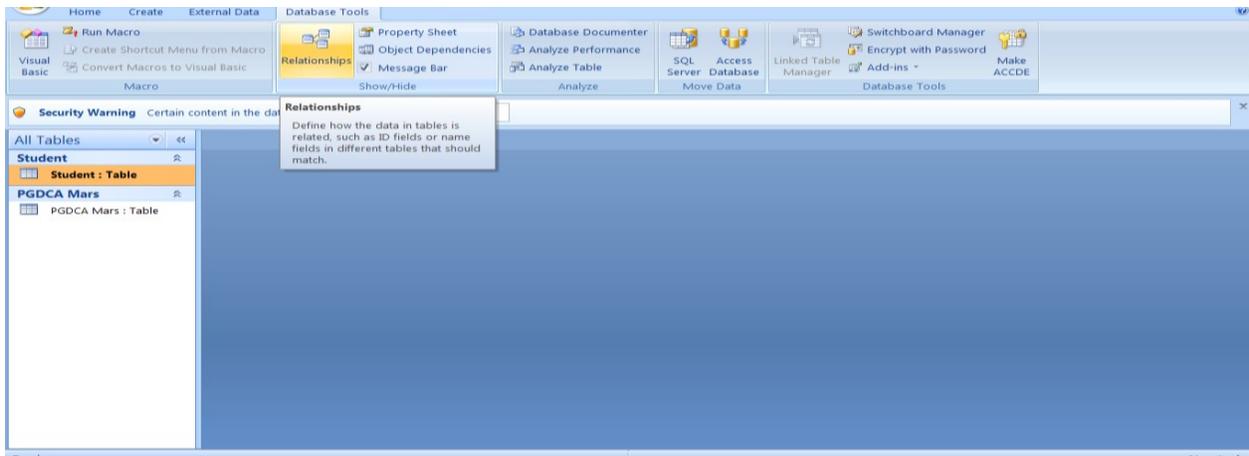
3. Many to Many Relationships

Many to many relationship थोड़ा जटिल हैं। आपको relationship के दोनों पक्षों पर विचार करना चाहिए क्योंकि दोनों टेबल्स में कई रिकॉर्ड होंगे। इस relationship को काम करने के लिए, आपको एक तीसरी टेबल की आवश्यकता होगी, जिसे जंक्शन टेबल कहा जाता है। यह टेबल विभिन्न relationship को स्पष्ट करने के लिए ज़िम्मेदार हो सकती है। Many to many relationship में दो entities के मध्य एक से अधिक relationship हो सकती है। जैसे एक आइटमको बहुत से लोग खरीद सकते है। या एक विद्यार्थी एक विद्यालय में अनेक कोर्स ले सकता है।

एमएस एक्सेस में रिलेशनशिप Creation की प्रक्रिया

एक रिश्ते आपको दो अलग-अलग Tabled से डेटा को गठबंधन करने में मदद करता है।

- 1) सबसे पहले Database Tools Tab में स्थिर Relationship group में से Relationship पर क्लिक करें।



2) आपको एक Show Table डायलॉग बॉक्स दिखाई देगा Show Table डायलॉग बॉक्स डेटाबेस में सभी टेबल और क्वेरी प्रदर्शित करता है। केवल टेबल देखने के लिए, टेबल्स पर क्लिक करें।

3) एक या अधिक टेबल का चयन करें, और फिर Add पर क्लिक करें। टेबल जोड़ने के बाद, Close पर क्लिक करें।

उदाहरण के लिये यदि हमने दो Tables पहला "Student" तथा दूसरा " PGDCA Marks" बनाया है तो दोनों को Add करेंगे ।

4) अब आप दो Box देखेंगे जो आपकी दोनों Tables का प्रतिनिधित्व करते हैं। Student table से Studentid को "PGDCA Marks Table में स्थित Studentid फ़ील्ड पर क्लिक करके खींचें।

5) Edit Relationship डायलॉग बॉक्स दिखाई देगा। Enforce Referential Integrity वाले चेक बॉक्स पर क्लिक करें और Ok पर क्लिक करें।

6) अब आप Studenttable से Studentid field और "PGDCA Marks" Table के Studentid के बीच स्थापित एक लाइन देखेंगे।

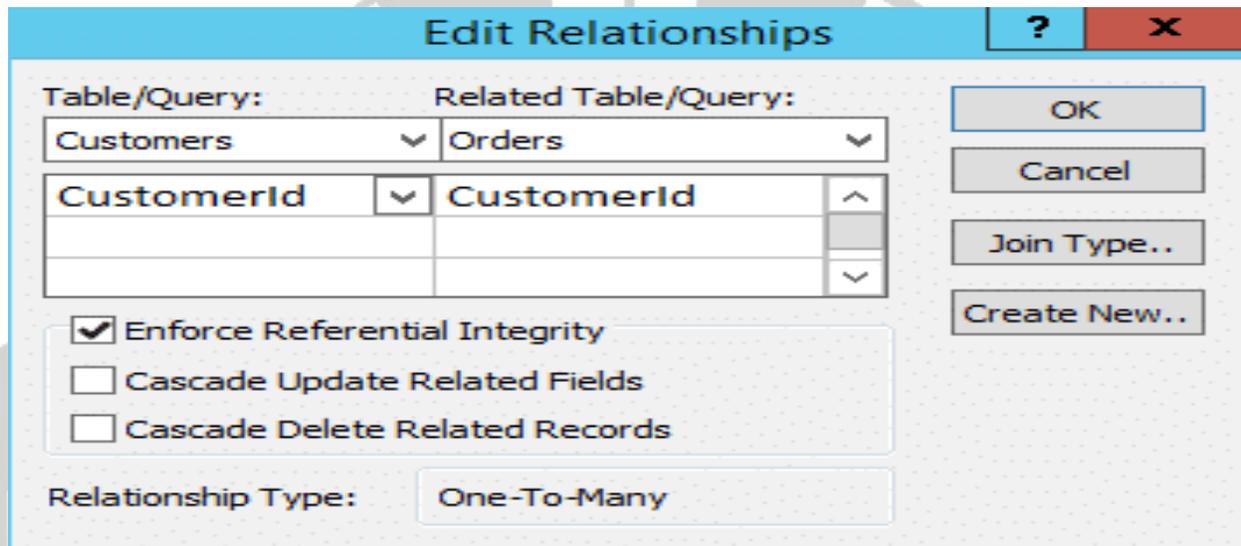
7) अंत में Create पर क्लिक करें।

8) इसे save करने के लिए save बटन पर क्लिक करें।

एमएस एक्सेस में रिलेशनशिप में कैसे सुधार करें

(How to Edit a Relationship in MS Access)

- 1) यदि आप MS Access में अपने रिलेशनशिप डिज़ाइन में कोई बदलाव करना चाहते हैं तो आपको सबसे पहले उस रिलेशनशिप टेबल को खोलना होगा उसके बाद ही आप उसमें बदलाव कर सकते हैं।
- 2) डेटाबेस में टेबल के बीच सभी Relationships को देखने के लिए Database tools tab पर, Relationships पर क्लिक करें।



- 3) All Relationships पर क्लिक करें।
- 4) Relationships के साथ सभी टेबल्स को प्रदर्शित किया जाता है और साथ ही रिलेशनशिप के बीच लाइन दिखाई देती हैं।
- 5) कर्सर को उस लाइन पर रखें जिस रिलेशनशिप में आप सुधार करना चाहते हैं और उसके बाद इसे चुनने के लिए लाइन पर क्लिक करें।
- 6) आप जैसे ही लाइन को सिलेक्ट करेंगे आपको Relationships लाइन मोटी दिखाई देने लगेगी।
- 7) रिलेशनशिप लाइन चयनित के साथ, इसे डबल-क्लिक करें। Edit Relationships डायलॉग बॉक्स ओपन हो जायेगा। आप जो भी बदलाव करना चाहते हैं वह करें और उसके बाद Ok पर क्लिक करें।
- 8) Edit Relationships डायलॉग बॉक्स आपको टेबल Relationships बदलने की अनुमति देता है। विशेष रूप से, आप Relationships के दोनों तरफ या किसी भी तरफ के fields पर tables या query को बदल सकते हैं। आप Join type को भी सेट कर सकते हैं, या रेफरेंशियल अखंडता को लागू कर सकते हैं और एक कैस्केड विकल्प चुन सकते हैं।

9) जब आप रिलेशनशिप विंडो में अपना काम समाप्त कर लेते हैं तो अपने सभी रिलेशनशिप लेआउट परिवर्तनों को सेव करने के लिए Save पर क्लिक करें।

एमएस एक्सेस में रिलेशनशिप को कैसे हटाएं

(How to delete a relationship in MS Access)

MS Access में कई बार हो सकता है जब आप टेबल्स के बीच रिलेशनशिप हटाना चाहते हैं। ऐसा करने के लिए आपको पहले वह टेबल ओपन करनी पड़ेगी जिनके बीच रिलेशनशिप हैं और जिन्हें आप हटाना चाहते हैं।

1. सबसे पहले Database Tools Tab पर स्थित Relationship group में से Relationship पर क्लिक करें।

2. इसके बाद Design Tab पर स्थित Relationship group में से All Relationship पर क्लिक करें।

3. Relationship के साथ सभी टेबल्स को प्रदर्शित किया जाता है।

4. Relationship के लिए उस रिलेशनशिप लाइन पर क्लिक करें जिसे आप हटाना चाहते हैं। आप जैसे ही लाइन को सिलेक्ट करेंगे आपको लाइन मोटी दिखाई देने लगेगी।

5. इसके बाद Delete Key दबाएं।

6. एक्सेस आपको एक संदेश प्रदर्शित करेगा Are you sure you want to permanently delete the selected relationship from your database? (क्या आप वाकई अपने डेटाबेस से चुने गए रिश्ते को स्थायी रूप से हटाना चाहते हैं?) यदि यह पुष्टिकरण संदेश प्रकट होता है, तो Yes पर क्लिक करें।

7. जब आप रिलेशनशिप विंडो का उपयोग कर समाप्त कर लें, तो अपने रिलेशनशिप लेआउट परिवर्तनों को Save करने के लिए Save पर क्लिक करें।

एमएस एक्सेस में जॉइन टाइप

(Join Type in MS Access)

जब आप किसी टेबल रिलेशनशिप को परिभाषित करते हैं, तो रिलेशनशिप्स के बारे में तथ्य आपके क्वेरी डिज़ाइन को सूचित करते हैं। उदाहरण के लिए, यदि आप दो टेबल्स के बीच

रिलेशनशिप परिभाषित करते हैं, और फिर आप एक क्वेरी बनाते हैं जो दो टेबल्स को नियोजित करता है, तो रिलेशनशिप में निर्दिष्ट फ़ील्ड के आधार पर स्वचालित रूप से डिफ़ॉल्ट मिलान फ़ील्ड का उपयोग करें। आप अपनी क्वेरी में इन शुरुआती डिफ़ॉल्ट मानों को ओवरराइड कर सकते हैं, लेकिन रिलेशनशिप द्वारा प्रदान किए गए मूल्य अक्सर सही होंगे। चूंकि एक से अधिक टेबल से डेटा एकत्र करना और डेटा लाने से कुछ ऐसा होता है जो आप सबसे सरल डेटाबेस में करते हैं, रिलेशनशिप को बनाकर डिफ़ॉल्ट सेट करना समय बचाने और फायदेमंद हो सकता है। एक से अधिक टेबल क्वेरी सामान्य फ़ील्ड में मानों को मेल करके एक से अधिक टेबल से जानकारी को जोड़ती है। मिलान और संयोजन करने वाले ऑपरेशन को Join कहा जाता है। उदाहरण के लिए, मान लीजिए कि आप Customer Order प्रदर्शित करना चाहते हैं। आप एक क्वेरी बनाते हैं जो ग्राहक टेबल फ़ील्ड पर ग्राहक टेबल और ऑर्डर टेबल में शामिल हो जाती है। क्वेरी परिणाम में केवल उन पंक्तियों के लिए ग्राहक जानकारी और ऑर्डर जानकारी होती है जहां एक संबंधित मिलान पाया गया था।

प्रत्येक रिलेशनशिप के लिए आप निर्दिष्ट वैल्यू में से एक join type हैं। जॉइन टाइप एक्सेस को बताता है कि एक क्वेरी परिणाम में कौन से रिकॉर्ड शामिल हैं। उदाहरण के लिए, फिर से एक क्वेरी पर विचार करें जो Customer table का प्रतिनिधित्व करने वाले सामान्य फ़ील्ड पर ग्राहक टेबल और ऑर्डर टेबल में शामिल हो। डिफ़ॉल्ट Join type का उपयोग करके, क्वेरी केवल ग्राहक पंक्तियों और ऑर्डर पंक्तियों को लौटाती है जहां आम फ़ील्ड्स बराबर होते हैं।

हालांकि, मान लीजिए कि आप सभी ग्राहकों को शामिल करना चाहते हैं - यहां तक कि जिन्होंने अभी तक कोई ऑर्डर नहीं दिया है। इसे पूरा करने के लिए, आपको शामिल होने वाले प्रकार को एक आंतरिक जुड़ने से बदलना होगा जिसे left outer join के रूप में जाना जाता है। एक left outer join से रिलेशनशिप के बाईं ओर टेबल में सभी पंक्तियां लौटाती हैं और केवल वे लोग जो दाईं ओर टेबल में मेल खाते हैं। एक सही बाहरी जुड़ाव दाईं ओर की सभी पंक्तियों को लौटाता है और केवल बाईं ओर से मेल खाता है।

एमएस एक्सेस में जॉइन टाइप कैसे सेट करें

(How to Set the Join Type in MS Access)

1) Edit Relationships dialog box में, Join Type पर क्लिक करें।

2) Join Properties डायलॉग बॉक्स प्रकट होता है।

3) आप जिसे सिलेक्ट करना चाहते हैं उस पर क्लिक करें, और उसके बाद Ok पर क्लिक करें।

निम्न टेबल (ग्राहक और ऑर्डर टेबल का उपयोग करके) तीन विकल्प दिखाते हैं जो Join Properties डायलॉग बॉक्स में प्रदर्शित होते हैं। जब आप विकल्प 2 या विकल्प 3 चुनते हैं, तो Relationship line पर एक तीर दिखाया जाता है। यह तीर Relationship के पक्ष में

इंगित करता है जो केवल मिलान पंक्तियों को दिखाता है।

1. Only include rows where the joined fields from both tables are equal.

(केवल पंक्तियां शामिल करें जहां दोनों तालिकाओं से जुड़े फ़ील्ड बराबर हैं।)

Inner join Matching rows Matching rows

2. Include ALL records from 'Customers' and only those records from 'Orders' where the joined fields are equal.

(‘ग्राहक’ से सभी रिकॉर्ड और ‘ऑर्डर’ से केवल उन रिकॉर्ड्स शामिल करें जहां शामिल फ़ील्ड बराबर हैं): Left outer join All rows Matching rows

3. Include ALL records from 'Orders' and only those records from 'Customers' where the joined fields are equal.

(‘ऑर्डर’ से सभी रिकॉर्ड शामिल करें और केवल ‘ग्राहक’ से उन रिकॉर्ड्स को शामिल करें जहां शामिल फ़ील्ड बराबर हैं।): Right outer join Matching rows

एमएस एक्सेस में जॉइन टाइप कैसे बदलें

(How to change Join Type in MS Access)

1) सबसे पहले database Tool tab पर स्थित Relationship group में से Relationship पर क्लिक करें।

2) Relationship विंडो दिखाई देती है। यदि आपने अभी तक किसी भी Relationship को परिभाषित नहीं किया है और यह पहली बार है कि आप Relationship विंडो खोल रहे हैं, तो Show Table डायलॉग बॉक्स प्रकट होता है। यदि डायलॉग बॉक्स प्रकट होता है, तो Close पर क्लिक करें।

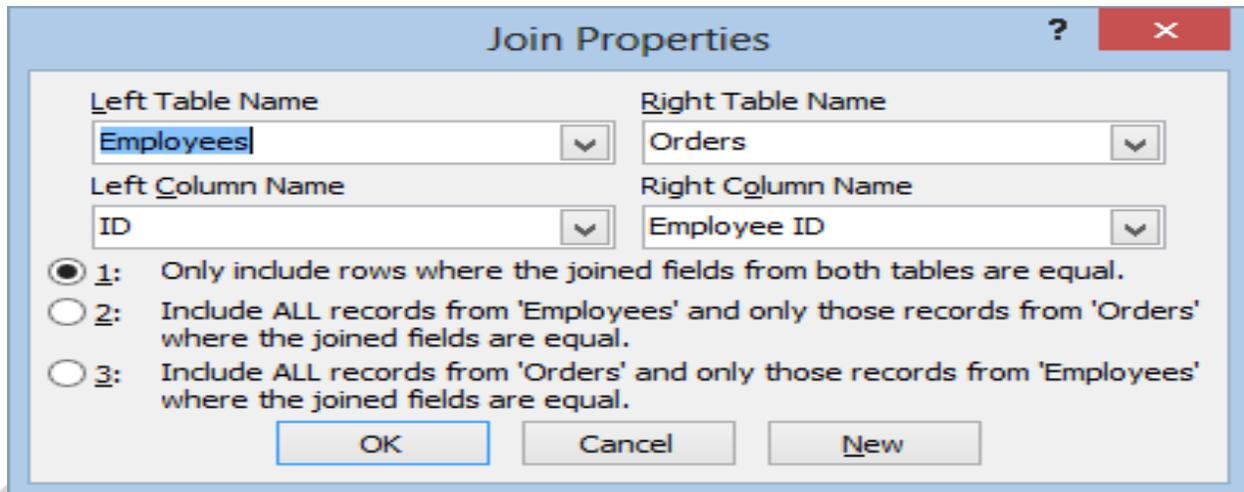
3) Design Tab पर स्थित Relationship group में से Relationship पर क्लिक करें। आपको सभी रिलेशनशिप टेबल्स दिखाई देने लगेंगी।

4) Relationship के लिए उस रिलेशनशिप लाइन पर क्लिक करें जिसे आप बदलना चाहते हैं। जब यह चुना जाता है तो Relationship Line मोटी दिखाई देती है। रिलेशनशिप लाइन को डबल-क्लिक करें।

Edit Relationship डायलॉग बॉक्स प्रकट होता है।

5) Join Type पर क्लिक करें।

6) Join Properties डायलॉग बॉक्स में, किसी एक विकल्प पर क्लिक करें, और उसके बाद Ok पर क्लिक करें।



7) Relationship में कोई अतिरिक्त बदलाव करें, और उसके बाद Ok पर क्लिक करें।

8) आप जिसे सिलेक्ट करना चाहते हैं उस पर क्लिक करें, और उसके बाद Ok पर क्लिक करें।

Referential Integrity Rule

Relationship Window में Referential Integrity का Option होता है इसका अर्थ है कि यदि दो Table के मध्य Referential Integrity rule स्थापित है तो प्रथम Table में यदि किसी Record में संशोधन या Updation करते हैं तो इससे संबंधित Table में स्वतः ही संशोधन हो जाता है इसी प्रकार यदि प्रथम table में से किसी record को delete करते हैं तो इससे संबंधित table में से वह record delete हो जाता है। Table की Relationship Create करते समय Edit Relationship Window में Referential Integrity के तीन Option होते हैं।

1) Cascade Update Related Fields-

यदि Enforce Referential Integrity के साथ Cascade Update Related Fields Check Box को सेलेक्ट करते हैं तो Table के किसी Record को Update करने पर इससे संबंधित Table में उस Record से संबंधित Information स्वयं Update हो जाएगी।

2) Cascade Delete Related Fields-

इसी प्रकार यदि Cascade Delete Related Check Box को सेलेक्ट करते हैं तो Table में किसी Record को Delete करने पर इससे जुड़ी हुयी Table में से Record स्वयं ही Delete हो जाएगा।

Enforce Referential Rule को Set करने के लिए Enforce Referential Integrity Check Box पर Click करते हैं इस के साथ ही अपनी आवश्यकतानुसार Cascade Update Related Fields एवं Cascade Delete Related Fields पर Click करके Create Button पर Click करते हैं।

Filter

फिल्टर का उपयोग टेबल में से इच्छित जानकारी को प्राप्त करने के लिए होता है। उदाहरण के लिए टेबल में छात्रों का डाटा डाला है, हमे सिर्फ "PGDCA" कोर्स के ही छात्रों का डाटा चाहिए, उस स्थिती में हम फिल्टर का उपयोग कर सकते हैं। किसी भी डाटाबेस प्रोग्राम में फिल्टर का उपयोग बहुत महत्वपूर्ण है इसकी सहायता से डाटा तेजी से प्राप्त किया जाता है।

Query

किसी भी डाटाबेस फाइल में उस संरचना के अनुकूल डाटा डाला जाता है। किसी भी व्यावसायिक कार्य में प्रयोग होने वाले डाटाबेस फाइल में बहुत बड़ा डाटा संग्रहित किया जाता है। डाटाबेस फाइल उन डाटा को क्रमवार तरीके से संग्रहित करती जाती है। डाटाबेस फाइल का मुख्य उद्देश्य वांछित डाटा को कम से कम समय में यूजर को दर्शाना है।

जो डाटा हम टेबल या डाटाबेस फाइल में संग्रहित करते हैं, उसकी आवश्यकता हमे दूसरे दिन, एक माह बाद या कुछ वर्षों बाद भी हो सकती है। सिर्फ टेबल के प्रयोग से यह कार्य मुश्किल हो सकता है लेकिन इच्छित डाटा प्राप्त करने के लिए क्वेरी (Query) बहुत अच्छा विकल्प है। जहाँ पर एक से अधिक टेबल का उपयोग हो रहा है, वहाँ क्वेरी का महत्व ओर अधिक बढ़ जाता है। क्वेरी की सहायता से प्रयोगकर्ता इच्छित डाटा को तेजी से प्राप्त कर सकता है। एक्सेस में क्वेरी से प्राप्त डाटा को From, Report आदि में भी प्रयोग कर सकते हैं। दूसरे शब्दों में "क्वेरी डाटा को जानकारी में परिवर्तित करती है।"

Query एवं Filter में अंतर

Filter-

- 1) Filter को सेव नहीं किया जा सकता है।
- 2) Filter का प्रयोग अस्थायी रूप से डाटा प्राप्त करने के लिए किया जाता है। एक बार Filter बंद करने के बाद, उस डाटा को प्राप्त करने के लिए फिर से नया Filter बनाना पड़ता है।
- 3) फॉर्म, रिपोर्ट आदि में Filter का प्रयोग नहीं किया जा सकता है।
- 4) फिल्टर में एक ही टेबल से डाटा प्राप्त किया जा सकता है।
- 5) जटिल प्रकार के डाटा या बड़े डाटाबेस के लिए Filter का उपयोग कम किया जाता है।

Query

- 1) क्वेरी को अलग फाइल में संग्रहित किया जाता है।
- 2) Queries में इच्छित डाटा स्थायी रूप में रखा जाता है। प्रयोगकर्ता कार्य के अनुसार उन्हें बार-बार प्रयोग कर सकता है।
- 3) फॉर्म, रिपोर्ट आदि में क्वेरी से प्राप्त डाटा का प्रयोग किया जा सकता है।
- 4) क्वेरी में एक से अधिक टेबल से इच्छित डाटा प्राप्त किया जा सकता है।
- 5) बड़े एवं जटिल डाटा के लिए क्वेरी अधिक उपयोगी हैं।

फिल्टर के प्रकार (Types of filter in Ms Access)

आप किसी भी डाटा टाइप के लिए एक्सेस में दिए गये फिल्टर में से किसी का भी प्रयोग कर सकते हैं। ये फिल्टर निम्न व्यू के मेनू कमांड में उपलब्ध हैं:

डाटाशीट, फॉर्म, रिपोर्ट और लेआउट।

आप Filter डॉक्यूमेंट टैब के अंदर आप्शन का प्रयोग कर के अपना खुद का फिल्टर भी बना सकते हैं। एमएस एक्सेस में निम्न प्रकार के फिल्टर होते हैं:

- 1) कॉमन फिल्टर्स- किसी खास वैल्यू या वैल्यूज के रेंज को फिल्टर करने के लिए इस फिल्टर का प्रयोग करते हैं।
- 2) फिल्टर बाई सिलेक्शन- टेबल के अंदर उन अभी रो को फिल्टर करने के जिनमे वो वैल्यूज होती हैं जो किसी खास रो के वैल्यू से मैच होती है।

3) फिल्टर बाई फॉर्म- फॉर्म या डाटाशीट के अंदर के क्षेत्रों को फिल्टर करने के लिए फिर किसी खास रिकॉर्ड को खोजने के लिए।

4) एडवांस फिल्टर- यहाँ आप खुद से फिल्टर के क्राइटेरिया को परिभाषित कर सकते हैं।

कॉमन फिल्टर का प्रयोग कैसे करें? (Use common filter in ms access)

किसी टेबल, क्वेरी, रिपोर्ट और फॉर्म को खोलें जिन्हें आप फिल्टर करना चाहते हैं। इन्हें उपर दिए गये चार व्यू में ही खोलें।

जहां आप फिल्टर करना चाहते हैं वहां किसी कॉलम पर click करें और Home टैब के अंदर Sort & Filter समूह में जाकर Filter पर click करें।

अब कॉमन फिल्टर अप्लाई करने के लिए Text, Number या Date पर पॉइंट करें और जो फिल्टर आप चाहते हैं उस पर click करें।

फिल्टर बाई सिलेक्शन कैसे करें? (Filter by selection in Ms Access)

किसी टेबल, क्वेरी, रिपोर्ट और फॉर्म को खोलें जिन्हें आप फिल्टर करना चाहते हैं। इन्हें उपर दिए गये चार व्यू में ही खोले।

इस बात का ध्यान रखें कि व्यू कहीं पहले से ही फिल्टर्ड तो नहीं है। रिकॉर्ड सिलेक्टर बार में जाकर ये जांच लें कि वहाँ Unfiltered या No Filter वाला आइकॉन उपस्थित है।

कॉलम के अंदर कहीं भी क्लिक कर के फिल्टर वाले ऑप्शन में जाएँ और Selection पर click करें।

फिल्टर बाई फॉर्म का प्रयोग कैसे करें? (Filter by form in ms access)

इसमें उपर दी गई प्रक्रिया को दोहराएँ, advanced के अंदर जाकर Filter by Form पर क्लिक करें और अगर आप डेटाबेस व्यू में हैं तो ऐसा करें:

कॉलम के पहले उस row पर click करें जिसे आप फिल्टर करना चाहते हैं। अब जो एरो खुलेगा उसमे से वैल्यू सेलेक्ट करें।

अगर आप फॉर्म व्यू में हैं तो ऐसा करें:

कण्ट्रोल के भीतर दिखने वाले एरो पर click करें और फिर जो वैल्यू फिल्टर करना चाहते हैं उन्हें सेलेक्ट करें।

किसी फिल्टर को हटाने के लियी या फिर सारे रिकार्ड्स देखने के लिए आप Toggle Filters पर click कर सकते हैं।

एमएस एक्सेस में एडवांस फ़िल्टर का उपयोग कैसे करें

(How to Use Advanced Filter in MS Access)

MS Access का सबसे शक्तिशाली फ़िल्टर Advanced Filter होता है, जिसका उपयोग एक से अधिक फ़ील्ड को सॉर्ट करने के लिए किया जाता है, जटिल फ़िल्टर Criteria और expressions का उपयोग करते हैं, साथ ही एक से अधिक And, OR विवरणों का उपयोग करते हैं। Advanced Filter के कई फायदे हैं। जैसे -

Sort by multiple fields (कई फ़ील्ड द्वारा सॉर्ट करें):

आप कई फ़ील्ड का उपयोग करके रिकॉर्ड्स सॉर्ट कर सकते हैं। उदाहरण के लिए, आप टेबल को अल्फाबेट तरीके से उसके First Name से टेबल को सॉर्ट कर सकते हैं।

Use complex filter criteria and expressions (जटिल फ़िल्टर मानदंडों और अभिव्यक्तियों का उपयोग करें):

आप डेटा की खोज के लिए advanced expressions और operators का उपयोग कर सकते हैं। उदाहरण के लिए, आप 20/11/2016 और 20/11/2018 के बीच आने वाली तिथियों के लिए फ़िल्टर लगाना सकते हैं।

Use multiple AND/OR statements (एकाधिक And / Or कथन का प्रयोग करें):

आप रिकॉर्ड्स के माध्यम से जाने के लिए एक से अधिक Criteria का उपयोग कर सकते हैं। उदाहरण के लिए, आप उन कर्मचारियों के लिए फ़िल्टर कर सकते हैं जो भोपाल से हैं और जो कंपनी के साथ पांच से अधिक वर्षों से रहे हैं।

प्रक्रिया:

उस टेबल को खोलें जिसमें वह डेटा है जिसे आप फ़िल्टर या सॉर्ट करना चाहते हैं और Advanced Filter विकल्प बटन पर क्लिक करें।

इसके बाद Advanced Filter / Sort का चयन करें।

फ़ील्ड सूची से आप जो फ़ील्ड शामिल करना चाहते हैं उसे डबल-क्लिक करें। नोट: फ़ील्ड जोड़ने के लिए फ़ील्ड सूची से फ़ील्ड पर फ़ील्ड ड्रैग भी खींच सकते हैं या डिज़ाइन ग्रिड की फ़ील्ड पंक्ति में लिस्ट तीर पर क्लिक कर सकते हैं और एक फ़ील्ड का चयन कर सकते हैं।

फ़ील्ड के लिए Sort list arrow पर क्लिक करें और Ascending, Descending में से किसी एक विकल्प का चयन करें।

Criteria Row में फ़ील्ड के लिए वांछित Search Criteria दर्ज करें।

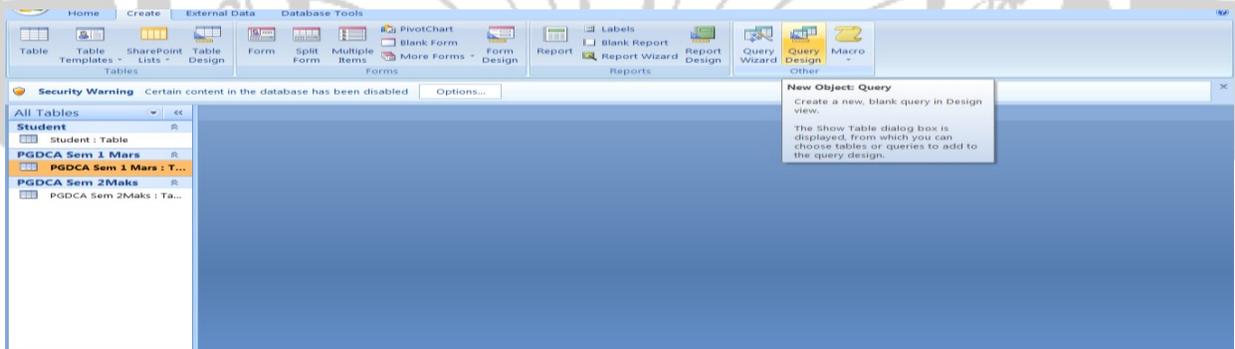
Apply Filter पर क्लिक करें। Advanced Filter लागू हो जायेगा।

एमएस एक्सेस में क्वेरी का निर्माण कैसे करें

(How to create Query in MS Access)

एमएस एक्सेस में क्वेरी का निर्माण करने के लिये हमें Create tab पर दो tools दिये होते हैं:

- 1) Query Design
- 2) Query Wizard



1) Query Design: इसके द्वारा हम एक new एवं blank query को Design view में बना सकते हैं।

2) Query Wizard: इसके द्वारा उपलब्ध wizard के माध्यम से हम simple query, crosstab query, a find duplicates query, एवं/या find unmatched query को आसानी से create कर सकते हैं।

Query Design द्वारा Select Query Creation:

SelectQuery

SelectQuery सबसे सरल प्रकार की क्वेरी है इसी कारण यह माइक्रोसॉफ्ट एक्सेस डेटाबेस में सबसे अधिक इस्तेमाल की जाती है। इसकी आवश्यकता के आधार पर इस का उपयोग किसी टेबल या डेटा की श्रृंखला को चुनने और प्रदर्शित करने के लिए किया जाता है। यह उपयोग कर्ता द्वारा निर्धारित मानदंड है जो डेटाबेस को बताता है कि चयन किस पर आधारित होना है।

SelectQuery के बाद यह एक “आभासी” टेबल बनाता है जहां डेटा बदला जा सकता है, लेकिन एक समय में एक से अधिक रिकॉर्ड नहीं।

एक अच्छी तरह से डिज़ाइन किए गए डेटाबेस में, डेटा जिसे आप किसी फॉर्म या रिपोर्ट के माध्यम से प्रस्तुत करना चाहते हैं, आमतौर पर कई tables में स्थित होता है। एक क्वेरी विभिन्न टेबलों से जानकारी ले सकती है और इसे फॉर्म या रिपोर्ट में प्रदर्शित करने के लिए इकट्ठा कर सकती है। क्वेरी आपको एक साधारण प्रश्न का उत्तर देती है, गणना कर सकती है, विभिन्न टेबलों से डेटा इकट्ठा कर सकती है, डेटाबेस से डेटा जोड़, बदल सकती है या हटा सकती है।

एम एस एक्सेस में सिलेक्ट क्वेरी का उपयोग कैसे करें

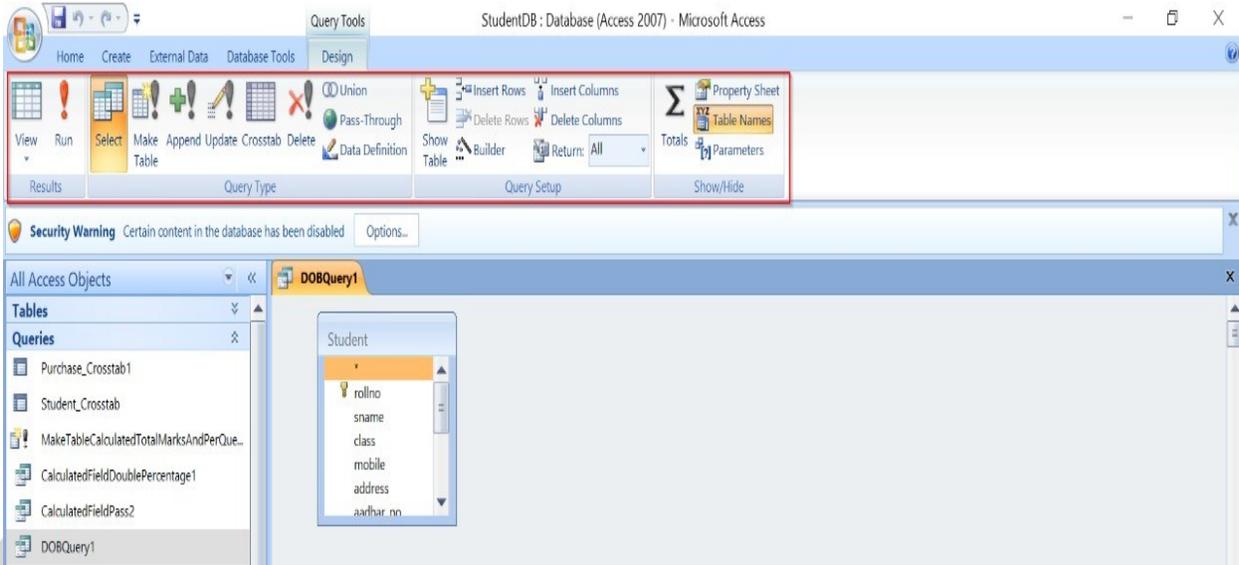
(How to Use a Select Query in MS Access)

यदि आप टेबल में केवल कुछ फ़ील्ड से डेटा देखना चाहते हैं, या एक साथ कई टेबलों से डेटा देखना चाहते हैं या कुछ मानदंडों के आधार पर डेटा देखना चाहते हैं, तो आप Select Query का प्रयोग कर सकते हैं। उदाहरण के लिए, यदि आपके डेटाबेस में Students के बारे में जानकारी वाली टेबल है और आप Student के नाम और उनके mobile no तथा एक विशिष्ट Date of Birth वाले Students की एक सूची देखना चाहते हैं, तो आप यह सिलेक्ट क्वेरी से देख सकते हैं।

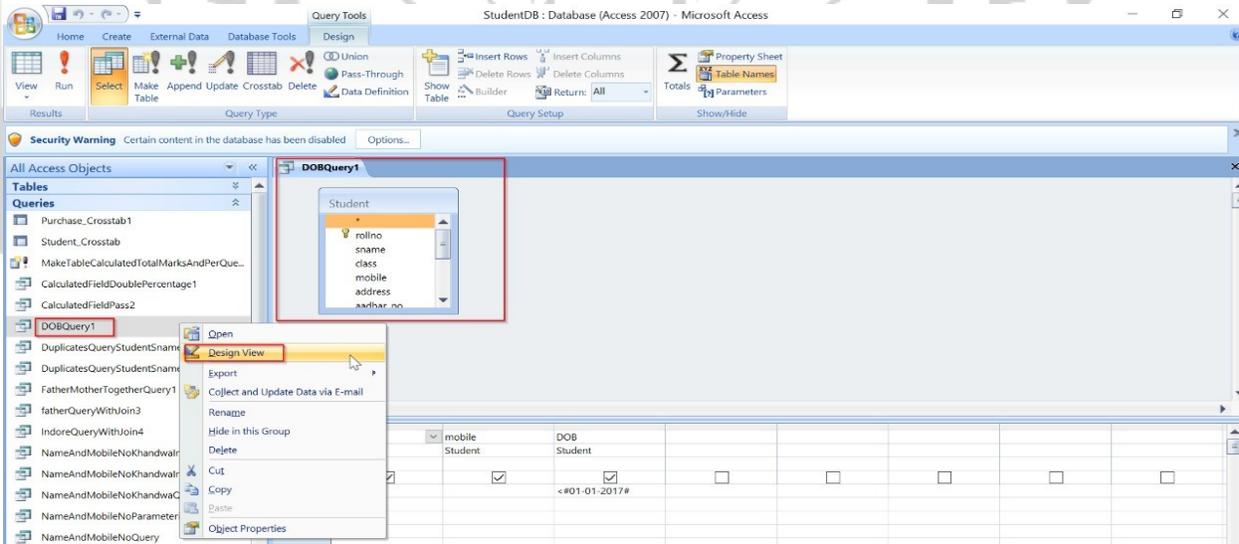
क्वेरी का उपयोग कैसे करें

- 1) Create tab के Query Design पर click करें।
- 2) Show Table box आता है जिससे हम query के लिये tables का selection कर

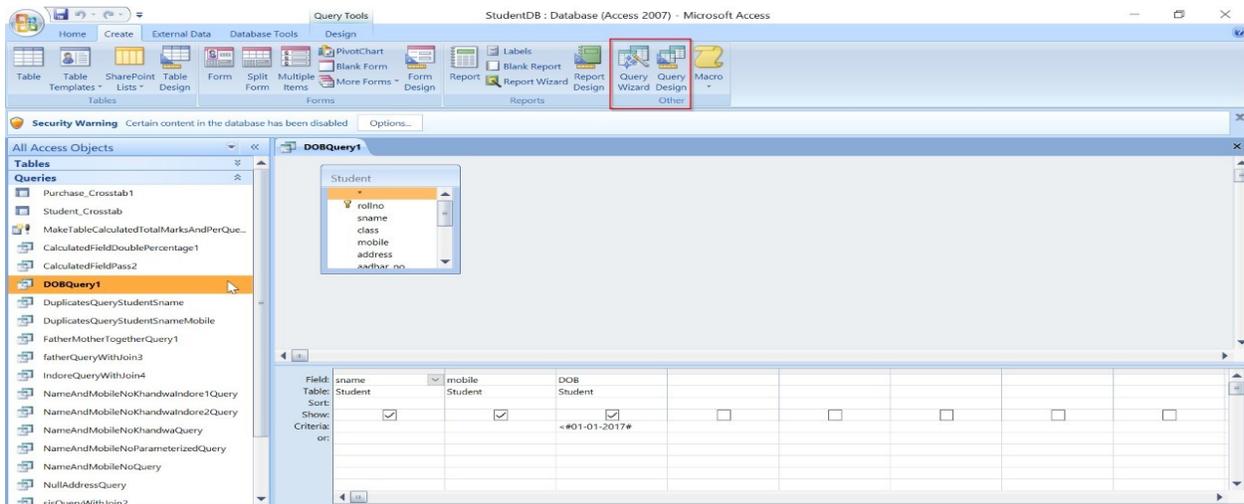
सकते हैं।



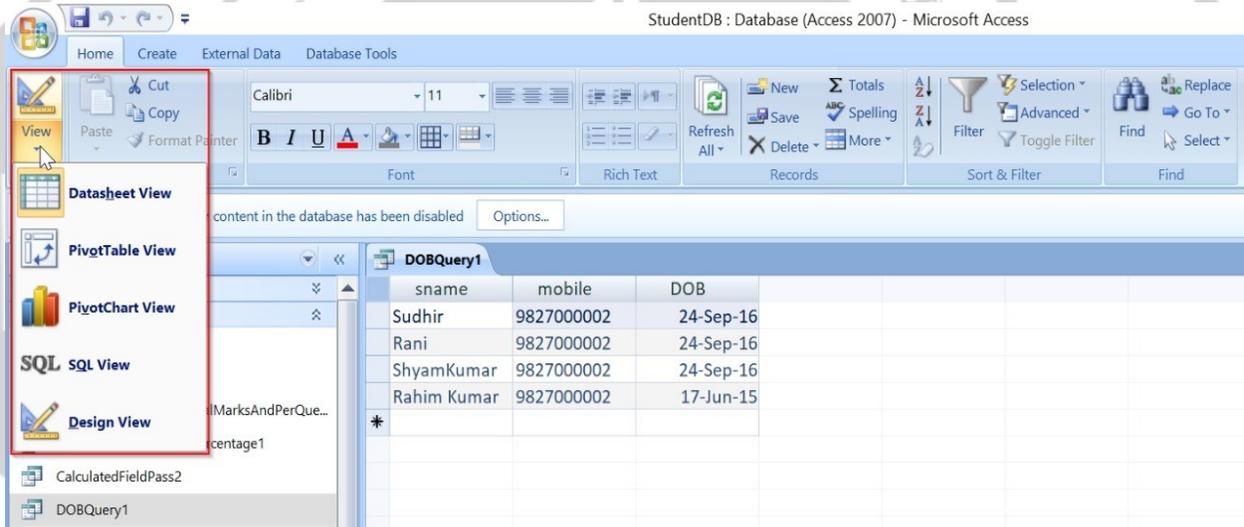
3) Design view में जाकर Table में से query के लिये उपयुक्त fields को selection करेंगे।



4) अब किसी विशिष्ट field के लिये criteria set करेंगे जैसे यहां DOB field के लिये किया गया है।



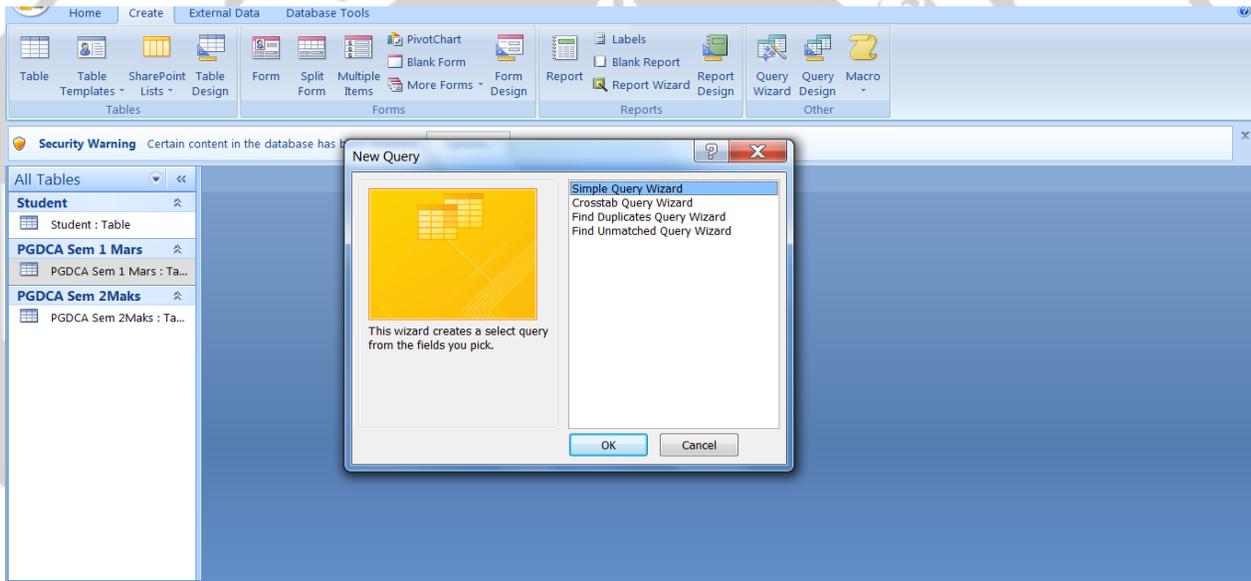
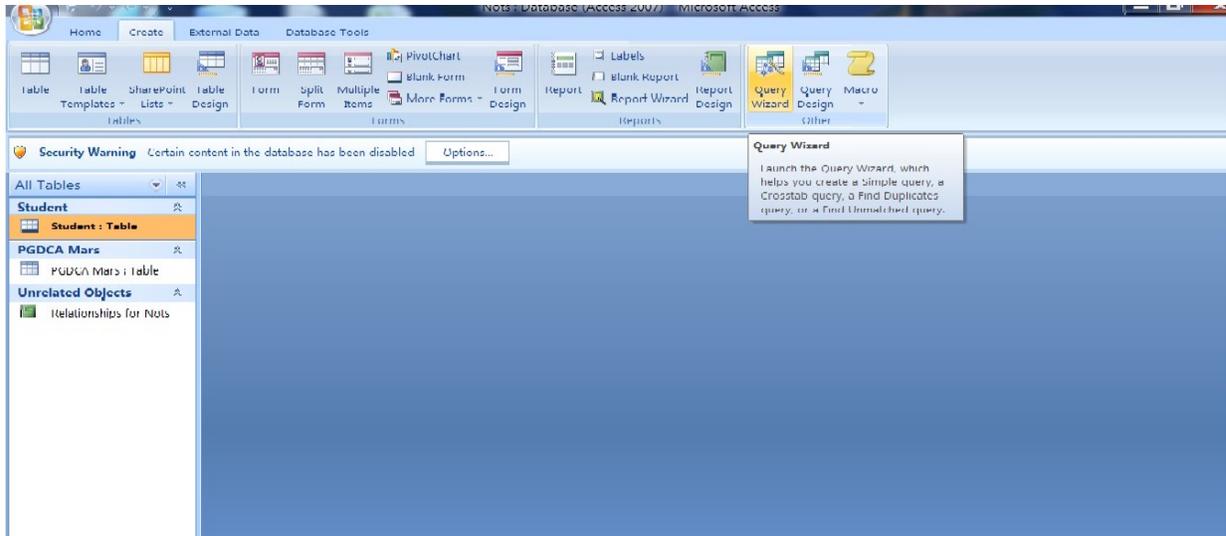
5) अब हम query को Datasheet view में open करेंगे जिस में दिये गए criteria अनुसार row and column format में प्रदर्शित होंगे।



6) Query tab पर right click करके save कर Design tab से Run किया जा सकता है।

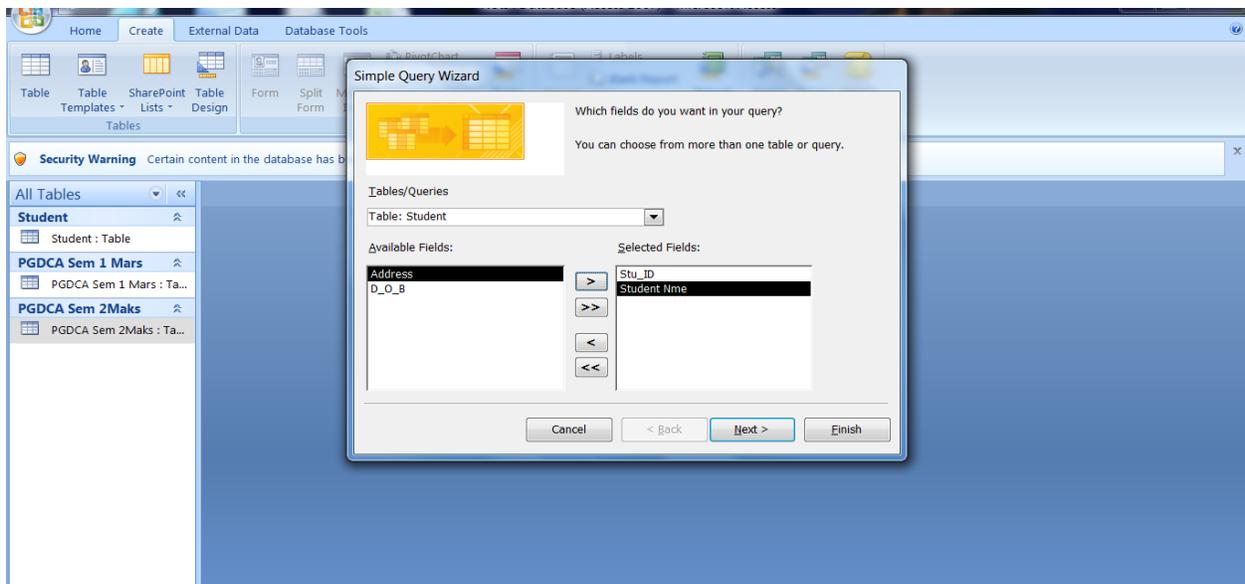
Query wizard द्वारा Query Creation

MS ACCESS में create tab पर Query Wizard पर click करते ही निम्न Box open होता है जो हमें विभिन्न प्रकार से query creation के options प्रदान करता है:

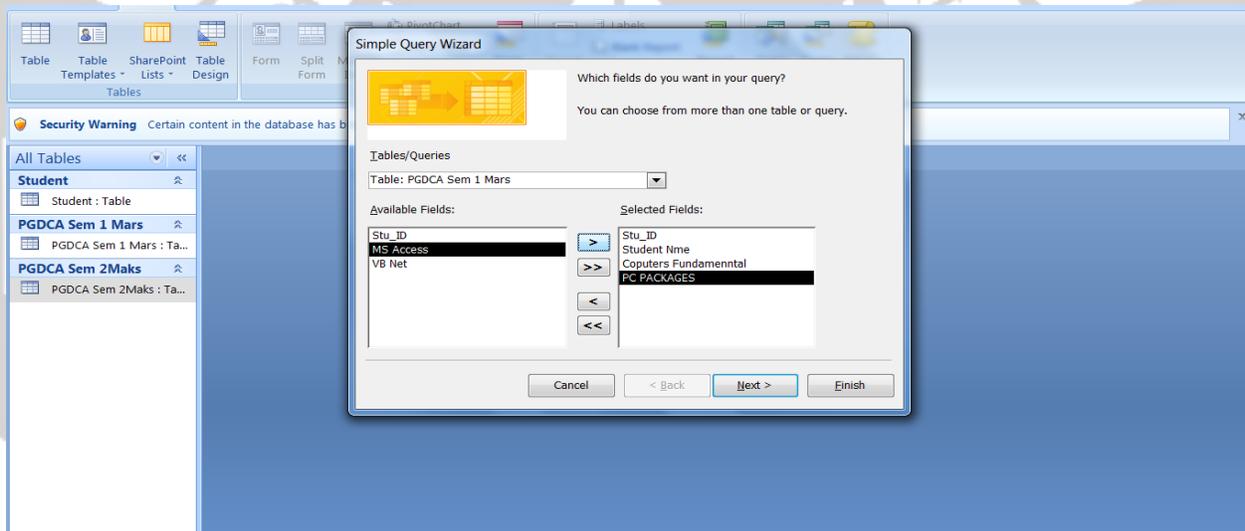


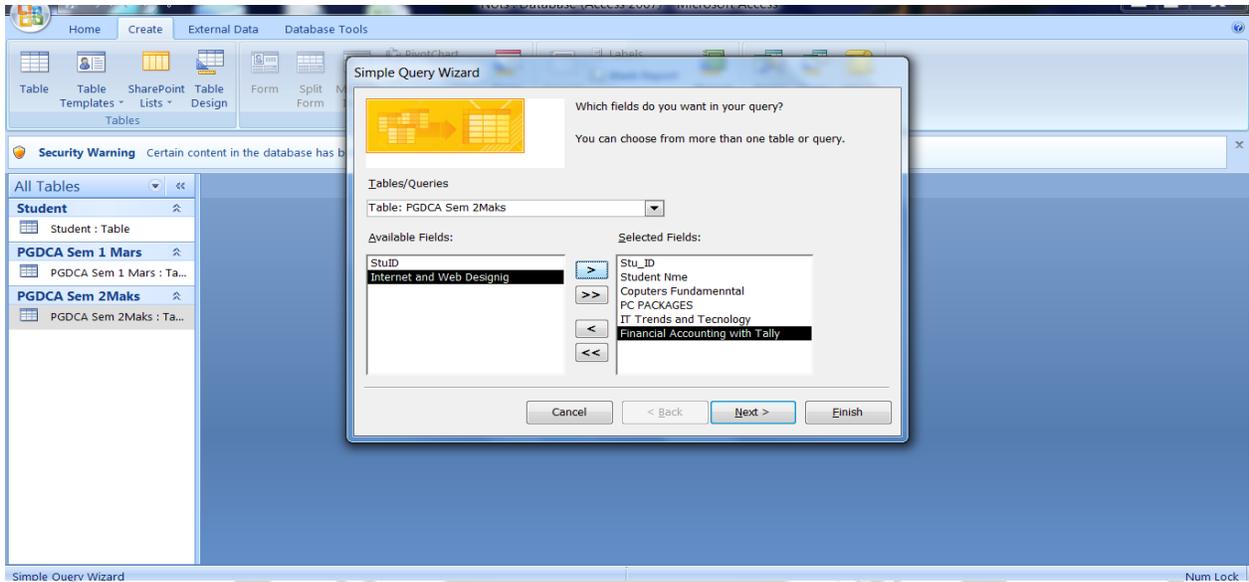
Simple Query Wizard द्वारा Query Creation:

1) Query Wizard में Simple Query Wizard पर click करते ही tables एवं उनकी fields selection के लिये screen open होती है जिसमे हम पहले प्रथम table को select कर उसकी fields जो हमें अपनी query में चाहिये Greater sign (>) द्वारा select करते हैं।

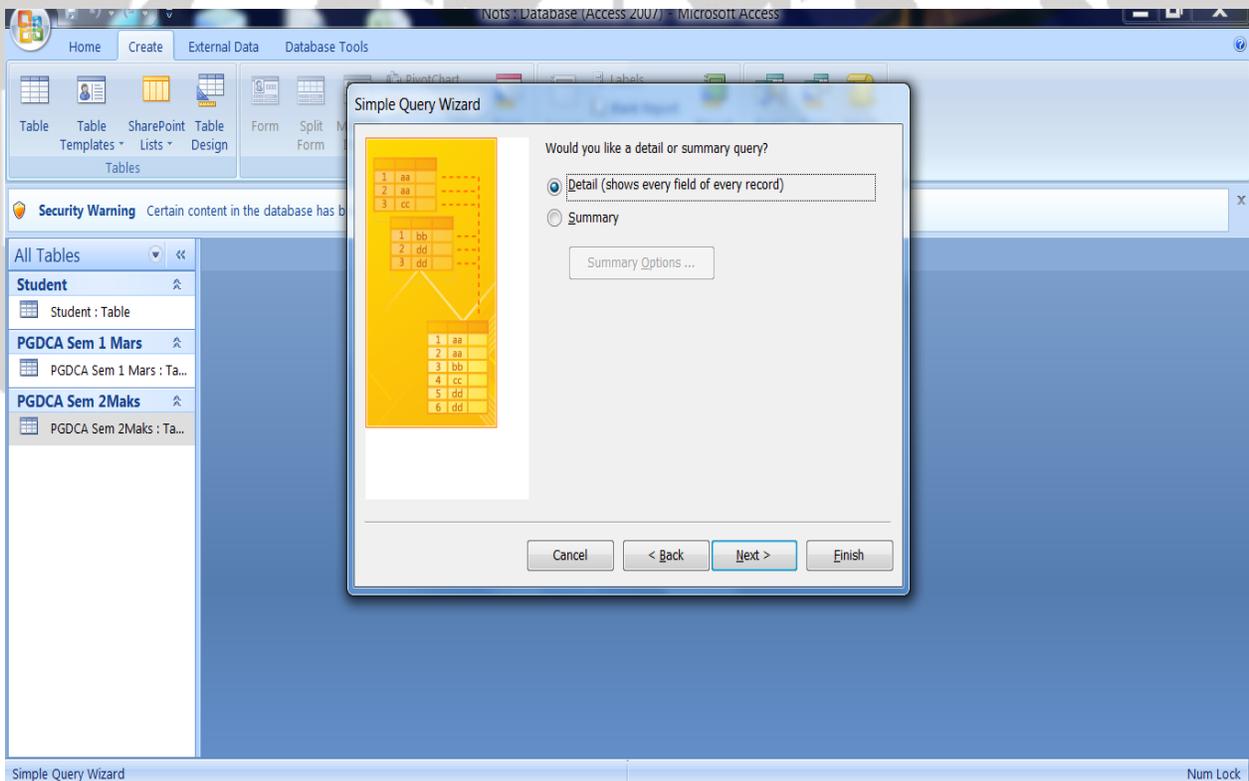


2) इसी तरह अन्य दो tables एवं उनकी fields को भी select करते हैं।

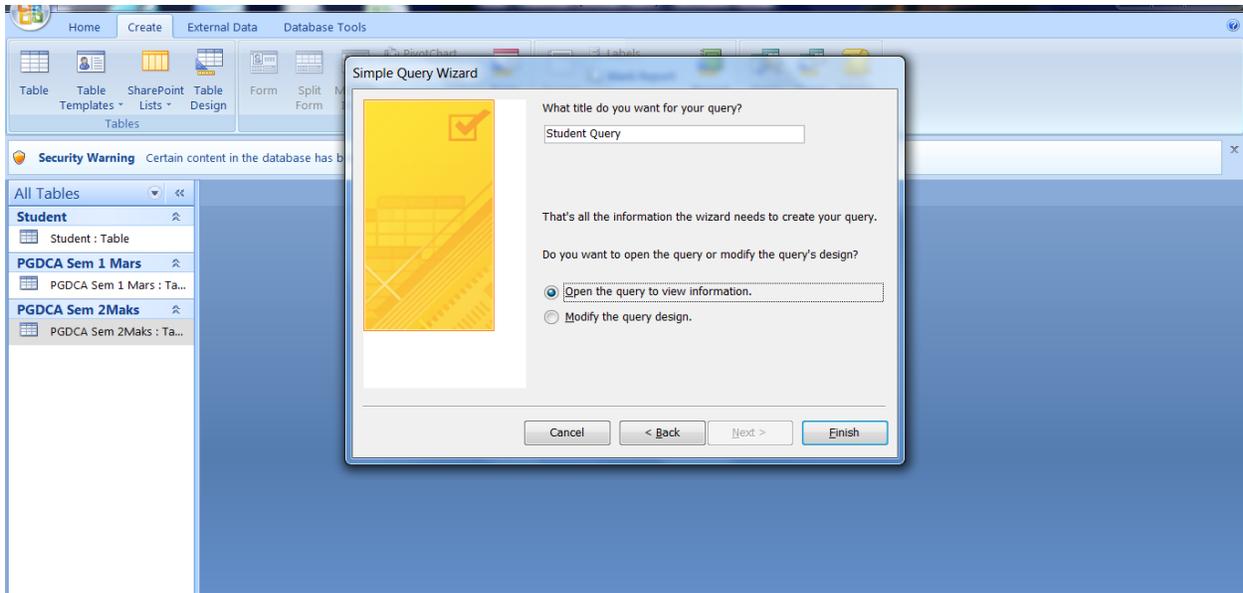




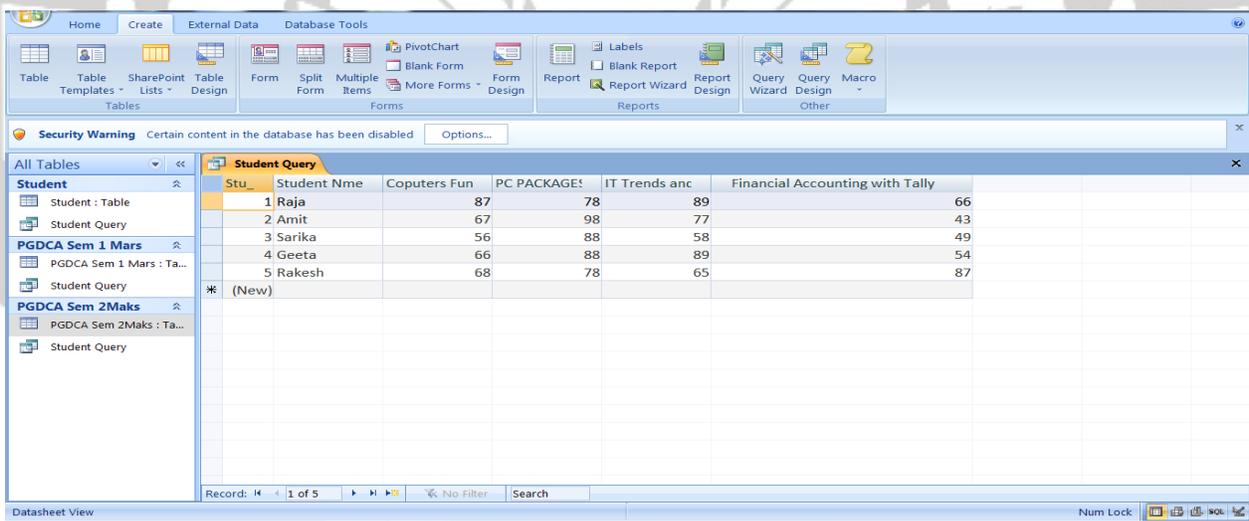
3) अब Next पर click करते हैं। तथा Detail या Summary जिस प्रकार Query चाहिये उसका चयन कर Next पर click करते हैं।



4) अब query को नाम देकर open को select कर finish पर click करते हैं।



5) Finish पर click करते ही query Data sheet view में प्रदर्शित होगी जिसमें select किये tables की selected विशिष्ट fields के अनुसार records प्रदर्शित होंगे।



6) Query name पर right click करके हम उसे Design view में निम्नतरहसे देख सकते हैं।

The screenshot shows the Microsoft Access interface with a table view. A context menu is open over the 'Student' table, listing options such as Open, Design View, Export, and Object Properties. The table data is as follows:

	Computers Fun	PC PACKAGE!	IT Trends anc	Financial Accounting with Tally
Student :	87	78	89	66
Student Q	67	98	77	43
PGDCA Sem 1	56	88	58	49
PGDCA Se	66	88	89	54
Student Q	68	78	65	87

The screenshot shows the Microsoft Access interface in Query Design View. The 'Student Query' is linked to three tables: 'Student', 'PGDCA Sem 2Maks', and 'PGDCA Sem 1 Mars'. The field list at the bottom is as follows:

Field:	Stu_ID	Student Nme	Coputers Fundamentall	PC PACKAGES	IT Trends and Tecnolog	Financial Accounting wit
Table:	Student	Student	PGDCA Sem 1 Mars	PGDCA Sem 1 Mars	PGDCA Sem 2Maks	PGDCA Sem 2Maks
Sort:						

फाइंड डुप्लिकेट क्वेरी क्या है? (What is Find Duplicates Query?)

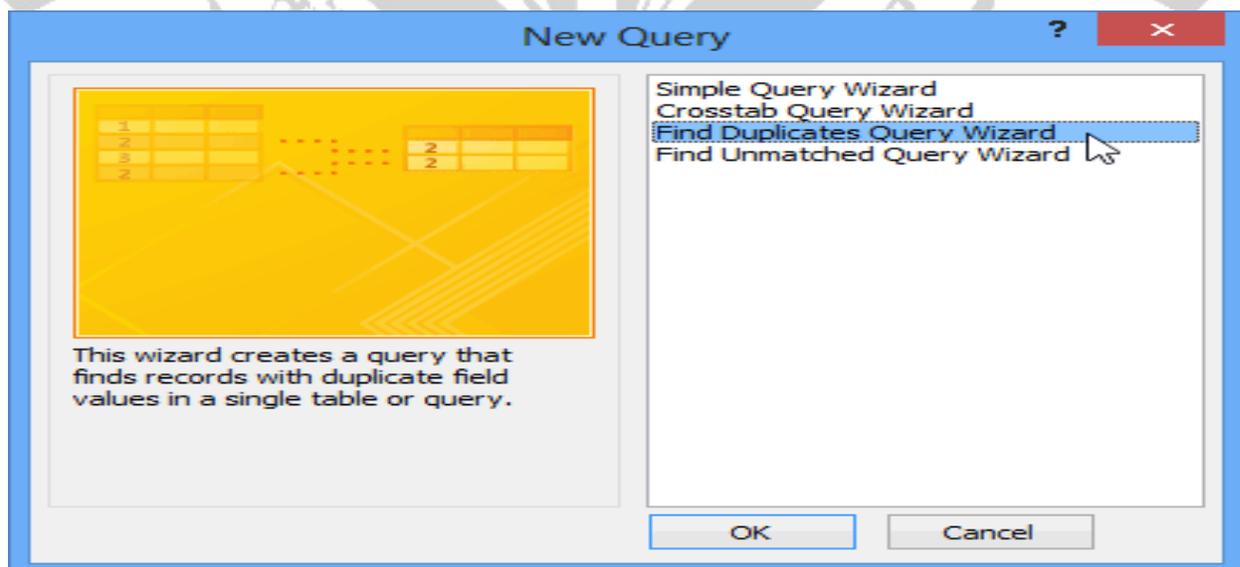
फाइंड डुप्लिकेट क्वेरी आपको टेबल या टेबल्स के भीतर डुप्लिकेट रिकॉर्ड खोजने और पहचानने की अनुमति देती है। डुप्लिकेट रिकॉर्ड एक रिकॉर्ड होता है जो एक ही चीज़ या व्यक्ति को दूसरे रिकॉर्ड के रूप में संदर्भित करता है।

समान जानकारी वाले सभी रिकॉर्ड डुप्लिकेट नहीं हैं। उदाहरण के लिए, अलग-अलग तिथियों पर रखे गए दो कमांड्स के रिकॉर्ड, लेकिन समान कंटेंट में डुप्लिकेट रिकॉर्ड नहीं होंगे। इसी प्रकार, सभी डुप्लिकेट रिकॉर्ड में पूरी तरह से समान जानकारी नहीं होती है। उदाहरण के लिए, दो ग्राहक रिकॉर्ड एक ही व्यक्ति को संदर्भित कर सकते हैं लेकिन अलग-अलग पते शामिल कर सकते हैं। पुराने पते के साथ रिकॉर्ड डुप्लिकेट रिकॉर्ड होगा।

अगर हमारे पास एक ग्राहक के लिए कई रिकॉर्ड हैं, तो उनके लिए ऑर्डर हिस्ट्री देखना मुश्किल होगा क्योंकि जानकारी कई अनलिक रिकॉर्ड्स में फैल जाएगी। यदि ऑर्डर जानकारी दर्ज करने वाला व्यक्ति पुराना रिकॉर्ड चुनता है तो हम गलत पते पर अपना ऑर्डर भी दे सकते हैं। यह देखना आसान है कि डुप्लिकेट रिकॉर्ड कैसे आपके डेटाबेस की अखंडता और उपयोगिता को कम कर सकते हैं।

सौभाग्य से, एक्सेस संभावित डुप्लिकेट रिकॉर्ड खोजने और ढूंढना आसान बनाता है। ध्यान दें कि एक्सेस आपके लिए रिकॉर्ड नहीं हटाएगी या यह पता लगाने में आपकी सहायता करेगी कि कौन सा चालू है-आपको अपने लिए उन चीजों को करना होगा। यदि आप अपने डेटाबेस में डेटा से परिचित हैं, हालांकि, डुप्लिकेट रिकॉर्ड से छुटकारा पाने के लिए एक प्रबंधनीय कार्य होगा।

एमएस एक्सेस में फाइंड डुप्लिकेट क्वेरी कैसे बनाएँ

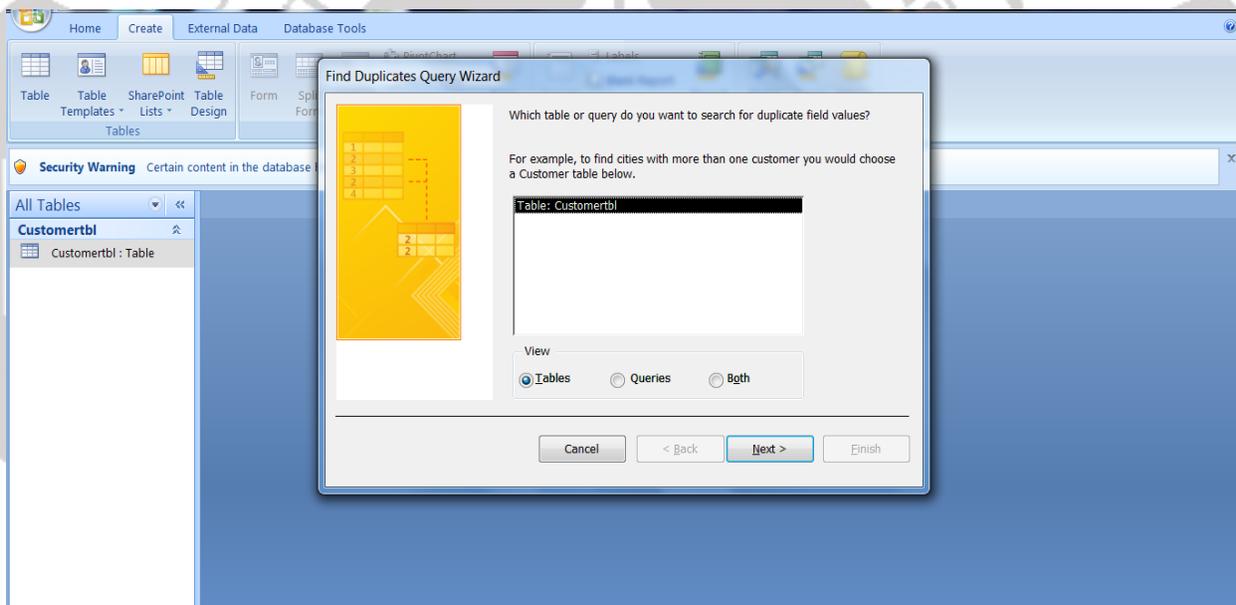


(How to Create a Find Duplicates Query in MS Access)

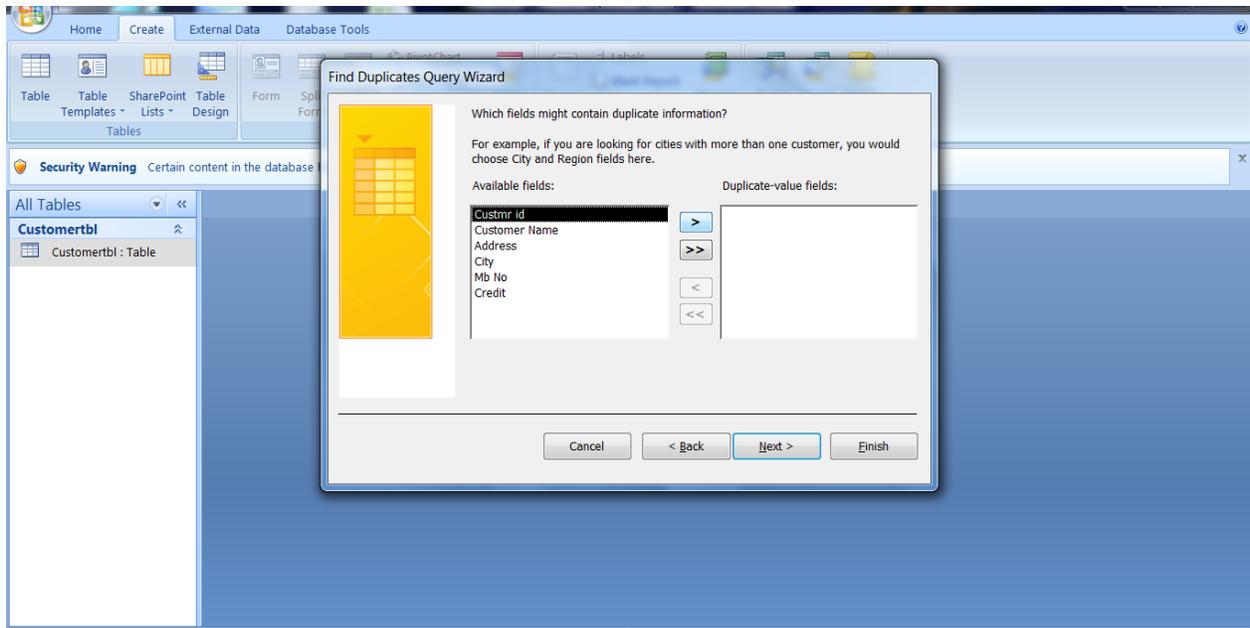
रिबन पर Create Tab पर स्थित Query group में से Query Wizard command पर क्लिक करें।

New Query डायलॉग बॉक्स दिखाई देगा। Query की सूची से Find Duplicates Query Wizard का चयन करें, फिर Ok पर click करें।

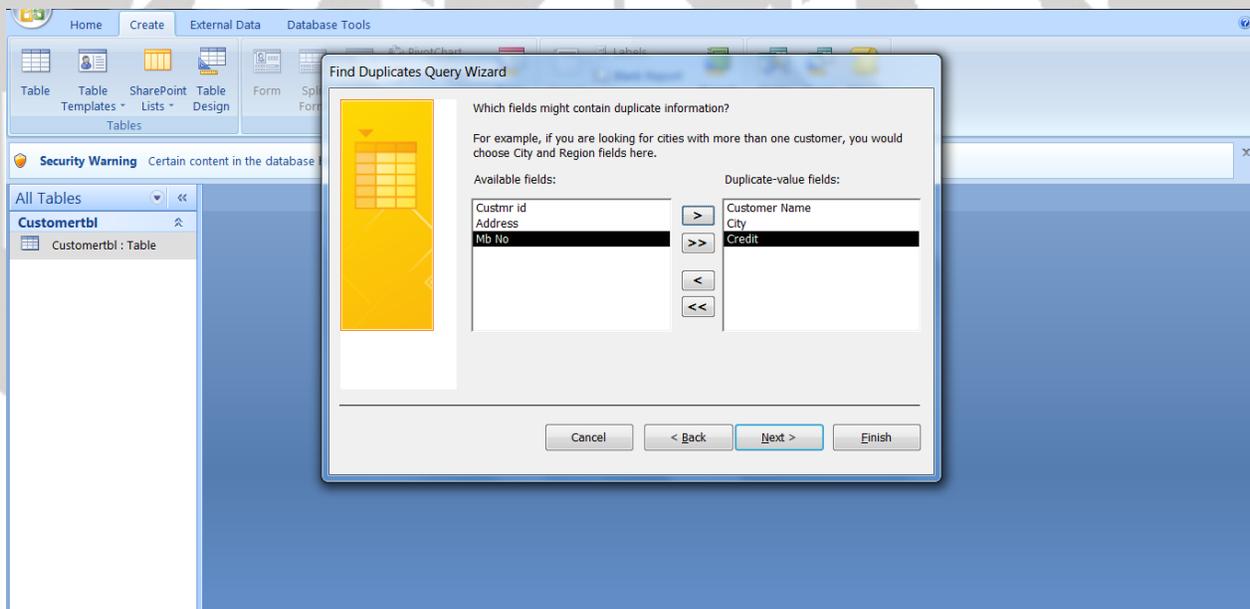
वह टेबल चुनें जिसे आप डुप्लिकेट रिकॉर्ड्स के लिए खोजना चाहते हैं, फिर Next पर क्लिक करें। हम डुप्लिकेट ग्राहक रिकॉर्ड खोज रहे हैं, इसलिए हम ग्राहक टेबल का चयन करेंगे



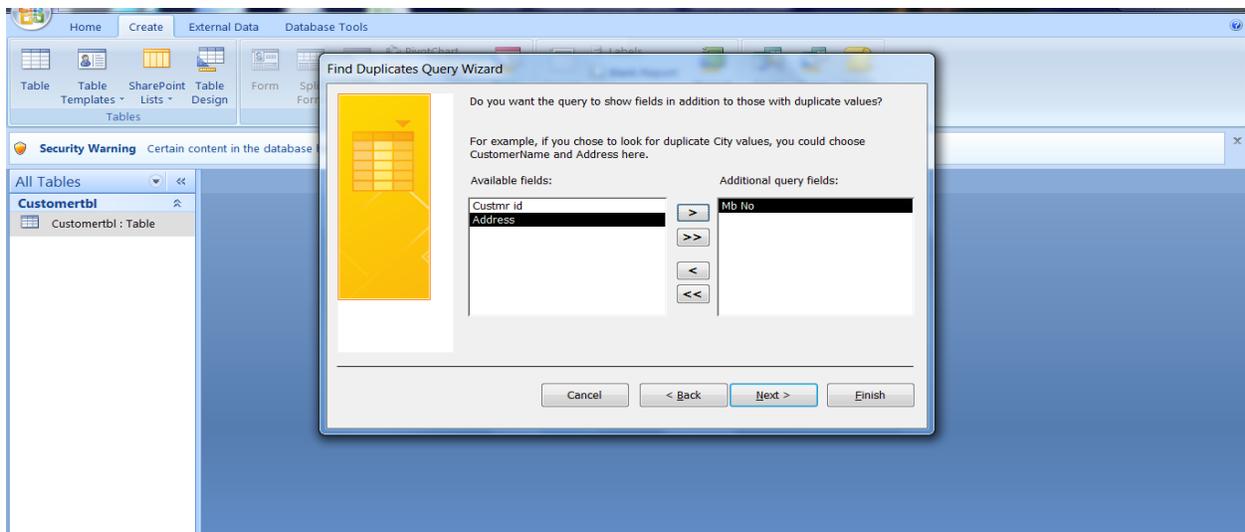
Find Duplicate Query Wizard विंडो में से उन फ़ील्ड को चुनें जिन्हें आप Duplicate value field में भेजना चाहते हैं फिर दायां तीर बटन पर क्लिक करें। केवल उन फ़ील्ड का चयन करें जो nonduplicate रिकॉर्ड में समान नहीं होना चाहिए। उदाहरण के लिए, हम डुप्लिकेट ग्राहकों की खोज कर रहे हैं, हम केवल customer name, city और credit फ़ील्ड्स का चयन करेंगे ।



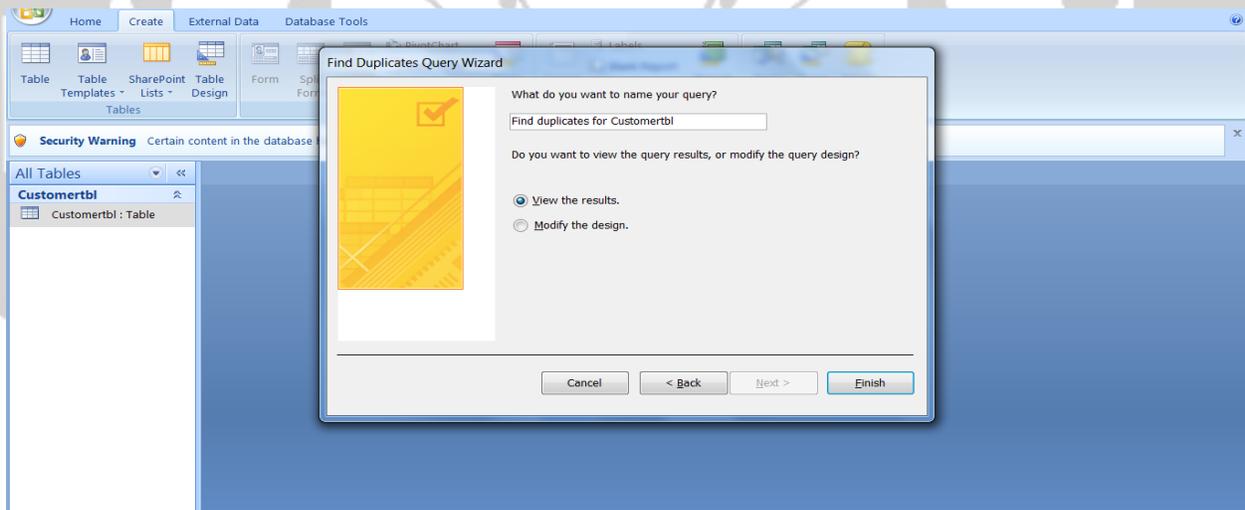
जब आप वांछित फ़ील्ड जोड़ ले उसके बाद Next पर क्लिक करें।



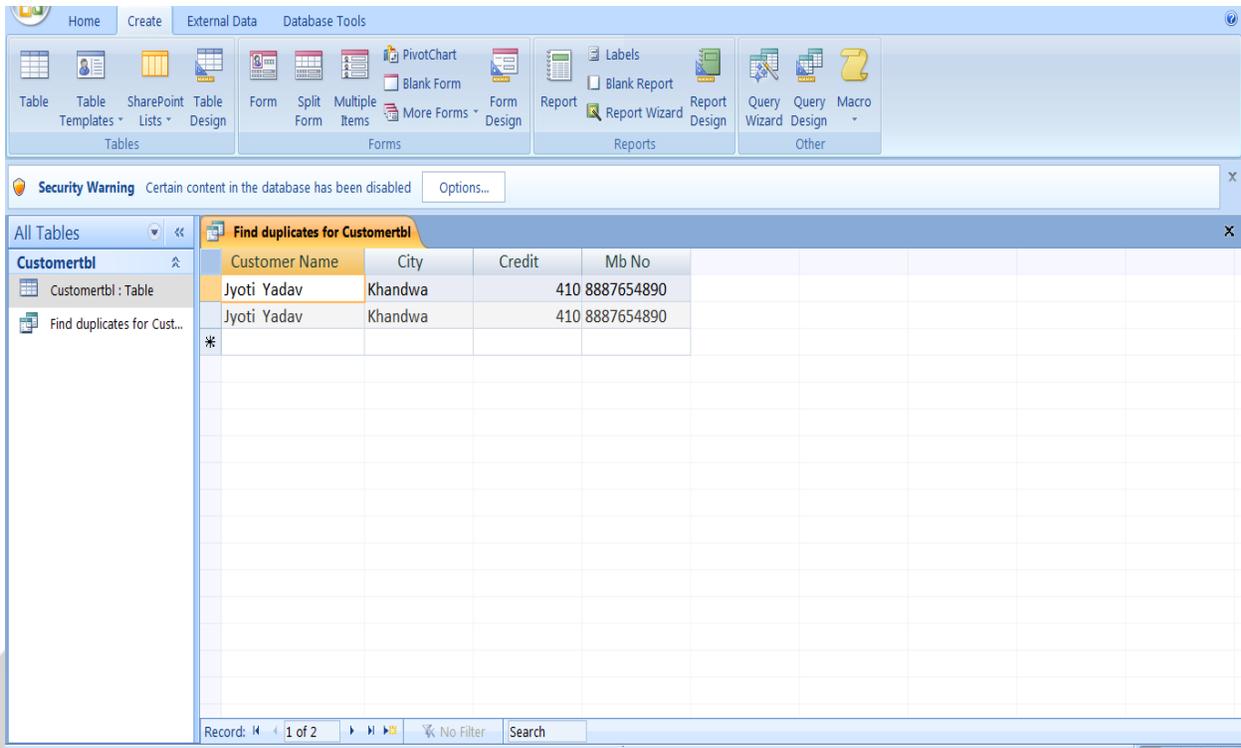
क्वैरी रिजल्ट्स में देखने के लिए additional fields का चयन करें। उन फ़ील्ड्स चुनें जो आपको डुप्लिकेट रिकॉर्ड के बीच अंतर करने में मदद करेंगे, और चुनें कि आप कौन सी रखना चाहते हैं। हमारे उदाहरण में, हम ग्राहक पते से संबंधित सभी फ़ील्ड, साथ ही mobile नंबर फ़ील्ड जोड़ देंगे क्योंकि समान ग्राहक नाम वाले रिकॉर्ड में इस क्षेत्र में गैर-कानूनी जानकारी हो सकती है। जब आप संतुष्ट हों, तो next पर क्लिक करें।



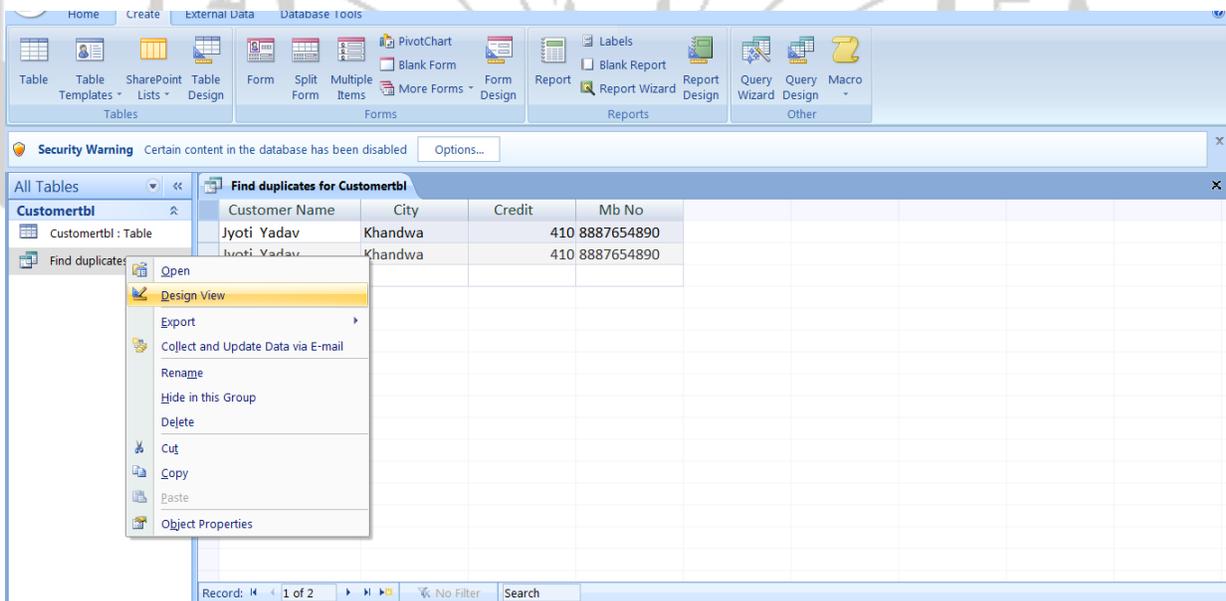
इसके बाद एक्सेस आपसे आपकी क्वेरी के लिए एक नाम पूछेगा, लेकिन यदि आप चाहें तो एक अलग नाम टाइप कर सकते हैं। जब आप क्वेरी नाम से संतुष्ट होते हैं, तो अपनी क्वेरी चलाने के लिए Finish पर क्लिक करें।



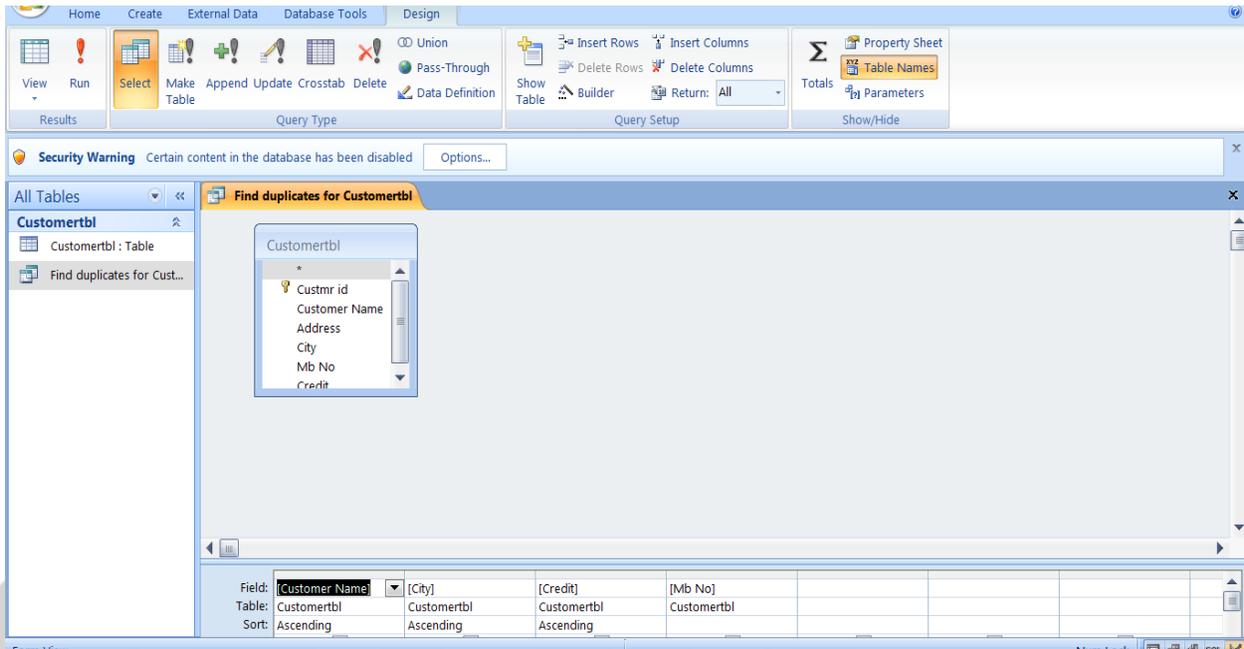
यदि एक्सेस को आपकी क्वेरी में कोई डुप्लिकेट रिकॉर्ड मिला, तो वे क्वेरी रिजल्ट्स में प्रदर्शित होंगे। रिकॉर्ड्स की समीक्षा करें और आवश्यकतानुसार किसी भी पुराने या गलत रिकॉर्ड को हटा दें।



अब query पर Right click करके हम उससे Design view में भी देख सकते हैं।



Design view निम्न प्रकार से प्रदर्शित होगा :

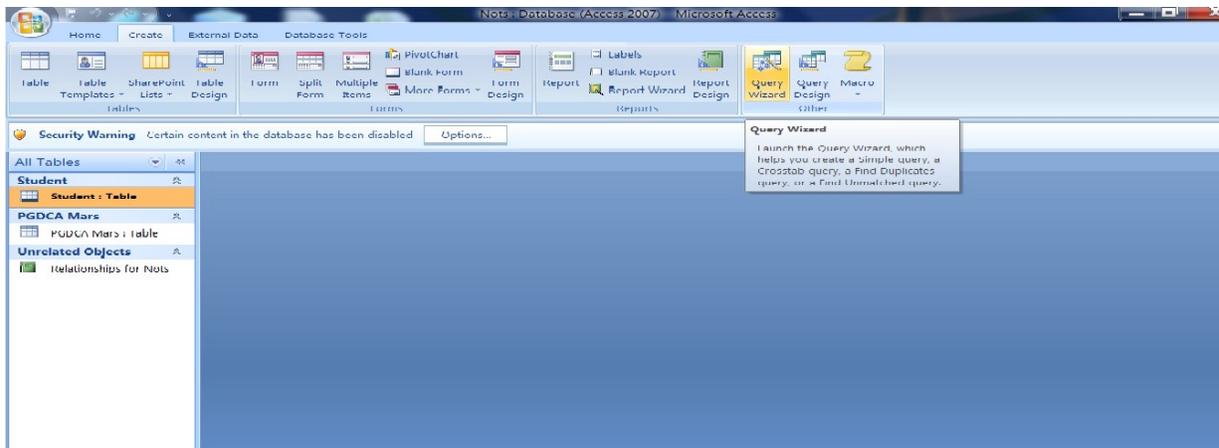


एमएस एक्सेस में Unmatched Query Wizard का उपयोग कैसे करें

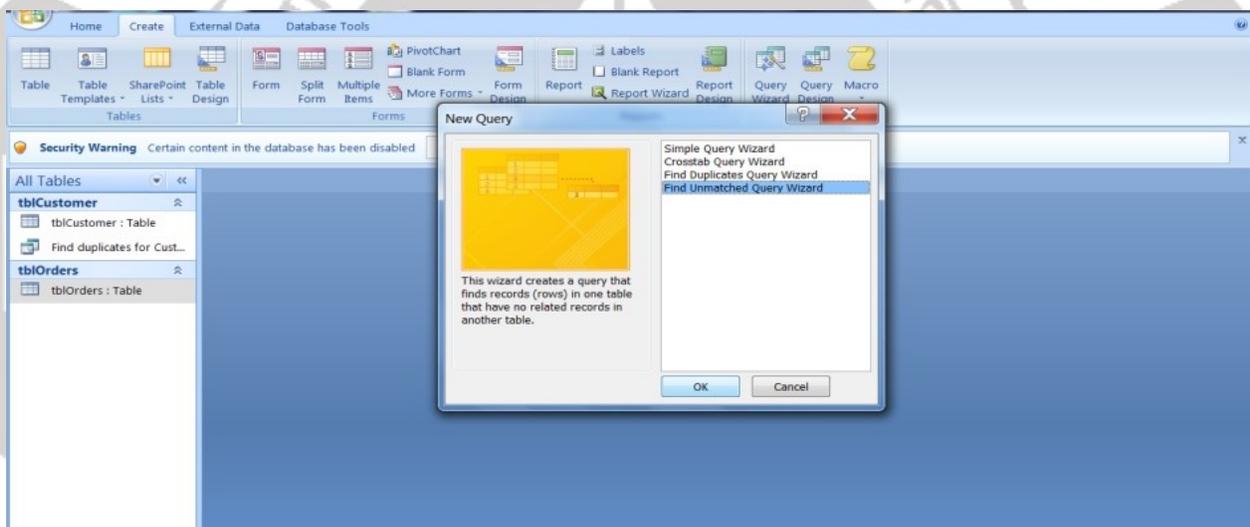
(How to Use Unmatched Query Wizard in MS Access)

MS Access में, एक और बहुत उपयोगी विज़ार्ड है जिसका नाम Find Unmatched Query Wizard है। Find Unmatched Query Wizard एक क्वेरी बनाता है जो एक टेबल में रिकॉर्ड या पंक्तियां पाता है जिसमें किसी अन्य टेबल में कोई संबंधित रिकॉर्ड नहीं होता है। यह एक्सेस में डिफॉल्ट जुड़ाव है, उदाहरण के लिए, यदि हम दो टेबल, Customers और Orders के साथ एक क्वेरी डिज़ाइन करते हैं, और ग्राहक आईडी द्वारा उन दो टेबल्स में शामिल होते हैं, तो यह क्वेरी केवल उन परिणामों को वापस लाएगी जो मेल खाते हैं। दूसरे शब्दों में, जिन ग्राहकों ने आदेश दिया है।

सबसे पहले अपना डेटाबेस खोलें जिसमें ग्राहक और ऑर्डर टेबल हो। Create tab पर स्थित Query Wizard बटन पर क्लिक करें ।

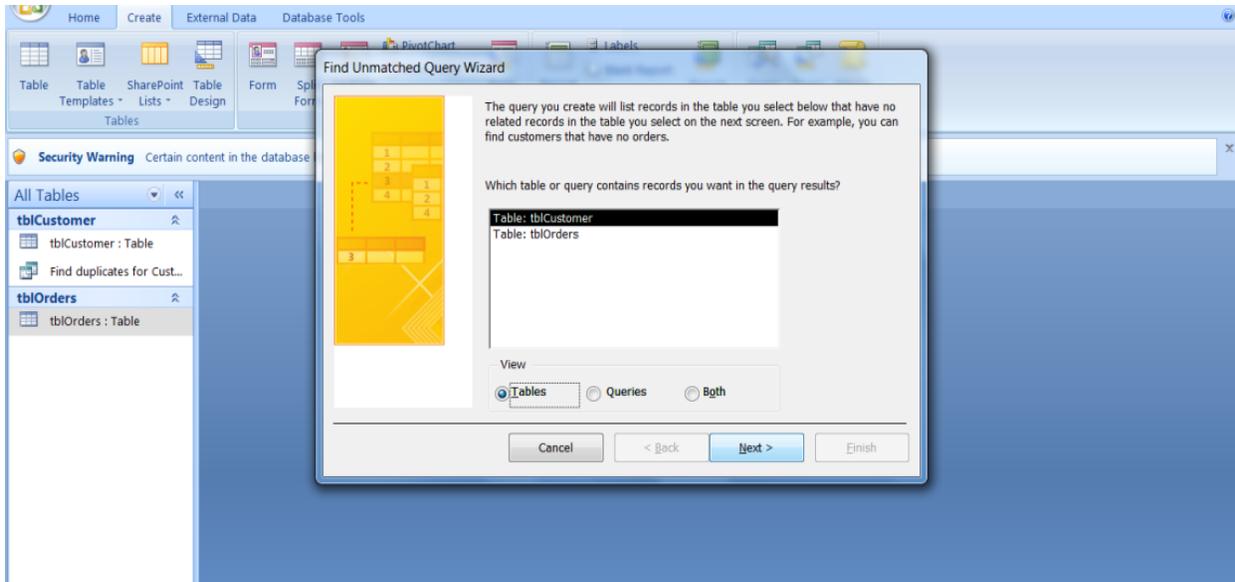


New Query विज़ार्ड डायलॉग बॉक्स ओपन हो जायेगा इसमें Find Unmatched Query Wizard विकल्प पर क्लिक करें और Ok पर क्लिक करें।

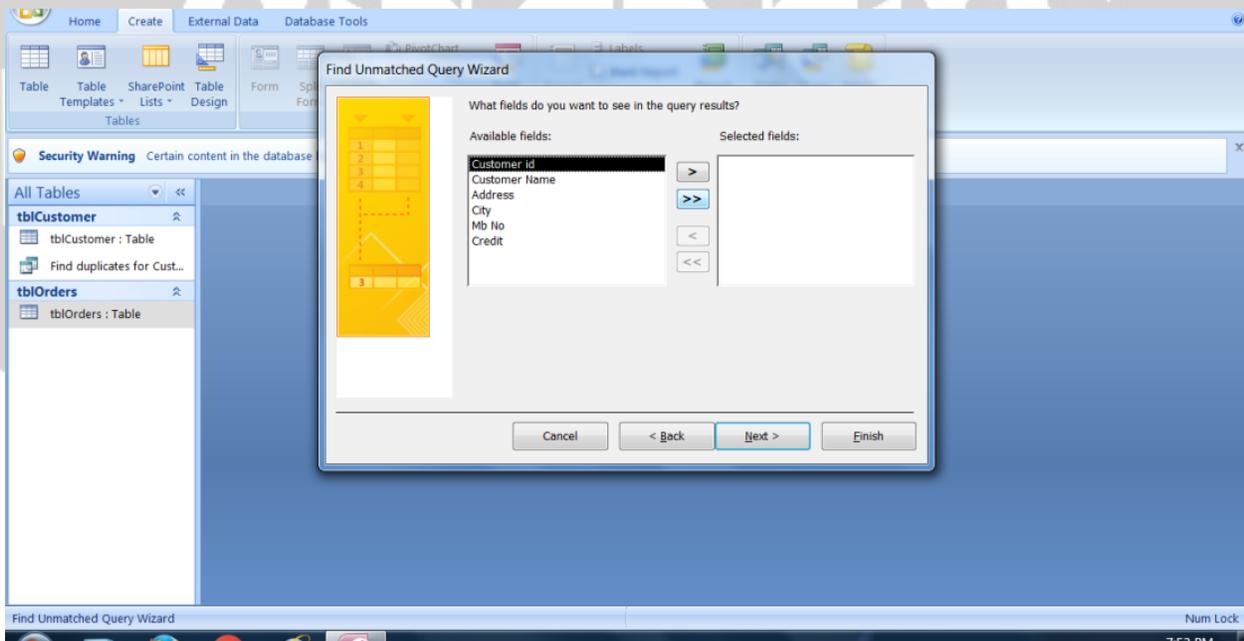


हम उन ग्राहकों के लिए देखेंगे जिन्होंने ऑर्डर नहीं दिया है। इस विंडो में वह टेबल या क्वेरी सिलेक्ट करें जिसमें आपका रिकॉर्ड्स शामिल हैं।

हम tblCustomers से ग्राहकों की एक सूची चाहते हैं। उस विकल्प का चयन करें और Next पर क्लिक करें।

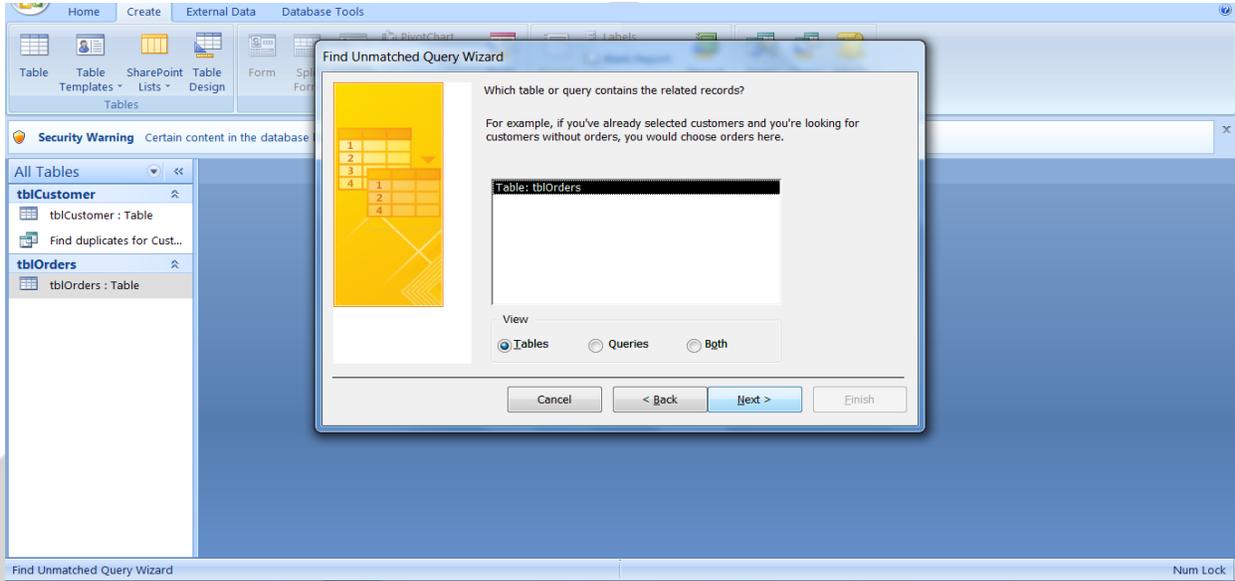


अब हम cutomer table से fields को select करेंगे।



निम्न स्क्रीन में, आपको यह निर्दिष्ट करने की आवश्यकता है कि कौन सी टेबल या क्वेरी में संबंधित रिकॉर्ड हैं। दूसरे शब्दों में, आप पहली बार तुलना करने के लिए किस टेबल का उपयोग कर रहे हैं। इसके लिए, हमें उन लोगों को ढूँढना होगा जिन्होंने ऑर्डर नहीं दिए हैं। हमें उस

टेबल का चयन करने की आवश्यकता है जिसमें सभी ऑर्डर - पर जानकारी शामिल है इसलिए हम tblOrders टेबल सिलेक्ट करेंगे फिर Next पर क्लिक करें।



निम्नलिखित स्क्रीन में, आपको यह निर्दिष्ट करना होगा कि दोनों टेबलओं में कौन सी जानकारी है।

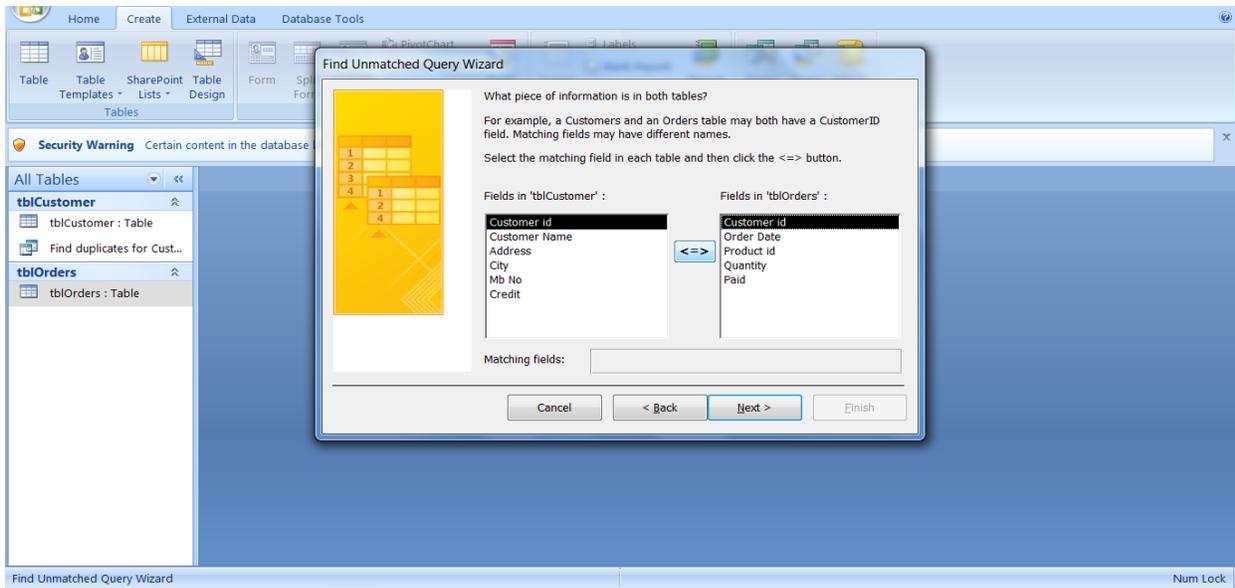
यह आमतौर पर किसी प्रकार की Primary key, Foreign key या Relationship होगा।

यदि आपके डेटाबेस में मौजूदा Relationship हैं, तो एक्सेस आपके लिए उन फ़ील्ड का चयन और मिलान करेगा।

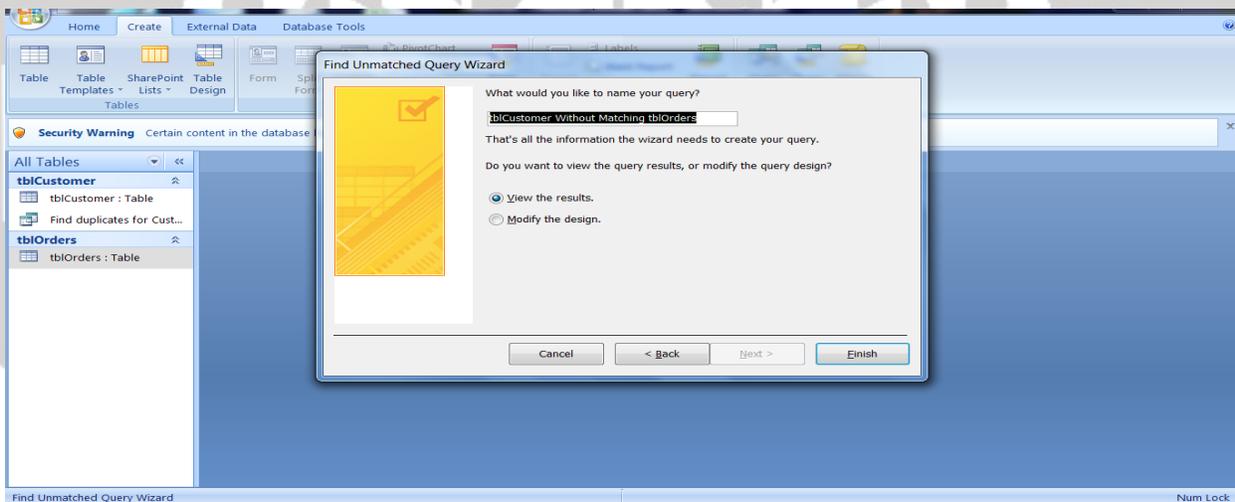
लेकिन, यदि आपके पास अन्य फ़ील्ड हैं जो आप एक साथ शामिल हो सकते हैं, तो समान जानकारी शामिल है, आप इसे यहां भी चुन सकते हैं।

यहां, हमारे पास 'tblCustomers' और 'tblOrders' दोनों फ़ील्ड में डिफ़ॉल्ट रूप से चयनित CustID है। इसके बाद Next पर क्लिक करें। निम्न स्क्रीन में, आप उन फ़ील्ड को चुन सकते हैं जिन्हें आप क्वेरी परिणामों में प्रदर्शित करना चाहते हैं।

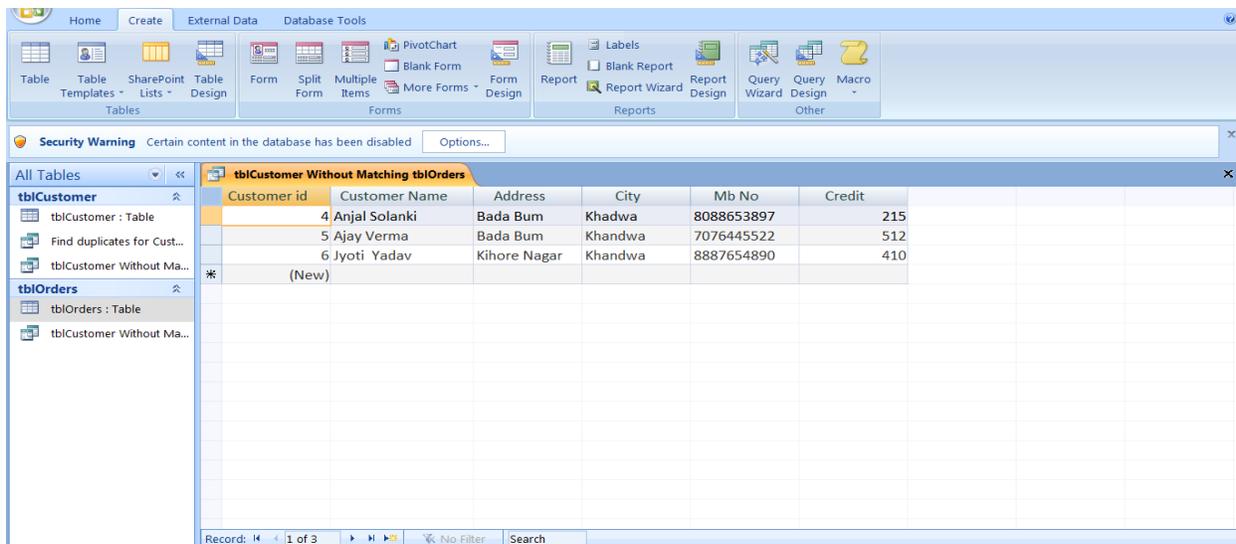
अब सभी उपलब्ध फ़ील्ड का चयन करें और डबल हेड वाले तीर पर क्लिक करें। यह सभी Available fields को Selected fields में ले जाता है। अब, next पर क्लिक करें।



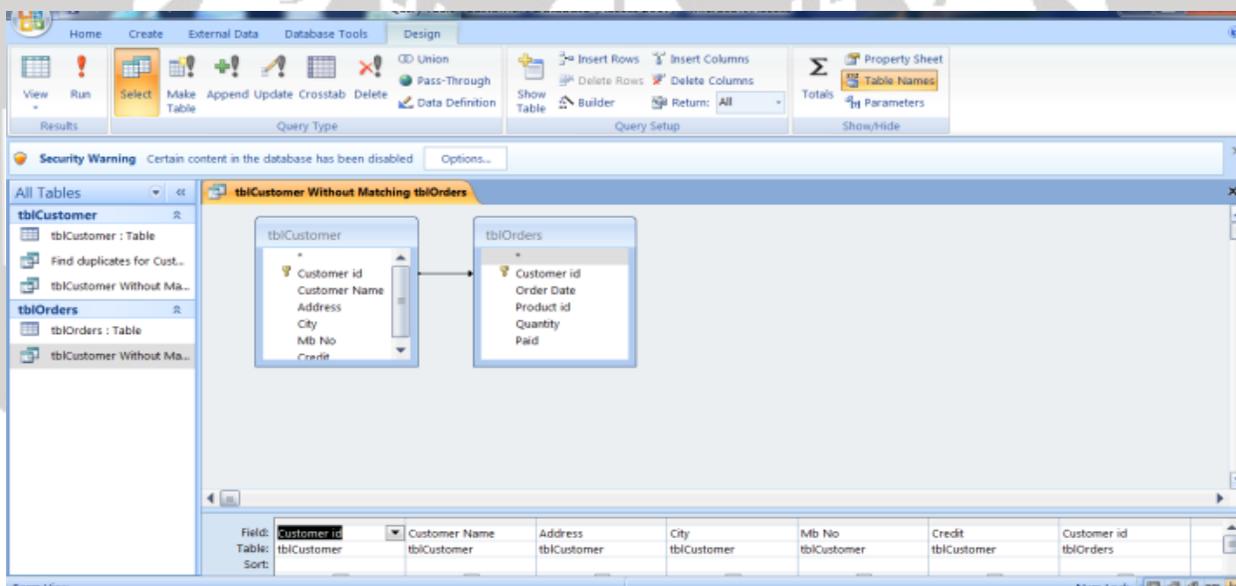
अंतिम स्क्रीन में क्वेरी के लिए नाम चुने और Finish पर क्लिक करें।



Finish पर click करते ही उन सभी customer के record, datasheet view में प्रदर्शित होंगे जिन्होंने order नहीं दिया अर्थात unmatched record::



Query पर Right click करके Design view में भी open कर सकते हैं।



इस विज़ार्ड ने tblCustomer और tblOrders के बीच एक बाहरी जुड़ाव बनाया है और Is Null criteria tblOrders से CustID में जोड़ा गया है। यह कुछ रिकॉर्ड को बाहर करने के लिए है। इस मामले में, यह वे ग्राहक हैं जिन्होंने आदेश दिया है, या जिनके पास tblOrders में संबंधित जानकारी है।

एमएस एक्सेसमें क्वेरी कैसे चलाएं

(How to Run Query In MS Access)

क्वेरी बनाने के बाद हमे उस क्वेरी को चला कर भी देखना पड़ता है कि अपने जो क्वेरी बनाई है वह Run हो रही है या नहीं तो क्वेरी को रन करने के लिए हम निम् प्रक्रिया अपनायेगे-
सबसे पहले आप अपनी क्वेरी को बना लीजिये।

उसके बाद Design Tab पर Run कमांड पर क्लिक करें।

एमएस एक्सेस में क्वेरी कैसे सेव करें

(How to Save Query In MS Access)

आप अपनी क्वेरी को Save भी कर सकते हैं क्वेरी को Save करने के लिए-
सबसे पहले Query Tab पर राइट-क्लिक करें।

फिर Save पर क्लिक करें।

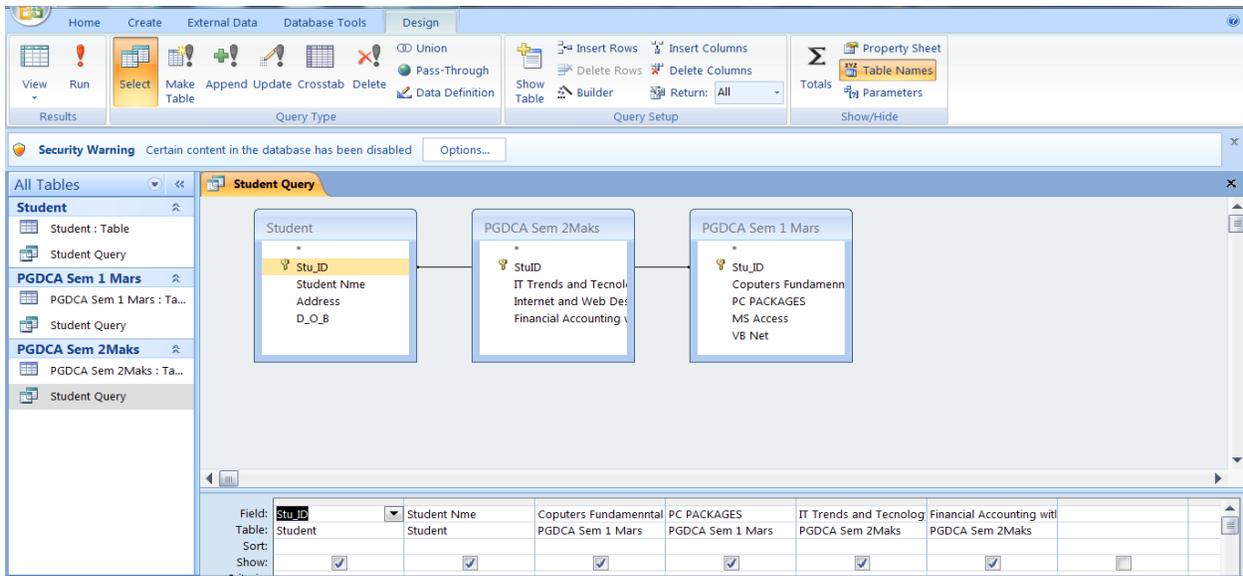
आपको एक Save as डायलॉग बॉक्स दिखाई देगा उसमे क्वेरी का नाम टाइप करें।

और ok पर क्लिक कर दें।

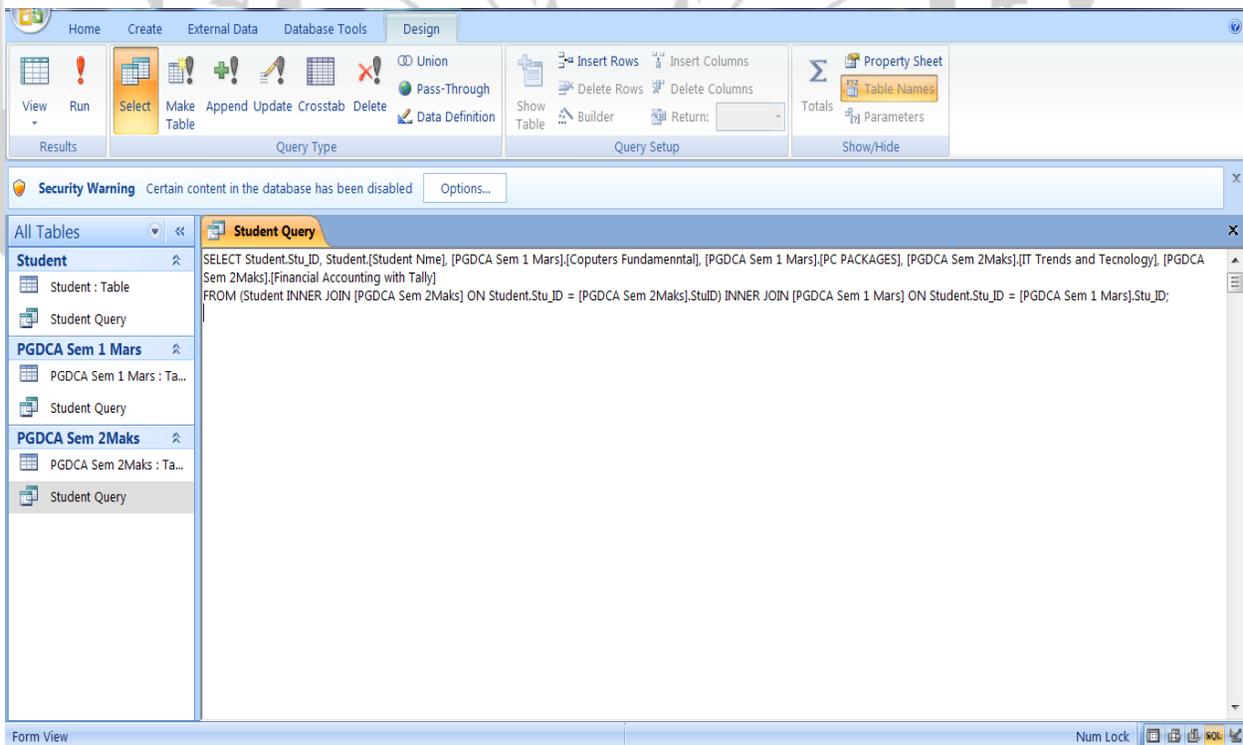
आपकी क्वेरी सेव हो जाएगी अब आप अपनी क्वेरी को कभी भी देख सकते हैं।

एमएस एक्सेस में क्वेरी के विभिन्न व्यूज़

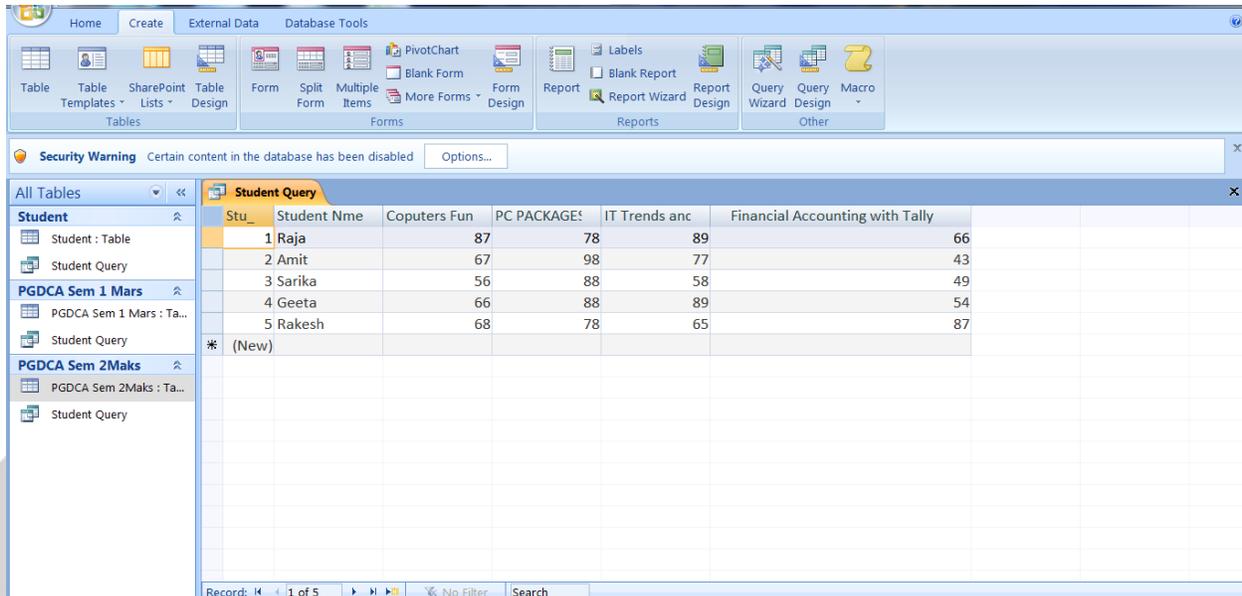
1. Design View
2. SQL View
3. Datasheet View
4. PivotTable View
5. PivotChart View



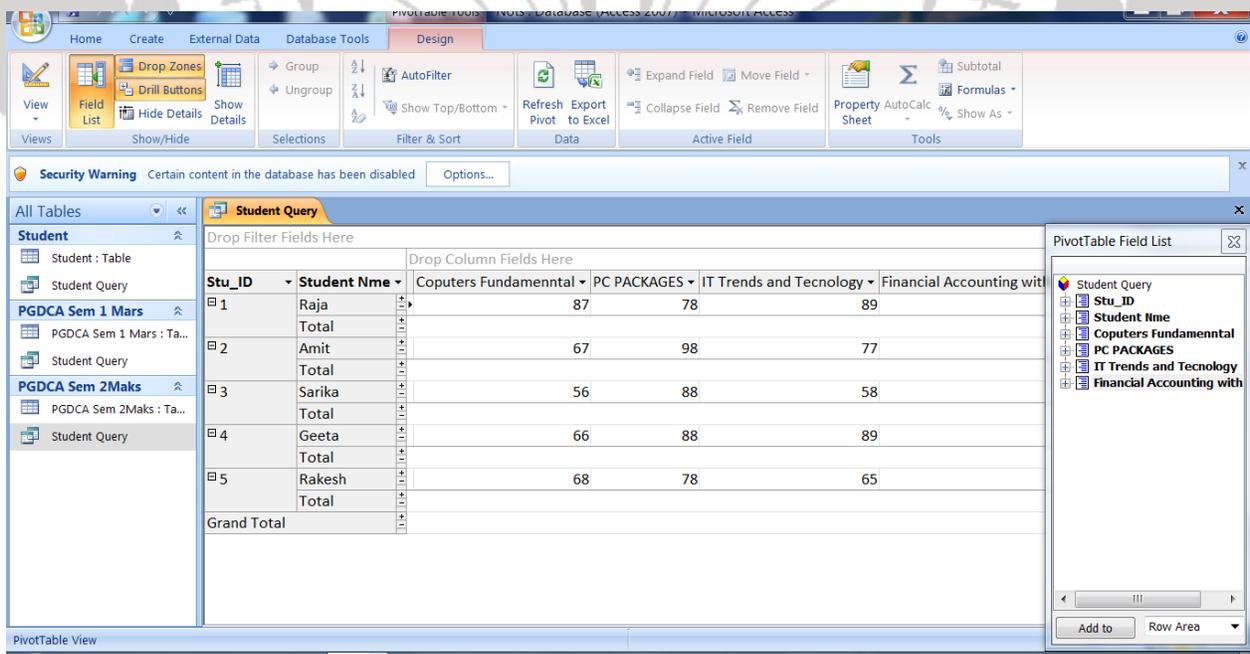
2.SQLView : इस view में हम अपनी बनाई क्वेरी के SQL Commands देख सकते हैं। हम इस view में SQL Commands के माध्यम से नई क्वेरी भी बना सकते हैं।



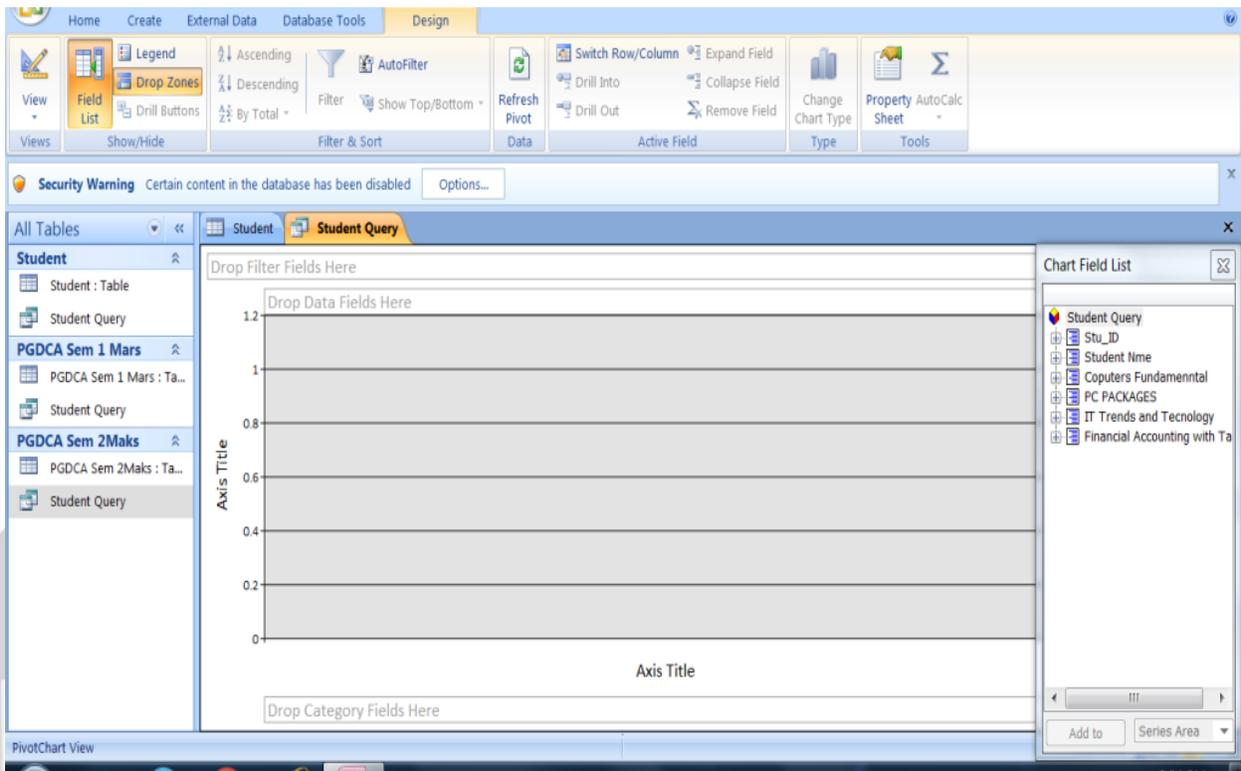
3. Datasheet View: इस view में क्वेरी का data, row and column format में display होता है, जहां हम fields को edit, add, delete या search भी कर सकते हैं।



4. PivotTable View: इस view में हम query का PivotTable बना सकते हैं। यह एक special business reporting table होता है।



4. PivotChart View: इस view में हम query का PivotChart बना सकते हैं। यह एक special business reporting Chart होता है।



डेटाबेस यूजिंग एमएस एक्सेस यूनिट - 2

एम.एस. एक्सेस में डाटा टाइप्स / DataTypes in MS Access

माइक्रोसॉफ्ट एक्सेस के अंदर डेटाबेस द्वारा काम होता है और वो डेटाबेस टेबल के रूप में होते हैं। टेबल के अंदर जितने भी फील्ड होते हैं उन सबकी प्रॉपर्टीज होती है और उन्हीं प्रॉपर्टीज की बदौलत फील्ड के characteristics और बर्ताव के बारे में पता चलता है। और फील्ड की सबसे महत्वपूर्ण प्रॉपर्टी जो होती है वो है डाटा टाइप्स। एक फील्ड का डाटा टाइप ये बताता है कि वो किस तरह के डाटा को स्टोर कर सकता है यानी रख सकता है।

- डाटा टाइप बताता है कि किसी फील्ड में किस प्रकार के डाटा रखे जा सकते हैं। यानी कि वो संख्या होगा या फिर टेक्स्ट या दोनों या कुछ और ये यहीं से निर्णय होता है।
- एक फील्ड एक ही टाइप के डाटा को स्टोर कर सकता है। एक फील्ड में एक से ज्यादा तरह के डाटा को स्टोर करने में एमएस एक्सेस अक्षम है। जैसे आप संख्या वाले डाटा फील्ड में मुद्रा नहीं डाल सकते।

डाटा टाइप्स का विवरण

डाटा के टाइप्स	विवरण	आकार
Short Text	टेक्स्ट या टेक्स्ट और संख्या का मिश्रण, ऐसे संख्या जिसकी गणना की जरूरत ना पड़े (जैसे फोन नम्बर्स)।	255 करैक्टर तक।
Long Text	लम्बे टेक्स्ट या टेक्स्ट और संख्या का मिश्रण।	63, 999 करैक्टर तक।
Number	गणित में प्रयोग होने वाले डाटा	1, 2, 4, या 8 बाइट (16 बाइट अगर Replication)

		ID सेट है)।
Date/Time	साल के लिए समय और दिनांक का मान	8 बाइट
Currency	मुद्रा का मान या संख्या जिसकी गणना की जाती है। दशमलव के बाद चार अंकों तक।	8 बाइट
AutoNumber	एक खास क्रमिक(1 से बढ़ाया हुआ) numberसंख्या या कोई रैंडम नम्बर जिसे एक्सेस ने डाला हो (जब भी टेबल में कोई नया रिकॉर्ड जोड़ा जाता है)	4 बाइट (16बाइट अगर Replication ID सेट हो)।
Yes/No	Yes/No, True/False, or On/Off	1 बिट

किसी फील्ड का डाटा टाइप उस फील्ड की कई महत्वपूर्ण क्वालिटी के बारे में बताता है; जैसे कि-

- फील्ड के साथ किस फॉर्मेट का प्रयोग किया जा सकता है।
 - फील्ड वैल्यू का अधिकतम आकार (Maximum size)।
 - एक्सप्रेशन में फील्ड का उपयोग कैसे किया जा सकता है।
 - फील्ड को इंडेक्स किया जा सकता है या नहीं।
 - किसी भी फील्ड का डाटा टाइप पहले से परिभाषित हो सकता है या फिर ये इसपर निर्भर करता है कि आप नया फील्ड कैसे बना रहे हैं। इसके लिए आप निम्नलिखित में से कोई एक आजमा कर देख सकते हैं:
- अगर आप किसी और टेबल के पहले से उपलब्ध डाटा फील्ड का प्रयोग करते हैं, तो डाटा टाइप पहले से ही टेम्पलेट में परिभाषित रहता है या फिर उसी टेबल में।
 - अगर आप किसी खाली कॉलम या क्षेत्र में डाटा डालते हैं, एमएस एक्सेस उस फील्ड को आपके द्वारा डाले गये डाटा के टाइप के आधार पर डाटा टाइप दे देता है। या फिर आप भी डाटा का प्रकार और फील्ड का फॉर्मेट तय कर सकते हैं।

- **Mobility Field** टैब के अंदर **Fields & Columns** समूह में जाएँ और **Add Fields** पर क्लिक करें। इसके बाद डाटा टाइप का एक लिस्ट आपके सामने खुल जाएगा जिसमे से एक आप चुन सकते हैं।

डाटा की प्रकृति के आधार पर डाटा कई प्रकार का होता है। एम एस एक्सेस में डाटा टाईप निम्न प्रकार के होते हैं।

Text - इस प्रकार के डाटा में (mathematical calculation) गणितीय गणनायें नहीं की जा सकती है। इसकी रेंज 0 To 255 अक्षर की होती है। अर्थात इस डाटा टाईप के फील्ड में अधिकतम 255 अक्षर लिखे जा सकते हैं। उदाहरण- name, city, address etc.

Number - इस प्रकार के डाटा में (mathematical calculation) गणितीय गणनायें की जा सकती है। इसके फील्ड में नंबर को स्टोर किया जाता है। इसको निम्न भागों में बाँटा गया है। जैसे Mark, Principle (मूल्य घन), Rate (दर), Time (समय) etc.

Date and Time - इसमें डेट एवं समय को स्टोर किया जाता है। इसके निम्न प्रकार के फार्मेट होते हैं।

Format	Use to display
Short Date	तारीख को एक छोटे प्रारूप में प्रदर्शित करें। आपकी क्षेत्रीय तिथि और समय सेटिंग्स पर निर्भर करता है। उदाहरण के लिए 31/10/2018
Medium Date	तारीख को मध्यम प्रारूप में प्रदर्शित करें। उदाहरण के लिए 31-अक्टूबर -18
Long Date	तिथि को एक लंबे प्रारूप में प्रदर्शित करें। आप पर निर्भर करता है क्षेत्रीय तारीख और समय सेटिंग्स। उदाहरण के लिए, बुधवार, 31 मार्च, 2018
Time am/pm	केवल 12 घंटे के प्रारूप का उपयोग करके समय प्रदर्शित करें जो क्षेत्रीय दिनांक और समय सेटिंग में परिवर्तनों का जवाब देगा।
Medium Time	Am / PM का समय प्रदर्शित करें।
Time 24hour	केवल 24 घंटे के प्रारूप का उपयोग करके समय प्रदर्शित करें जो क्षेत्रीय दिनांक और समय सेटिंग में परिवर्तनों का जवाब देगा

Currency - इस प्रकार के डाटा मे करंसी, पैसा को स्टोर किया जाता है। इसमे गणतीये गणनाये भी कर सकते है। प्रत्येक देश की अपनी करंसी होती है। एवं उसका एक चिन्ह होता है।

जैसे \$4500, \$520000 etc.

Memo - यह एक विशेष प्रकार का डाटा टाईप है। इसके टैक्ट को स्टोर करने की कोई सीमा नहीं होती है। इसका प्रयोग तब किया जाता है। जब किसी के बारे मे ज्यादा जानकारी स्टोर करनी हो।

OLE Object - इसका पूरा नाम Object linking embedding है। इसके किसी भी फाईल को लिंक कराया जा सकता है। जिस पर क्लिक करके खोला जा सकता है।OLE object मे लिंक करना.

Insert menu→ Object→ Insert object dialog box→ create from file→ select file→ ok

Logical - इसमे लाजिकल डाटा को स्टोर किया जाता है। जिसके केवल दो आपशन होते है। Yes/No

Yes/No:

पह आपको दो विकल्प में से किसी एक को चुनने की अनुमति प्रदान करता है जैसे - Yes/No, true/false, on/off आदि।

Data Type	Use to display
Check Box	check box.
Yes/No	Yes or No options
True/False	True or False options.
On/Off	On or Off options.

Auto number- इस डाटा टाईप से सीरियल नंबर अपने आप आते है। इसका प्रयोग सीरियल नंबर को स्टोर करने के लिये किया जाता है।

Number Data Type Field Properties :

- 1. Field size-** इसमें टेबल के फ़ील्ड की साइज को निर्धारित किया जाता है। कि उसमें कितना डाटा स्टोर कर सकते हैं, यह प्रॉपर्टी केवल text and number डाटा टाईप में होती है। टेक्स्ट में यह 0 to 255 character होती है। एवं number में दिये गये फॉर्मेट में से किसी एक को सिलेक्ट करते हैं।
- 2. Caption-** जब फार्म या रिपोर्ट को तैयार किया जाता है। तब इस प्रॉपर्टी का प्रयोग किया जाता है। इसमें जो कप्शन होता है। उसका प्रयोग अपने आप फार्म या रिपोर्ट में हो जाता है। जिससे वहां पर लेबिल लिखने की आवश्यकता नहीं होती है। यह प्रॉपर्टी लगभग सभी टाईप में होती है।
- 3. Format-** फ़ील्ड में डाटा को किस फॉर्मेट में लिखना है। इसका निर्धारण किया जाता है। यह एक महत्वपूर्ण फ़ील्ड प्रॉपर्टी होती है। अलग-अलग डाटा टाईप अलग-अलग फॉर्मेट में होता है। जिसमें से किसी एक फॉर्मेट को चुना जाता है। जिस फॉर्मेट में डाटा को इनपुट करना होता है।
- 4. Default Value-** इस फ़ील्ड प्रॉपर्टी में उस value को सेट करना होता है। जिस फ़ील्ड में डाटा default रूप से सेट करना है। इसे बाद में replace भी किया जा सकता है। जिससे टेबल में डाटा इनपुट करने में आसानी होती है।
- 5. Data Validation Rules-** यह टेबल में डाटा इनपुट करने के लिये एक महत्वपूर्ण property होती है। इससे डाटा को इनपुट करने के लिये नियम बनाये जाते हैं। जिससे गलत डाटा इनपुट ही नहीं हो पाता है। इसमें हम function, formula एवं गणतीय एवं लॉजिकल operation का भी प्रयोग कर सकते हैं। जिससे सही डाटा इनपुट होता है।
- 6. Input Mask-** इसमें फ़ील्ड में कितने अक्षर को इनपुट करना है। और किस प्रकार के अक्षर इनपुट करना है। इसको निर्धारित किया जाता है। इसके लिये अलग अलग चिन्हों का प्रयोग किया जाता है।

Field Properties in MS Access

MS Access में प्रत्येक टेबल फ़ील्ड से बनी होती है। किसी फ़ील्ड की प्रॉपर्टी उस फ़ील्ड में जोड़े गए डेटा की विशेषताओं और व्यवहार का वर्णन करते हैं। एक फ़ील्ड का डेटा प्रकार सबसे महत्वपूर्ण प्रॉपर्टी है क्योंकि यह निर्धारित करता है कि फ़ील्ड किस प्रकार का डेटा स्टोर कर सकता है। आप टेबल फ़ील्ड की प्रॉपर्टी सेट कर सकते हैं जिन्हें आप टेबल के डिज़ाइन व्यू में बनाते हैं। जब आप डिज़ाइन व्यू में टेबल खोलते हैं, तो आप फ़ील्ड का नाम देते हैं और उन्हें स्क्रीन डिज़ाइन ग्रिड नामक स्क्रीन के शीर्ष आधे हिस्से का उपयोग करके डेटा प्रकार असाइन

करते हैं। नीचे, “Field Property” अनुभाग में, आप उस फ़ील्ड के गुण सेट करते हैं जो वर्तमान में “General” और “Lookup” लेबल वाले दो टैबों पर टेबल डिज़ाइन ग्रिड में चयनित है।

Field Properties:- अन्य चीजों के साथ डेटा को संग्रहीत और प्रस्तुत करने के तरीके को प्रभावित करते हैं। आपके लिए उपलब्ध Field Properties की सूची उस फ़ील्ड के लिए चुने गए Data Type पर निर्भर करती है। कुछ Field Properties टेक्स्ट फ़ील्ड के लिए विशिष्ट हैं, और अन्य संख्या फ़ील्ड के लिए विशिष्ट हैं। Field Properties डिज़ाइन व्यू में पाए जा सकते हैं। जैसे ही आप प्रत्येक फ़ील्ड पर क्लिक करते हैं, आप उस फ़ील्ड के लिए Field Properties देखेंगे।

Property type	Description
Field Size	किसी टेक्स्ट फ़ील्ड को एक विशिष्ट संख्या तक सीमित करता है Character; एक विशिष्ट फ़ील्ड में एक संख्या फ़ील्ड को सीमित करता है।
Format	डेटाशीट व्यू में मूल्यों के तरीके को नियंत्रित करता है
Decimal places	केवल संख्या और मुद्रा फ़ील्ड के लिए उपलब्ध है, यह निर्धारित करता है कि फ़ील्ड में कितने दशमलव स्थान दिखाई देंगे; इस प्रॉपर्टी प्रकार का सामान्य प्रारूप का उपयोग कर संख्या फ़ील्ड पर कोई प्रभाव नहीं पड़ता है
Input Mask	फील्ड में दर्ज डेटा के लिए एक पैटर्न बनाता है (जैसे किसी टेलीफोन नंबर के भीतर ब्रैकेट या हाइफ़न जोड़ना)
Caption	फील्ड नाम के अलावा एक लेबल बनाता है; कैप्शन टेबल में और रूपों और रिपोर्टों पर दिखाई देगा
Default Value	वह मान निर्दिष्ट करता है जिसे आप चयनित फ़ील्ड में सभी नए रिकॉर्ड्स में दिखाना चाहते हैं
Validation Rule दर्ज किया गया है	निर्दिष्ट आवश्यकता को पूरा करने के लिए चयनित फ़ील्ड में डेटा
Validation Text	जब दर्ज किया गया डेटा सत्यापन नियम का उल्लंघन करता है तो प्रकट होने के लिए एक त्रुटि संदेश बनाता है
Required	आवश्यक निर्दिष्ट करता है कि फ़ील्ड को खाली नहीं छोड़ा जा सकता है

Allow Zero Length	यह निर्धारित करने के लिए कि टेक्स्ट के लिए कोई डेटा नहीं है, यह निर्धारित करने के लिए कि आप टेक्स्ट, मेमो या हाइपरलिंक फ़ील्ड में उद्धरण चिह्न ("") दर्ज कर सकते हैं या नहीं
Indexed	एक फ़ील्ड में डेटा की पुनर्प्राप्ति को गति देता है; सभी प्राथमिक कुंजी फ़ील्ड स्वचालित रूप से अनुक्रमित होते हैं
Unicode Compression:	इंगित करें कि क्या आप केवल सादा टेक्स्ट दर्ज किए जाने पर स्थान बचाने के लिए एक्सेस चाहते हैं
IME Mode	Input Method editor - सेटिंग केवल तभी लागू होती है जब आप पूर्वी एशियाई भाषाओं का उपयोग कर रहे हों
IME Sentence Mode	सेटिंग्स केवल जापानी भाषा पर लागू होती हैं।
Smart Tags	उस फ़ील्ड में अतिरिक्त कार्यक्षमता जोड़ता है जिसे आपको सामान्य रूप से करने के लिए एक और प्रोग्राम खोलना होगा।
Text Align	Left, Right, Center या Justify फ़ील्ड में डेटा को संरेखित करता है (डेटा सेल में फैला हुआ)

- 1. Field Size:-** संख्या Data Type के साथ काम करते समय, आपको Field Size की Property का सामना करना पड़ा था। यह Property सामान्य टेक्स्ट Data Type के लिए भी मौजूद है। यह आपको उस कॉलम में दर्ज डेटा पर अधिकतम आकार सीमा निर्धारित करने में सक्षम बनाता है। टेक्स्ट Data Type के लिए, आकार उस कॉलम में टेक्स्ट डेटा की लंबाई (वर्णों और रिक्त स्थानों की संख्या) को संदर्भित करता है। उदाहरण के लिए, कर्मचारी टेबल को देखते हुए, आप राज्य के लिए एक फ़ील्ड देखते हैं। आपकी फर्म आपको बताती है कि राज्यों के नाम उनके दो-अक्षर पदनाम का उपयोग करके दर्ज किए जाने चाहिए। यदि आप राज्य कॉलम के लिए फ़ील्ड आकार 2 पर सेट करते हैं, तो उपयोगकर्ता दो अक्षरों से अधिक लंबा टेक्स्ट टाइप करने में असमर्थ होगा। तो एक्सेस के साथ, आप न केवल किसी विशेष कॉलम में किसी निश्चित Data Type को मजबूर करने में सक्षम होते हैं, आप केवल उस व्यक्तिगत कॉलम को कस्टमाइज़ कर सकते हैं जो केवल आपके द्वारा निर्दिष्ट कठोर फॉर्मेट में डेटा स्वीकार कर सके।

2. **Format :-** यह प्रॉपर्टी आपको डाटा को सटीक तरीके से सेट करने में सक्षम बनाती है जिसमें एक्सेस टेबल्स में स्थित डेटा प्रदर्शित करता है या प्रिंट करता है। Field Size के साथ, चयन के लिए उपलब्ध फॉर्मेट उस कॉलम के Data Type पर निर्भर करता है। उदाहरण के लिए, एक Currency field के साथ, आप डेटा को ऐसे फॉर्म में प्रदर्शित कर सकते हैं जो डॉलर के चिह्न, यूरो चिह्न या कोई संकेत नहीं उपयोग करता है। डेटा को इन सेटिंग्स के साथ बदला नहीं जाएगा फॉर्मेट का एक और बहुत ही उपयोगी कार्य Date / Time डेटा प्रकारों के साथ है। चाहे आप लंबे फॉर्मेट या लघु फॉर्मेट में डेटा प्रदर्शित करना चाहते हैं, यह गुण आपको उस विकल्प को सेट करने में सक्षम बनाता है।
3. **Input Mask:** - यह सुविधा डेटा एंट्री स्थितियों में उपयोगी होती है। जहां फॉर्मेट नियंत्रित करता है कि डेटा कैसे प्रदर्शित होता है, इनपुट मास्क नियंत्रित करता है कि किसी विशेष फ़ील्ड में डेटा कैसे दर्ज किया जाता है। इनपुट मास्क निम्न डेटा प्रकारों के लिए उपलब्ध है: टेक्स्ट, संख्या, दिनांक/समय और मुद्रा। उदाहरण के लिए, यदि किसी उपयोगकर्ता को एक टेलीफोन नंबर दर्ज करने की आवश्यकता है, तो इनपुट मास्क अक्षर और संरचना बना सकता है जिसके साथ आप सभी परिचित हैं। उपयोगकर्ता प्रकार के रूप में, संख्या स्वचालित रूप से एक फ़ोन नंबर फॉर्मेट मानती है: (###) ### - #####।
4. **Decimal Places:-** संख्या फ़ील्ड में, आप दशमलव बिंदुओं के दाईं ओर Decimal Places की संख्या सेट कर सकते हैं जिन्हें रिकॉर्ड किया जाएगा। एक ऑटो सेटिंग है, जो स्थानों की सही संख्या निर्धारित करने के लिए फॉर्मेट सेटिंग को रोकती है। ऑटो के अलावा, आप Decimal Places की संख्या के लिए 0 से 15 का चयन कर सकते हैं।
5. **Default Value:-** एक महत्वपूर्ण डेटाबेस अवधारणा, Default Value डेटा एंट्री प्रक्रिया में समय बचाने में मदद कर सकता है। जब भी कोई नया रिकॉर्ड जोड़ा जाता है तो Default Value उस कॉलम में स्वचालित रूप से रखा जाता है। डिफॉल्ट को ओवरराइड किया जा सकता है, इसलिए यह आपके कॉलम को केवल उस विशेष मान के लिए मजबूर नहीं कर रहा है।
6. **Required:-** यह एक महत्वपूर्ण Property हैं, यह उचित Data Type का उपयोग करके उपयोगकर्ता को कुछ मूल्य दर्ज करने के लिए मजबूर करता है। यदि आवश्यक फ़ील्ड ठीक से भरा नहीं है तो एक नया रिकॉर्ड नहीं जोड़ा जाएगा। इनपुट मास्क के साथ, यह Property डेटा एंट्री प्रक्रिया पर अधिक नियंत्रण देने के लिए एक उत्कृष्ट तंत्र है।

Table in MS Access

Create a Table in MS Access (एम एस एक्सेस में टेबल का निर्माण करना)

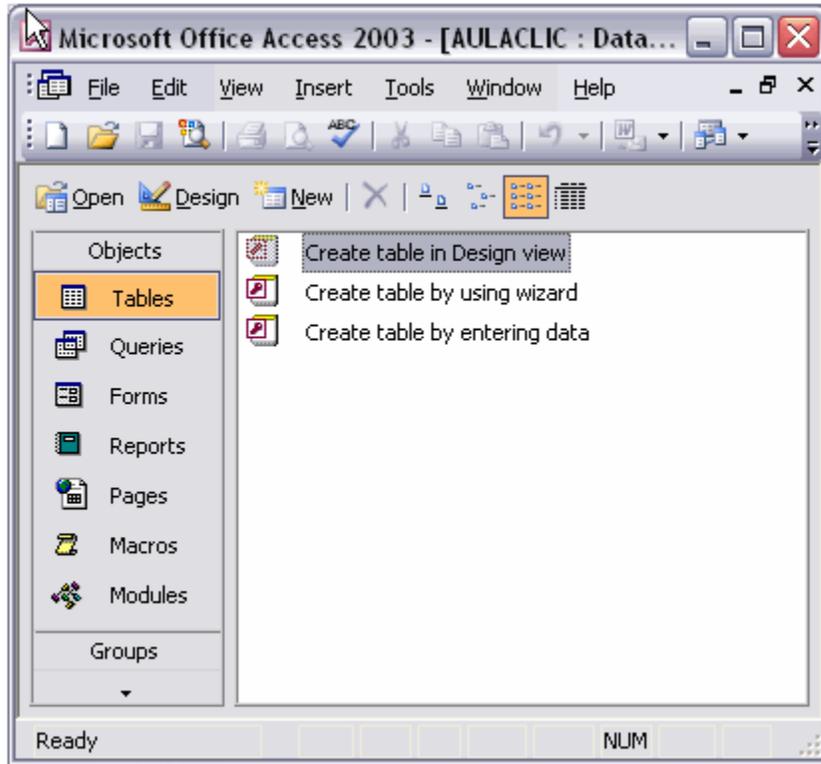
MS Access में डाटा को स्टोर करने के लिये टेबिल का निर्माण करना होता है। टेबिल डाटाबेस फाईल के अंदर होती है। एक डाटाबेस फाईल के अंदर एक से अधिक टेबिल हो सकती है। टेबिल का निर्माण रो एवं कॉलम से मिलकर होता है। फील्ड में डाटा टाईप को सेट किया जाता है।

- First step-

Go to file menu → new → click on blank database → insert file name → click on create button

- Second step-

Select table object इसमें तीन प्रकार से टेबिल को बनाया जा सकता है।

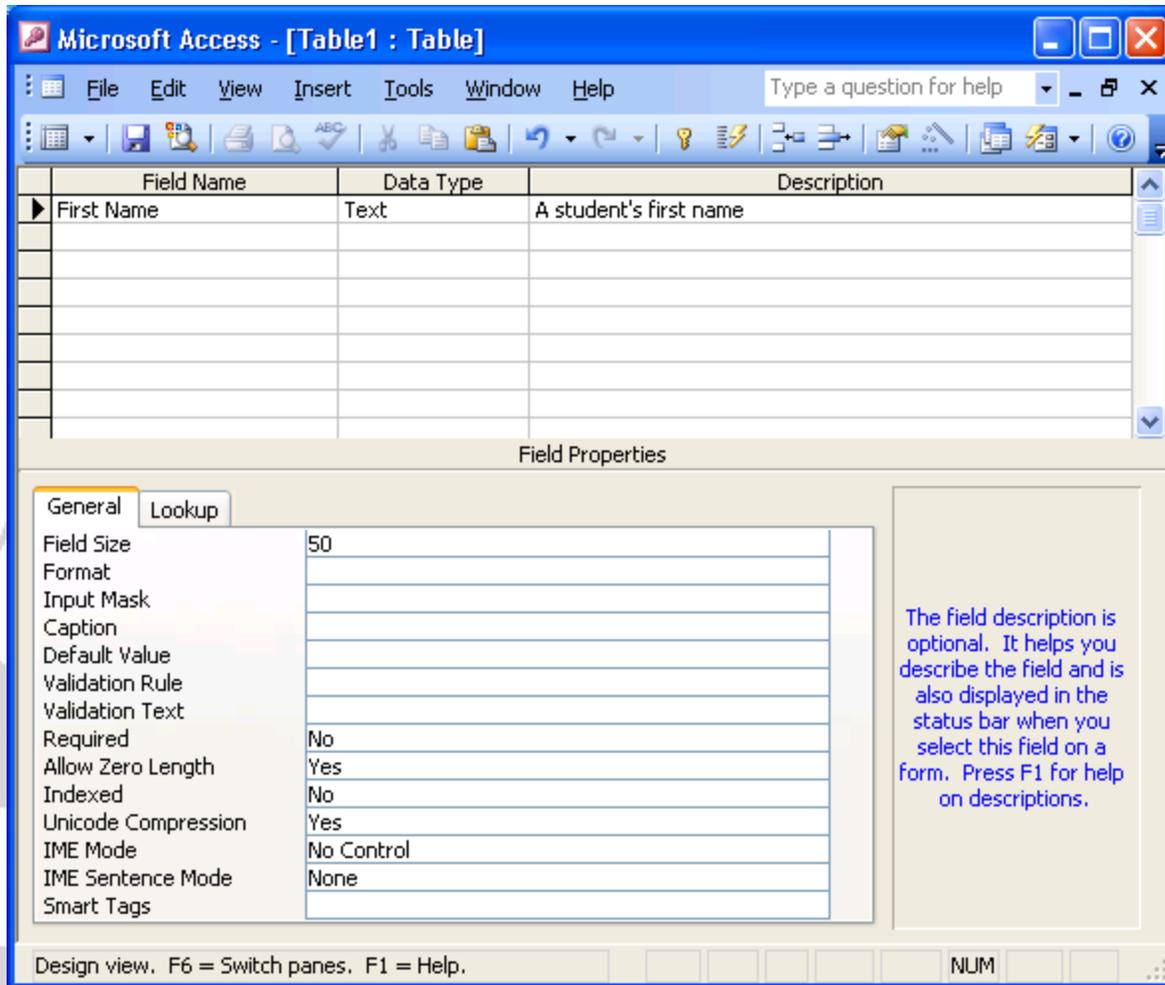


1. Create a Table in Design View:-

इसमें टेबल को यूजर के द्वारा डिजाइन किया जाता है। इसमें फील्ड का नाम देते हैं। और उसके डाटा टाईप को सिलेक्ट करते हैं और उस फील्ड की प्रॉपर्टी को सेट करते हैं।

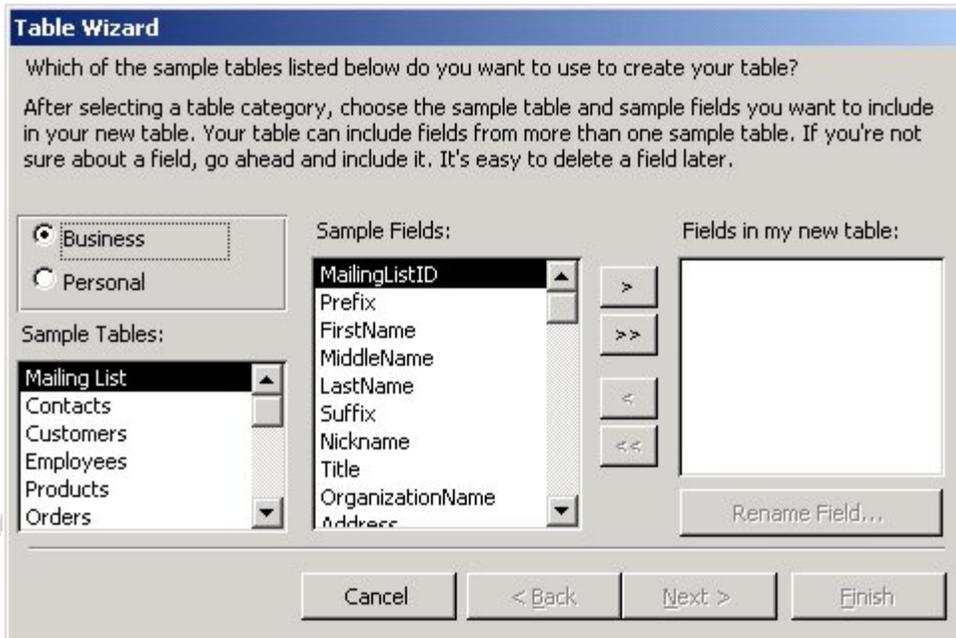
Design view में चार प्रकार से जा सकते हैं।

1. डिजाईन व्यू ऑप्शन पर डबल क्लिक करके Design view में जा सकते हैं।
2. सिलेक्ट करके Design Button पर क्लिक करके Design view में जा सकते हैं।
3. Right click on design view → design view में जा सकते हैं।
4. Open Option पर क्लिक करके Design view में जा सकते हैं।

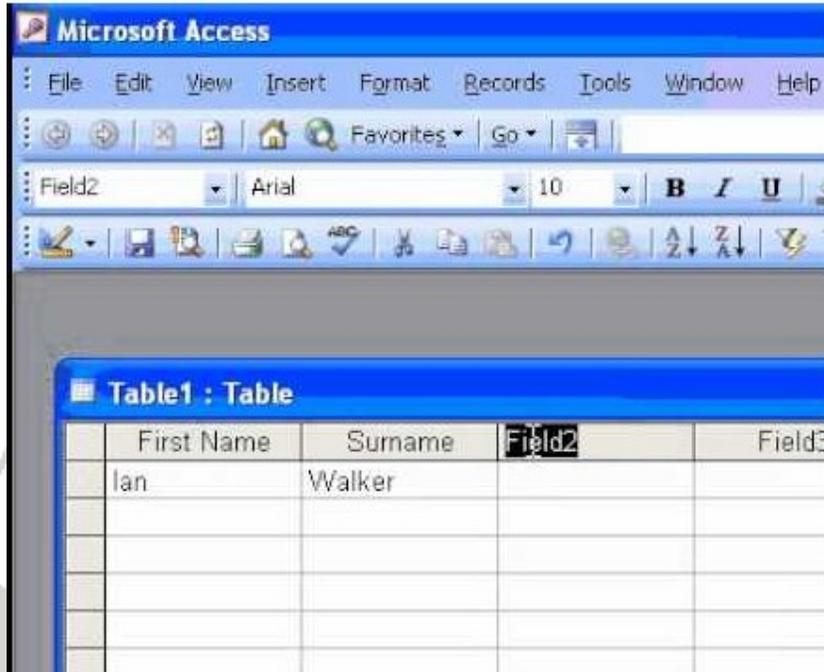


2. Create Table by using Wizard-

इससे टेबिल के structure को आसानी से कम समय मे तैयार किया जा सकता है। इसमे बने बनाये फील्ड होते है। जिसको सिलेक्ट करके न्यू टेबल मे transfer करते है। उसको रिनेम किया जा सकता है। आवश्यकता के अनुसार फील्ड को transfer करते है। और next button पर क्लिक कर सकते है। दूसरे डायलाॅग बाक्स मे टेबिल का नाम देते है। और primary key को सेट करना है। या नही इसके बाद next button पर क्लिक करते है। तीसरे डायलाॅग बाक्स मे यह निर्धारित किया जाता है। कि नई टेबल को किस व्यू मे देखना है।



3. Create Table by using entering Data- इसकी सहायता से सरलता से टेबिल को तैयार किया जाता है। इसमे डाटा सीट के फील्ड पर राईट क्लिक करके रिनेम किया जाता है। इसके बाद टेबिल को सेव किया जाता है।



MS Access में हम एक टेबल में निम्नलिखित कार्य कर सकते हैं

1. Add Record
2. Delete Record
3. Edit Record
4. Sort Record
5. Find and Replace
6. Filter & Select

1. Add Record in a Table :- पहले से बनी हुई Table में नया Record जोड़ने के लिए उस Table को Open करते हैं और रिकॉर्ड को जोड़ते हैं। टेबल में नया रिकॉर्ड जोड़ने के लिए निम्नलिखित स्टेप्स को फॉलो करेंगे |

- i. यदि Table में नए Record को जोड़ना हैं तो सबसे पहले उस Table को Select करते हैं,जिसमे रिकॉर्ड जोड़ना है |

- ii. अब Table Screen पर Display हो जाएगी। इसमें Last record पर Mouse के Curser को Point करते हैं तथा Mouse का Right Button Click करते हैं।
- iii. जिससे एक Popup Menu Display होता है इसमें Add Record Option पर Click करते हैं तो Curser Last Record के First Field में पहुँच जाता है अब इसमें हम नए Record को Add कर सकते हैं।

2. Delete a Record :- किसी Table में यदि किसी Record की आवश्यकता नहीं होती है अर्थात् जो अनावश्यक Record होते हैं उन्हें डिलीट कर दिया जाता है। अगर किसी रिकॉर्ड को टेबल से हटाना है तो निम्नलिखित स्टेप्स को फॉलो करेंगे।

- i. सबसे पहले जिस Record को Delete करना है उस उस Record को select करते हैं तथा Mouse का Right Button Click करते हैं
- ii. इसके Delete Record Option पर Click करते हैं ऐसा करने पर एक Message Display होता है।
- iii. यह Message हमसे Record को Delete करने के बारे में पूछता है। कि आप इसे Delete करने के लिए तैयार हैं या नहीं।
- iv. जब हम इस Message Box में Ok पर Click करते हैं तो Select किया हुआ Record Delete हो जाता है।

3. Edit Record :- यदि हम Table में हम किसी Record को संशोधित करना चाहते हैं तो इस कार्य को आसानी से कर सकते हैं।

Example :- यदि हमारे पास एक Employee नाम की Table है और उसमें हम किसी Employee के Record में Employee Name में Editing करना चाहते हैं तो निम्नलिखित स्टेप्स को फॉलो करेंगे।

- i. सबसे पहले वह टेबल ओपन करे जिसमें सुधार करना है।
- ii. इसके बाद Table में Record को Select करते हैं, इसके बाद उस Field को Select करते हैं जिस Field में सुधार करना है।
- iii. इसके बाद उस Field में संशोधन करके उस Table को Save कर देते हैं।

tblEmployees : Table		
lngEmpID	strEmpName	strEmpPassword
1	Graham	*****
2	Gavin	*****
3	Lynne	*****
4	David	*****
* [itoNumber]		
Record: 1 of 4		

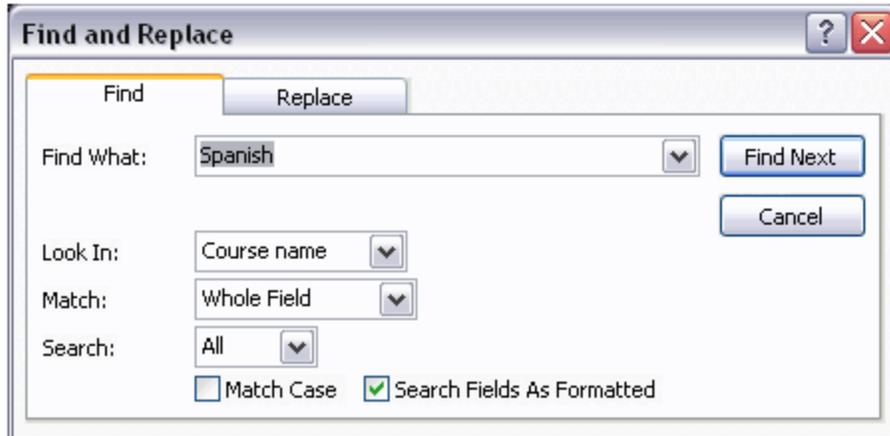
4. Sort Record :- हम Table में किसी भी Field को Sort कर सकते हैं। Sorting से तात्पर्य किसी Field को Ascending या Descending Order में arrange करना होता है। रिकॉर्ड को Ascending या Descending Order में arrange करने के लिए निम्नलिखित स्टेप्स को फॉलो करेंगे |

- i. किसी Field को Sorting करने के लिए उस Field को Select करते हैं |
- ii. इसके बाद Record Menu पर Click करते हैं |
- iii. इसके बाद Sort Sub Menu पर Click करते हैं |
- iv. यहाँ Sorting order (Ascending or Descending) को चुनते हैं। ऐसा करने पर Select की गयी Field उस Sorting order में व्यवस्थित हो जाती है जिसे आपने चुना है |

5. Find and Replace :- Table में किसी Particular Field में किसी विशिष्ट Value का पता लगाने के लिए Find Option का उपयोग करते हैं।

Example :- Employee Table में हजारों की संख्या में Record हैं उनमें से किसी Particular Employee के नाम का पता लगाना है तो इसके लिए Find Option का Use करते हैं। रिकॉर्ड को Find करने के लिए निम्नलिखित स्टेप्स को फॉलो करेंगे |

- i. Table में किसी Field में Particular Value पता करने के लिए सबसे पहले Table को Open करते हैं |
- ii. इसके बाद Edit Menu में Find Menu पर Click करते हैं |
- iii. इससे Find & Replace Window Display होगी |

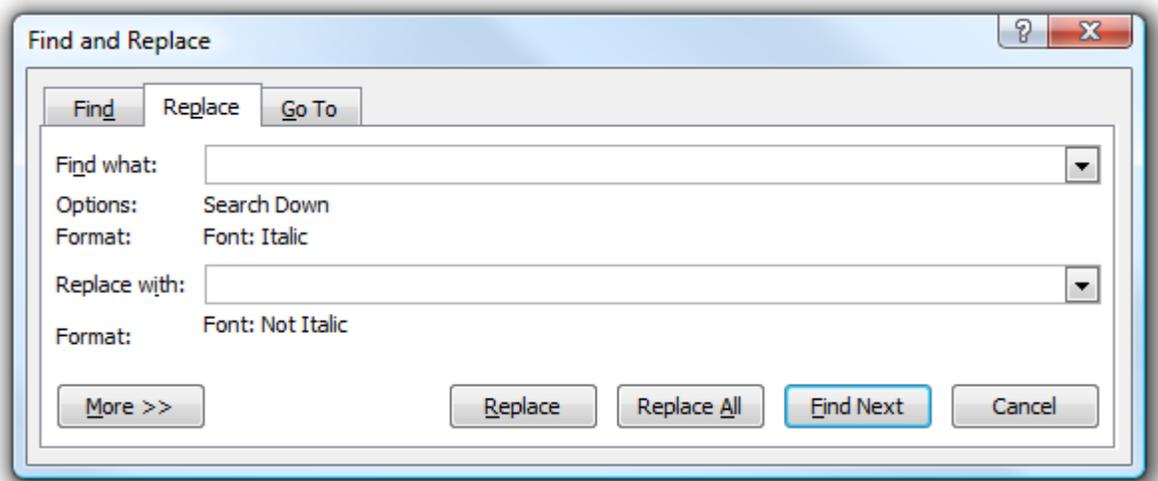


- iv. इसमें Find What Box में वह Value Type करते हैं जो Find करनी हैं। Curser उस Field की Value पर Highlight हो जाता हैं |
- v. यदि Same Text फिर से Find करना हो तो Find Next Button पर Click करते हैं अन्यथा Find & Replace Window को बंद कर देते हैं।

6. Replace :- किसी Record की Value को नयी value से बदलने के लिए Replace Option का Use करते हैं।

Example:- अगर हम Employee Table में Employee Name Raj की जगह Ram लिखना हैं तो इसके लिए पहले Raj को Find करते हैं तथा Ram के साथ Replace करते हैं तो Raj की जगह Ram हो जाता हैं। Replace करने के लिए निम्न स्टेप्स को फॉलो करेंगे |

- i. Table में किसी Field में कोई Value Replace करने के लिए Replace Sub Menu पर Click करते हैं |
- ii. Find & Replace Window Open हो जाती हैं इस Window में Find वाले Box में Source Name (जिस नाम को बदलना हैं) लिखते हैं। तथा Replace With Box में Destination Name (जो नाम लिखना हैं) लिखते हैं तथा Replace Button पर Click करते हैं तो पुराना नाम नए नाम के साथ Replace हो जाता हैं |



iii. यदि Find वाले Box में लिखे गए नाम की जगह पूरी Table में Replace With Box में लिखे गए नाम को लिखना है तो Replace All Button पर Click करते हैं।

7. Filter & Select :- Table में Record को Filter करने के लिए Filter/Select Option का Use करते हैं। फ़िल्टर का प्रयोग किसी कंडीशन के आधार पर रिकार्ड्स को देखने के लिए किया जाता है। Table में Filtering करने के लिए Table को Select कर Open करते हैं। इसके बाद Record Menu में Filter Sub Menu पर Click करते हैं। इसमें चार Option उपलब्ध होते हैं।

- Filter By Form
- Filter By Selection
- Filter Excluding Selection
- Advance Filter Sort

इसमें से हम अपनी आवश्यकता के अनुसार ऑप्शन का चुनाव करते हैं। Table में एक बार Filter ऑप्शन का प्रयोग करने के बाद उसे हटा भी सकते हैं।

8.Remove Filter :-Filter को हटाने के लिए Record Menu पर Click करते हैं इसमें Removed Filter Sub Menu पर Click करते हैं तो Filter Remove हो जाता है।

Rearrange Column or Row Size :

MS Access में टेबिल में Column और Row की साइज़ को आसानी से बदला जा सकता है, इसके स्टेप्स निम्नलिखित हैं।

Resize Row -किसी Table में Row की Size को Change करने के लिए Row की लंबाई को Gray Color की Line के द्वारा Mouse से Drag करके ऊपर या नीचे करके Change किया जा सकता है। Data Sheet पर एक Row की लंबाई या ऊँचाई बदल देने पर सभी Row की ऊँचाई बदल जाती है। तथा नया मान निर्धारित हो जाता है।

Resize Column- इसी प्रकार से Column की चौड़ाई भी Change की जा सकती है। Column की बीच की Line पर Double Click करने से Column की सबसे लंबी Value के अनुसार इसकी चौड़ाई निर्धारित हो जाती है। इसके आलावा कॉलम की चौड़ाई Menu Bar से Format Menu को Select करके भी Set की जा सकती है।

Freeze Columns in MS Access

अन्य फ़िल्ड को स्कॉल करते समय किसी मुख्य फ़िल्ड को दृश्यमान (Show) रखने के लिए फ़्रीज़ विकल्प का प्रयोग किया जाता है आप एक या अधिक फ़िल्ड (कॉलम) को फ़्रीज़ कर सकते हैं। आपके द्वारा फ़्रीज़ किए जाने वाले फ़िल्ड डेटाशीट पर बाईं ओर स्थित स्थान पर आ जाते हैं। कॉलम को freeze करने पर उसको एक स्थान से दूसरे स्थान पर move नहीं किया जा सकता है। जिस column को freeze करते हैं, वह टेबिल में पहले स्थान पर आ जाता है। इसके बाद उसको मूव नहीं किया जा सकता है। यदि फिर उसको मूव करना है। तो उसको unfreeze करना होता है।

Datasheet View में उस टेबल को ओपन करें जिसके कॉलम को आप फ़्रीज़ करना चाहते हैं।

- उस फ़िल्ड का चयन करें जिसे आप फ़्रीज़ करना चाहते हैं, कई फ़िल्ड का चयन करने के लिए, फ़िल्ड पर क्लिक करते समय SHIFT को दबाकर रखें।
- चयनित फ़िल्ड पर राइट-क्लिक करें, और फिर Freeze Fields पर क्लिक करें।



- यदि आप इसे Save करना चाहते हैं तो आप इसे Save भी कर सकते हैं।

यदि आप फ्रीज विकल्प को अपनी टेबल से हटाना चाहते हैं तो इसके लिए -

- सबसे पहले उस फ़ील्ड पर राइट-क्लिक करें जिसे आप Unfreeze करना चाहते हैं
- और फिर Unfreeze All Fields पर क्लिक करें।

प्राइमरी की क्या है? (What is Primary Key in MS Access)

प्राइमरी की को आप सरल शब्दों में किसी टेबल का ID कार्ड समझ सकते हैं। ये टेबल कि पहचान बताता है। किसी सोशल सिक्यूरिटी नम्बर या फिर किसी referenced नम्बर को प्राइमरी की के तौर पर प्रयोग किया जा सकता है।

ये यूनिक होता है। या यूँ कह लीजिये प्राइमरी की एक टेबल के बारे में बहुत कुछ बता देता है।

ये सम्बन्धित रिकॉर्ड रखे हुए एक से ज्यादा टेबल को आपस में जोड़ने में भी मदद करता है। टेबल के लिंकिंग में प्राइमरी कि बहुत काम आता है।

Field Name	Data Type
ID	AutoNumber
CommentDate	Date/Time
Comment	Long Text

एक ध्यान देने वाली बात ये है कि आप किसी डुप्लीकेट रिकॉर्ड को प्राइमरी की की तरह नहीं प्रयोग कर सकते हैं। ऐसे डाटा को निकालने या सॉर्ट करने में आसानी हो जाती है। एक अच्छे प्राइमरी की की पहचान है:

- ये खास तौर पर प्रत्येक रो कि पहचान करता है।
- ये कभी खाली या शून्य नहीं हो सकता। इसके अंदर हमेशा कोई न कोई मान होता है।
- इसके अंदर एक बार जो मान आ जाता है उसमे फिर कभी बदलाव शायद ही होता है। या तो वो कभी नहीं बदलता या फिर किसी खास केस में ही बदलता है।

प्राइमरी की को कैसे डिफाइन करें? (Define Primary Key in MS Access)

एक प्राइमरी की को डिफाइन या परिभाषित करने के लिए निम्नलिखित स्टेप्स को फॉलो करें:

- सबसे पहले तो Design View में जाएँ और एक टेबल को ओपन करें।



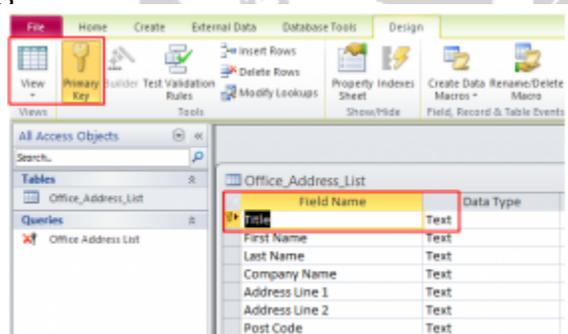
ये आपको उपर टैब में मिलेगा।

- टेबल हेडर पर राईट क्लिक करें।



ऐसा आप नेविगेशन पेन या फिर डाटाशीट व्यू में से किसी भी एक में कर सकते हैं।

- अब Option मेनू में जाकर डिजाइन व्यू को सेलेक्ट करें।
- आप जिस भी फील्ड को प्राइमरी की की तरह प्रयोग करना चाहते हैं उसे सेलेक्ट करें। इस बात का ध्यान रखें कि पहली प्राइमरी की हमेशा से डिफॉल्ट होती है।
- अब डिजाइन टैब के अंदर जाके Primary Key पर क्लिक करें जिसके बाद आपका चुना हुआ क्षेत्र परिमरी की बन जाएगा।



आप उपर चित्र में देख सकते हैं कैसे हमने इस टेबल का प्राइमरी की को परिभाषित किया है।

- अब View बटन पर क्लिक कर के वापस डेटाशीट व्यू में स्विच करें।

- अब आपके सामने एक Save As नामक डायलाग बॉक्स खुलेगा। वहां Yes पर क्लिक कर के टेबल को सेव करें।

प्राइमरी की को कैसे हटाएँ? (Remove Primary Key in MS Access)

प्राइमरी की को रिमूव करने के लिए निम्न प्रक्रिया का पालन करें:

- सबसे पहले ये जांच लें कि वो प्राइमरी की किसी टेबल के रिलेशनशिप में तो नहीं है। अगर ऐसा है तो सबसे पहले उस रिलेशनशिप को डिलीट करें तभी इसके बाद आप प्राइमरी की को हटा पाएँगे।
- जब आपने रिलेशनशिप को डिलीट कर दिया तो उसके बाद नेविगेशन पेन में उस टेबल पर क्लिक करें जिसके प्राइमरी की को आप हटाना चाहते हैं। अगर आपको नेविगेशन पेन नहीं दिख रहा तो आप अपने कीपैड से F11 दबा सकते हैं।
- अब Design View पर क्लिक करें।
- अब करंट प्राइमरी की के लिए रो सिलेक्टर पर क्लिक करें। अगर प्राइमरी की एक ही फील्ड के लिए है तो उस फील्ड के रो सिलेक्टर पर क्लिक करें।
- अगर वो प्राइमरी की बहुत सारे फील्ड का है यानि एक से ज्यादा फील्ड का है टी किसी भी फील्ड के रो सिलेक्टर पर क्लिक करें।
- अब Design Tab के Tools समूह में Primary Key को सेलेक्ट करें। इसके बाद आप जिस प्राइमरी की को हटाना चाहते हैं वो डिलीट हो जाएगा।

डेटाबेस यूजिंग एमएस एक्सेस

1. डाटा मॉडल

आधारभूत डेटाबेस की संरचना को समझना डेटा मॉडल है। डेटा, डेटा रिलेशनशिप, डेटा शब्दार्थ (data semantics), और स्थिरता बाधाओं (consistency constraints) का वर्णन करने के लिए वैचारिक उपकरणों का एक संग्रह है।

एक डेटा मॉडल की अवधारणा को समझाने के लिए, हम दो डेटा मॉडल को विस्तार से समझेंगे : इकाई-संबंध (Entity-Relationship) मॉडल और संबंधपरक मॉडल (Relational Model)। दोनों तार्किक स्तर पर एक डेटाबेस के डिजाइन का वर्णन करने का तरीका प्रदान करते हैं।

1.1 इकाई-संबंध मॉडल / The Entity-Relationship Model

इकाई-संबंध (ई-आर) डेटा मॉडल एक वास्तविक दुनिया की धारणा पर आधारित है जिसमें मूल वस्तुओं का एक संग्रह होता है, जिसे इकाई (एंटिटी) कहा जाता है, और इन वस्तुओं के बीच संबंधों का एक संग्रह होता है। एक एंटिटी वास्तविक दुनिया में एक "बात" या "वस्तु" है जो अन्य वस्तुओं से अलग है। उदाहरण के लिए, प्रत्येक व्यक्ति एक एंटिटी है, और बैंक खातों को एंटिटी के रूप में माना जा सकता है।

किसी डेटाबेस में एंटिटीज़ को विशेषताओं (**attributes**) के एक सेट द्वारा वर्णित किया गया है। उदाहरण के लिए, एट्रिब्यूट खाता-संख्या और शेषराशि एक बैंक में किसी विशेष खाते का वर्णन कर सकते हैं, और खाता एंटिटीसेट के एट्रिब्यूट हो सकते हैं। इसी तरह, एट्रिब्यूट ग्राहक-नाम, ग्राहक-सड़क का पता और ग्राहक-शहर एक ग्राहक एंटिटी का वर्णन कर सकते हैं।

ग्राहकों को विशिष्ट रूप से पहचानने के लिए एक अतिरिक्त एट्रिब्यूट ग्राहक-आईडी का उपयोग किया जाता है (क्योंकि एक ही नाम, सड़क का पता और शहर वाले दो ग्राहक होना संभव हो सकता है)।

एक अद्वितीय ग्राहक पहचानकर्ता, प्रत्येक ग्राहक को सौंपा जाना चाहिए। जैसे भारत में, कई उद्यम/संस्थाएं ग्राहक की पहचानकर्ता के रूप में एक व्यक्ति की आधार संख्या (uidai, भारत सरकार प्रत्येक व्यक्ति को प्रदान करता है) का उपयोग करते हैं।

एक संबंध (Relationship) कई एंटिटीयों के बीच एक संबंध/जुड़ाव है। उदाहरण के लिए, एक जमाकर्ता रिलेशनशिप प्रत्येक खाते (जो उसके पास है) के साथ एक ग्राहक को जोड़ता है।

एक ही प्रकार की सभी एंटिटीयों (संस्थाओं) के सेट और एक ही प्रकार के सभी रिलेशनशिप (संबंधों) के सेट को क्रमशः एक एंटिटी सेट (इकाई सेट) और रिलेशनशिप सेट (संबंध सेट) कहा जाता है।

एक डेटाबेस के समग्र तार्किक संरचना (Logical Structure) (स्कीमा / Schema) को ई-आर आरेख (E-R Diagram) द्वारा रेखांकन द्वारा (Graphically) व्यक्त किया जा सकता है।

1.2 संबंधपरक मॉडल (Relational Model)

संबंधपरक मॉडल डेटा और उन डेटा के बीच संबंधों (Relationships) को दर्शाने के लिए तालिकाओं (Tables) के संग्रह का उपयोग करता है। प्रत्येक तालिका (Table) में कई स्तंभ (Columns) होते हैं और प्रत्येक स्तंभ (Column) का एक अद्वितीय नाम होता है।

डेटा को एक ऐसे रिलेशन में व्यवस्थित किया जाता है जिसे दो आयामी तालिका (two dimensional table) में दर्शाया जाता है। डेटा को टपल्स (tuples) के रूप में टेबल में डाला जाता है (टपल्स मतलब पंक्तियां / rows)। एक टपल का गठन एक या एक से अधिक ऐट्रिब्यूट्स द्वारा किया जाता है। तालिका में किसी भी संख्या में टपल्स हो सकते हैं, लेकिन सभी टपल में अलग-अलग मानों (values) के साथ निश्चित और समान विशेषताएं (ऐट्रिब्यूट्स) होती हैं। संबंधपरक मॉडल (Relational Model) को डेटाबेस में इस प्रकार लागू किया जाता है, जहां एक संबंध (Relation) एक तालिका (Table) द्वारा दर्शाया जाता है। एक टपल (tuple) को एक पंक्ति (Row) द्वारा दर्शाया जाता है। टेबल के एक कॉलम द्वारा एक ऐट्रिब्यूट को दर्शाया जाता है। ऐट्रिब्यूट का नाम ही कॉलम का नाम होता है जैसे नाम, शहर आदि। ऐट्रिब्यूट की वैल्यू ही किसी रो में कॉलम के लिए वैल्यू होती है। नियमों और शर्तों (Constraints) को टेबल पर लागू किया जाता है और तार्किक स्कीमा (Logical Schema) बनाते हैं। टेबल से किसी विशेष पंक्ति / टपल के चयन को सुविधाजनक बनाने के लिए कॉलम के नामों (मतलब ऐट्रिब्यूट्स) का उपयोग किया जाता है। और पंक्तियों (rows) के चयन के लिए कुछ फ़िल्ड्स (fields) को विशिष्ट रूप से अनुक्रमित (indexes) के रूप में उपयोग करने के लिए परिभाषित किया जाता है। इससे आवश्यक डेटा खोजने में मदद मिलती है। सभी संबंधपरक बीजगणित संचालन (Relational Algebra Operations), जैसे कि चयन (Select), अंतर्ग्रहण (Intersection), उत्पाद (Product), संघ (Union), अंतर (Difference), परियोजना (Project), सम्मिलित (Join), विभाजन (Division), विलय (Merge) आदि भी संबंधित डेटाबेस मॉडल (Relational Database Model) पर किए जा सकते हैं। रिलेशनल डेटाबेस मॉडल पर

संचालन (Operations) विभिन्न Conditional expressions, विभिन्न Key Attributes, पूर्व-परिभाषित Constraints आदि की सहायता से किया जाता है।

2. इकाई-संबंध मॉडल (Entity Relationship Model) :

एंटिटी-रिलेशनशिप (ईआर) मॉडल को मूल रूप से 1976 में पीटर द्वारा नेटवर्क और रिलेशनल डेटाबेस विचारों को एकजुट करने के लिए प्रस्तावित किया गया था। सीधे तौर पर कहा गया है कि ई-आर मॉडल एक वैचारिक डेटा मॉडल है, जो वास्तविक दुनिया को संस्थाओं (एंटिटी) और रिश्तों (रिलेशन) के रूप में देखता है। इस मॉडल का एक मूल घटक इकाई-संबंध चित्र (एंटिटी-रिलेशनशिप डायग्राम) है, जो डेटा ऑब्जेक्ट्स को चित्र रूप से प्रस्तुत करने के लिए उपयोग किया जाता है। चूंकि चेन ने उनके पेपर में लिखा था कि यह मॉडल बढ़ाया गया है और आज इसका इस्तेमाल आमतौर पर डेटाबेस डिज़ाइन के लिए किया जाता है। एक डेटाबेस डिज़ाइनर के लिए, ER मॉडल की उपयोगिता है :

यह संबंधपरक मॉडल (Relational Model) को अच्छी तरह से मैप करता है। ई-आर मॉडल में उपयोग किए गए निर्माण (Constructs) आसानी से रिलेशनल टेबल में बदल सकते हैं।

न्यूनतम प्रशिक्षण के साथ इसे समझना सरल और आसान है। इसलिए, इस मॉडल का उपयोग डेटाबेस डिज़ाइनर द्वारा डिज़ाइन को अंतिम उपयोगकर्ता (End User) तक पहुंचाने के लिए किया जा सकता है।

इसके अलावा, एक विशिष्ट डेटाबेस प्रबंधन सॉफ्टवेयर (Specific Database Management Software) में डेटा मॉडल को लागू करने के लिए डेटाबेस डेवलपर द्वारा डिज़ाइन योजना (Design Plan) के रूप में इस मॉडल का उपयोग किया जा सकता है।

2.1 ई-आर मॉडलिंग के बुनियादी निर्माण / Basic Constructs of E-R Modeling :

ई-आर मॉडल, एंटिटीयों के निर्माण और एंटिटीयों के बीच सम्बन्ध के रूप में वास्तविक दुनिया को देखता है।

2.1.1 संस्थाओं / Entities (एंटिटीस) :

एंटिटीज प्रमुख डेटा ऑब्जेक्ट हैं जिनके बारे में जानकारी एकत्र की जानी है। एंटिटीज आमतौर पर पहचानने योग्य अवधारणाएँ (recognizable concepts) होती हैं, या तो ठोस (concrete) या अमूर्त (abstract), जैसे व्यक्ति, स्थान, चीज़ें, या घटनाएँ जिनकी डेटाबेस में

प्रासंगिकता होती है। एंटीटीज के कुछ उदाहरण कर्मचारी, परियोजनाएं, बिल हैं। एक एंटीटी, संबंधपरक मॉडल (Relational Model) में एक तालिका (Table) के अनुरूप है।

एंटीटीस को स्वतंत्र या निर्भर के रूप में वर्गीकृत किया जाता है (इनके लिए क्रमशः मजबूत (Strong) और कमजोर (Weak) शब्दों का प्रयोग होता है)। एक स्वतंत्र एंटीटी वह है जो पहचान के लिए दूसरे पर निर्भर नहीं होती है। एक निर्भर / आश्रित एंटीटी वह है जो पहचान के लिए दूसरे पर निर्भर होती है।

एक इकाई उपस्थिति (Entity occurrence) (जिसे एक उदाहरण (instance) भी कहा जाता है) एक एंटीटी की एक व्यक्तिगत उपस्थिति है। एक उपस्थिति संबंधपरक टेबल (Relational Table) में एक पंक्ति (Row) के अनुरूप है।

विशेष एंटीटी के प्रकार (Special Entity Types) :

एसोसिएटिव एंटीटीस (Associative Entities), दो या दो से अधिक एंटीटीस को जोड़ने के लिए उपयोग की जाने वाली एंटीटीस हैं, जिससे कई-से-कई संबंधों (Many-to-Many Relationship) बनाये जा सकें। इनको इंटरसेक्शन एंटीटीस भी कहा जाता है।

उपप्रकार इकाइयाँ (Subtypes Entities) का उपयोग सामान्यीकरण पदानुक्रम (Generalization Hierarchies) में अपने पैरेंट एंटीटी (Parent Entity) के उदाहरणों (instances) के सबसेट (subset) का दर्शाने के लिए किया जाता है। पैरेंट एंटीटी (Parent Entity) को सुपरपाइप (Supertype) कहा जाता है। लेकिन इस सबसेट के उदाहरणों (instances) में ऐसे गुण या संबंध होते हैं, जो केवल सबसेट पर लागू होते हैं।

एसोसिएटिव एंटीटीस (Associative Entities) और सामान्यीकरण पदानुक्रमों (Generalization Hierarchies) के बारे में नीचे और अधिक विस्तार से चर्चा की गई है।

2.1.2 रिश्ता (रिलेशनशिप / Relationships) :

एक रिश्ता (Relationship) दो या दो से अधिक एंटीटीस के बीच संबंध (association) को दर्शाता है। एक रिश्ते (Relationship) का एक उदाहरण निम्नानुसार है :

कर्मचारियों को परियोजनाओं (Projects) पर लगाया गया है।

परियोजनाओं के उपकेंद्र (subtasks) हैं।

विभाग (Departments) एक या अधिक परियोजनाओं का प्रबंधन करते हैं।

डिग्री, कनेक्टिविटी, कार्डिनैलिटी और अस्तित्व (Degree, Connectivity, Cardinality and existence) के संदर्भ में रिश्तों (Relationship) को वर्गीकृत किया जाता है ।

इन अवधारणाओं पर नीचे चर्चा की जाएगी।

2.1.3 विशेषताएँ / Attributes :

विशेषताएँ उस इकाई (Entity) का वर्णन करती हैं, जिससे वे संबद्ध हैं । एक विशेषता (Attribute) का एक विशेष उदाहरण (instance), एक मान (value) है। उदाहरण के लिए, "किशोर" एट्रिब्यूट "नाम" का एक मान है। एक एट्रिब्यूट का डोमेन (Domain) उन सभी संभावित मानों (Values) का संग्रह है, जो एक एट्रिब्यूट की हो सकती है। Name का डोमेन एक वर्ण स्ट्रिंग (Character String) है ।

एट्रिब्यूट्स को पहचानकर्ता (Identifiers) या वर्णनकर्ता (Descriptors) के रूप में वर्गीकृत किया जा सकता है । पहचानकर्ता (Identifiers), जिन्हें आमतौर पर कुंजी (Keys) कहा जाता है, विशिष्ट रूप से (uniquely) एक एंटीटी के एक उदाहरण (आवृत्ति / instance) की पहचान करते हैं। एक वर्णनकर्ता (Descriptors) एक एंटीटी उदाहरण (instance) की एक गैर-विशिष्ट (non-unique) विशेषता का वर्णन करता है।

2.2 रिश्तों का वर्गीकरण (Classifying Relationships) :

रिश्तों (Relationships) को उनकी डिग्री (Degree), कनेक्टिविटी (Connectivity), कार्डिनैलिटी (Cardinality), दिशा (Direction), प्रकार (Type) और अस्तित्व (Existence) द्वारा वर्गीकृत किया जाता है। सभी मॉडलिंग पद्धतियां इन सभी वर्गीकरणों का उपयोग नहीं करती हैं।

2.2.1 रिलेशनशिप की डिग्री (Degree of a Relationship) :

एक रिश्ते (रिलेशनशिप) की डिग्री रिलेशनशिप से जुड़ी एंटीटीस की संख्या है। n-ary रिलेशनशिप, डिग्री n के लिए सामान्य रूप है। बाइनरी (Binary) और टर्नरी (Ternary) विशेष रिलेशनशिप हैं, जहां डिग्री क्रमशः 2, और 3 है।

बाइनरी रिलेशनशिप, दो एंटीटीस के बीच संबंध (association) वास्तविक दुनिया में सबसे आम प्रकार है। एक पुनरावर्ती (recursive) द्विआधारी रिश्ता (बाइनरी रिलेशनशिप) तब होता है जब एक एंटीटी खुद से संबंधित होती है। एक उदाहरण हो सकता है जैसे "कुछ

कर्मचारी दूसरे कर्मचारियों से शादी करते हैं"। यहाँ एंटीटी कर्मचारी का रिलेशनशिप खुद एंटीटी कर्मचारी से ही है ।

एक त्रिगुट रिश्ता (टरनरी रिलेशनशिप) में तीन एंटीटीस शामिल होते हैं और इसका उपयोग तब किया जाता है जब एक बाइनरी रिलेशनशिप अपर्याप्त होता है। कई मॉडलिंग पद्धतियां केवल द्विआधारी रिश्तों (बाइनरी रिलेशनशिप) को मानते हैं। टर्नरी या n-ary रिश्ते, दो या अधिक बाइनरी रिलेशनशिप में विघटित किये जाते हैं।

2.2.2 कनेक्टिविटी और कार्डिनैलिटी (Connectivity and Cardinality):

एक रिश्ते की कनेक्टिविटी (Connectivity of a Relationship), रिश्ते में संबंधित एंटीटी उदाहरणों (instances) की मैपिंग का वर्णन करती है। कनेक्टिविटी का मान (value) "एक" या "कई" होता है।

एक रिश्ते की कार्डिनैलिटी (Cardinality of a Relationship), प्रत्येक दो एंटीटीस के लिए संबंधित घटनाओं (occurrences) की वास्तविक संख्या होती है।

रिश्तों (Relationship) में मुख्य प्रकार की कनेक्टिविटी (Connectivity) हैं : एक-से-एक, एक-से-कई, और कई-से-कई (one-to-one, one-to-many, many-to-many Relationship)।

2.2.3 रिश्ते की दिशा (Direction of Relationship) :

एक रिश्ते (Relationship) की दिशा (Direction) एक द्विआधारी रिश्ते (Binary Relationship) की उत्पत्ति एंटीटी (Originating Entity) को इंगित करती है। जिस एंटीटी से एक रिश्ता (रिलेशनशिप) उत्पन्न होता है वह पैरेंट एंटीटी (Parent Entity) होती है तथा वह एंटीटी जहां रिलेशनशिप समाप्त हो जाता है , वह चाइल्ड एंटीटी (Child Entity) होती है।

किसी रिलेशनशिप की दिशा (Direction) उसकी कनेक्टिविटी (Connectivity) से तय होती है। एक-से-एक रिश्ते (one-to-one Relationship) में दिशा (Direction), स्वतंत्र एंटीटी (Independent Entity) से एक आश्रित एंटीटी (Dependent Entity) की तरफ होती है। यदि दोनों एंटीटीस स्वतंत्र हैं, तो दिशा कोई भी हो सकती है। एक-से-कई रिश्तों (one-to-many Relationship) में, एक बार आने वाली एंटीटी, पैरेंट एंटीटी होती है। कई-से-कई रिश्तों (many-to-many Relationship) की दिशा कोई भी हो सकती है।

2.3 संबंधपरक कुंजी / (Relational Keys) :

रिश्तों (Relations) में दो तरह की कुंजी (Keys) होती हैं। पहली कुंजी पहचान करने वाली कुंजी होती है। इसमें प्राथमिक कुंजी (Primary Key / प्राइमरी की) मुख्य अवधारणा (Concept) है। जबकि दो अन्य कुंजी - सुपर कुंजी (Super Key / सुपर की) और उम्मीदवार कुंजी (Candidate Key / कैंडिडेट की) - संबंधित अवधारणाएं (Related Concepts) हैं। दूसरी तरह की कुंजी विदेशी कुंजी (Foreign Key / फॉरेन की) है।

2.3.1 पहचान की / Identity Keys :

(i) सुपर की / (Super Key)

एक "सुपर की" उन ऐट्रिब्यूट्स का एक समूह है, जिनके मान (values) का उपयोग किसी संबंध (relation) के भीतर एक टपल (tuple) को विशिष्ट रूप से पहचानने के लिए (uniquely identify) किया जा सकता है। एक संबंध (relation) में एक से अधिक सुपर की हो सकती हैं, लेकिन इसमें हमेशा कम से कम एक सुपर की ऐसी होती है जिसमें सभी ऐट्रिब्यूट्स का एक सेट हो जो संबंध (relation) बनाते हैं।

(ii) उम्मीदवार कुंजी (Candidate Key / कैंडिडेट की) :

एक कैंडिडेट की ऐसी एक सुपर कुंजी है जो न्यूनतम है। अर्थात्, इसका कोई उचित उपसमूह (Proper subset) नहीं है, जो स्वयं एक सुपर की हो। एक रिलेशन में एक से अधिक कैंडिडेट की हो सकती है, और अलग-अलग कैंडिडेट की में ऐट्रिब्यूट्स की संख्या अलग-अलग हो सकती हैं। दूसरे शब्दों में आपको, सबसे कम ऐट्रिब्यूट्स वाली सुपर कुंजी ही कैंडिडेट की है, इस प्रकार से 'न्यूनतम' की व्याख्या नहीं करनी चाहिए।

एक कैंडिडेट की में दो गुण होते हैं:

- R के प्रत्येक टपल (tuple) में, K के मान विशिष्ट रूप से उस टपल की पहचान (uniquely identify) करते हैं : **(विशिष्टता / Uniqueness)**
- K के किसी भी प्रॉपर सबसेट (Proper Subset) में विशिष्टता गुण (uniqueness property) ना हो : **(Irreducibility)**

R : Relation और K : key है।

(iii) प्राथमिक कुंजी (Primary Key) :

एक रिलेशन (relation) की प्राथमिक कुंजी (Primary Key) एक उम्मीदवार कुंजी (Candidate Key) है जिसे विशेष रूप से रिलेशन के लिए कुंजी (Key) चुना जाता है। दूसरे शब्दों में, यह एक कुंजी का चुनाव है, और केवल एक उम्मीदवार कुंजी (Candidate Key) हो सकती है जिसे प्राथमिक कुंजी (Primary Key) के लिए मान्य किया गया हो ।

पहचान कुंजीयों के बीच संबंध :

कुंजीयों के बीच संबंध:

सुपर की (Super Key) \supseteq उम्मीदवार कुंजी (Candidate Key) \supseteq प्राथमिक कुंजी (Primary Key)

2.3.2 विदेशी कुंजी (Foreign Key) :

एक संबंध (relation) के ऐसे एट्रिब्यूट्स, जो किसी अन्य संबंध (relation) की एक उम्मीदवार कुंजी (Candidate Key) से मेल खाते हैं, फॉरेन की कहलाते हैं । एक संबंध (relation) में कई विदेशी कुंजी (फॉरेन की / Foreign Key) हो सकती हैं, ये कुंजियाँ विभिन्न संबंधों (relations) से जुड़ी हो सकती हैं। विदेशी कुंजियाँ (Foreign Key) उपयोगकर्ताओं (users) को एक संबंध (relation) की जानकारी को दूसरे संबंध (relation) की जानकारी से लिंक करने की अनुमति देती हैं। फॉरेन की के बिना, एक डेटाबेस असंबंधित तालिकाओं (tables) का मात्र एक संग्रह होगा।

3. रेफरेंसियल इंटीग्रिटी क्या है? (What is Referential Integrity ?)

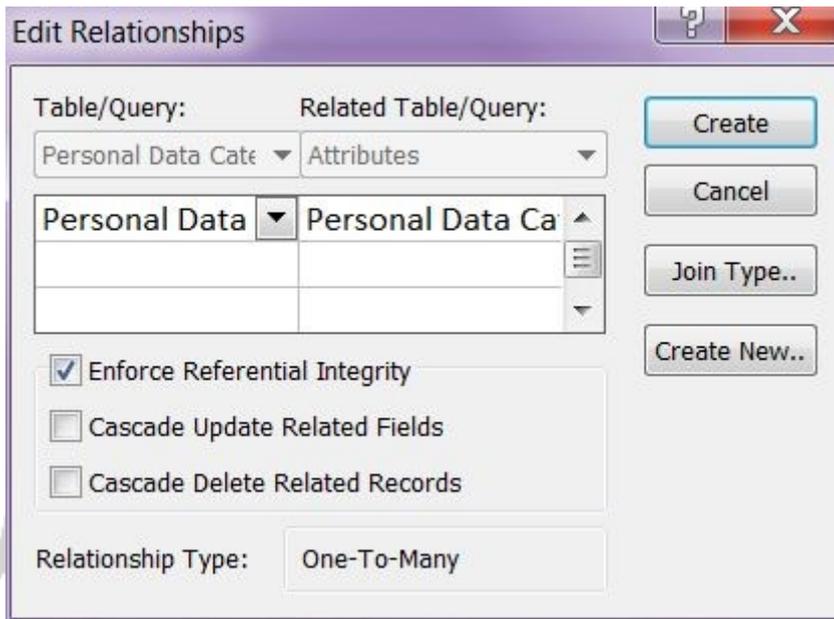
जब आप डेटाबेस डिज़ाइन करते हैं, तो आप डेटा रिडंडेंसी को कम करने के लिए अपनी डेटाबेस जानकारी को कई विषय-आधारित टेबल्स में विभाजित करते हैं। फिर आप सामान्य टेबल्स को संबंधित टेबल्स में रखकर डेटा को एक साथ लाने के लिए एक तरीका प्रदान करते हैं। उदाहरण के लिए, **one-to-many relationship** का प्रतिनिधित्व करने के लिए आप “एक” टेबल से **Primary key** लेते हैं और इसे “**Many**” टेबल में अतिरिक्त फ़ील्ड के रूप में जोड़ते हैं। डेटा को एक साथ वापस लाने के लिए, एक्सेस ” **Many**” टेबल में मान लेता है और “एक” टेबल में संबंधित मान को देखता है। इस तरह ” **Many**” टेबल में मान “एक” टेबल में संबंधित मानों को संदर्भित करते हैं।

टेबल रिलेशनशिप को **Referential Integrity** के मानकों का पालन करना चाहिए, नियमों का एक सेट जो नियंत्रित करता है कि आप संबंधित टेबल्स के बीच डेटा को कैसे हटा या संशोधित कर सकते हैं। टेबल रिलेशनशिप में **Referential Integrity** उपयोगकर्ताओं को गलती से संबंधित डेटा को हटाने या बदलने से रोकती है। आप **Referential Integrity** लागू कर सकते हैं जब: सामान्य फ़ील्ड प्राथमिक टेबल की **Primary key** है; संबंधित फ़ील्ड्स में एक ही प्रारूप है; या दोनों टेबल एक ही डेटाबेस से संबंधित हैं। यदि प्राथमिक टेबल में कर्मचारियों और संबंधित टेबल्स की एक सूची होती है तो उन कर्मचारियों के बारे में अतिरिक्त जानकारी होती है, और एक कर्मचारी छोड़ देता है, उसका रिकॉर्ड प्राथमिक टेबल से हटा दिया जाता है। उनके रिकॉर्ड सभी संबंधित टेबल्स में भी हटा दिए जाने चाहिए। एक्सेस आपको संबंधित डेटा को बदलने या हटाने की अनुमति देता है, लेकिन केवल तभी जब इन परिवर्तनों को संबंधित टेबल्स की श्रृंखला के माध्यम से कैस्केड किया जाता है। आप कैस्केड अपडेट संबंधित फ़ील्ड्स और कैस्केड को रिलेशनशिप कॉन्फ़िगरेशन डायलॉग बॉक्स में संबंधित रिकॉर्ड्स चेक बॉक्स हटाकर इसे कर सकते हैं।

3.1 एमएस एक्सेस में रेफरेंशियल इंटीग्रिटी नियम (Referential Integrity Rules in MS Access)

Relationship Window में Referential Integrity का Option होता है इसका अर्थ है कि यदि दो टेबल के मध्य **Referential Integrity rule** स्थापित हैं तो प्रथम टेबल में यदि किसी रिकॉर्ड में सुधार या अपडेट करते हैं तो इससे संबंधित टेबल में स्वतः ही संशोधन हो जाता है इसी प्रकार यदि प्रथम टेबल में से किसी रिकॉर्ड को **delete** करते हैं तो इससे संबंधित टेबल में से वह रिकॉर्ड **delete** हो जाता है। टेबल की **Relationship Create** करते समय **Edit Relationship Window** में **Referential Integrity** के तीन **Option** होते हैं।

- a. Enforce Referential Integrity
- b. Cascade Update Related Fields
- c. Cascade Delete Related Fields



(a) Cascade Update Related Fields :

यदि **Enforce Referential Integrity** के साथ **Cascade Update Related Fields Check Box** को सेलेक्ट करते है तो टेबल के किसी रिकॉर्ड को **Update** करने पर इससे संबंधित टेबल में उस रिकॉर्ड से संबंधित **Information** स्वयं **Update** हो जाएगी।

(b) Cascade Delete Related Fields :

इसी प्रकार यदि **Cascade Delete Related Check Box** को सेलेक्ट करते हैं तो टेबल में किसी रिकॉर्ड को **Delete** करने पर इससे जुड़ी हुई टेबल में से रिकॉर्ड स्वयं ही **Delete** हो जाएगा।

(c) Enforce Referential Integrity :

Enforce Referential Rule को **Set** करने के लिए **Enforce Referential Integrity Check Box** पर **Click** करते हैं इस के साथ ही अपनी आवश्यकतानुसार **Cascade Update Related Fields** एवं **Cascade Delete Related Fields** पर **Click** करके **Create Button** पर **Click** करते हैं।

डेटाबेस यूजिंग एमएस-एक्सेस

माइक्रोसॉफ्ट एक्सेस

माइक्रोसॉफ्ट ने 13 नवंबर, 1992 को एक्सेस का पहला वर्जन 1.0 जारी किया और मई 1993 में एक्सेस 1.1 रिलीज किया। इसके बाद एम एस एक्सेस के कई वर्जन बाजार में आये सभी वर्जन की अपनी अलग अलग विशेषताये हैं।

एम एस एक्सेस एक Data Management System Software है। ये MS Office का ही एक Software है। इसकी मदद से खुद का Data Entry Program बनाया जा सकता है। इसमें किसी विशेष समूह या संस्था की अलग-अलग जानकारी को रखा जाता है, जैसे कि किसी कॉलेज के Students का Data, किसी कंपनी में काम करने वाले कर्मचारियों की जानकारी जैसे उनका नाम, पता, सैलरी, उम्र आदि।

Introduction to database -What is a Database ?

डेटाबेस का परिचय - डेटाबेस क्या होता है ?

डेटाबेस के बारे में जानने से पहले हमें डाटा क्या होता है यह पता होना चाहिए। डेटा का अर्थ है वैल्यू, जैसे 27, 2500, मोती। डेटा जिसका कुछ अर्थ हो उसे जानकारी कहते हैं। मीनिंगफुल डाटा / सार्थक डेटा का अर्थ है सूचना (इंफॉर्मेशन)। मतलब ऐसा डाटा जिसके साथ एट्रिब्यूट भी हो उसे सूचना (इंफॉर्मेशन) कहा जाता है।

डाटा को तथ्यों या आंकड़ों के रूप में परिभाषित किया जाता है या ऐसी जानकारी जो एक कंप्यूटर में संग्रहित या उपयोग की जा सकती है जैसे टेक्स्ट संख्याएं यहां चित्र या धोनी का डिजिटल रूप।

एक डेटाबेस डाटा का एक व्यवस्थित और संगठित संग्रह होता है। सामान्यतः इसे कंप्यूटर सिस्टम पर इलेक्ट्रॉनिक रूप से रखा (संग्रहीत) और एक्सेस किया जाता है। जहां डेटाबेस अधिक जटिल होते हैं, उन्हें अक्सर औपचारिक डिजाइन और मॉडलिंग तकनीकों का उपयोग करके विकसित किया जाता है।

डेटाबेस में किसी समूह या संस्था की जानकारी रखी जाती है।

डेटाबेस मैनेजमेंट सिस्टम (DBMS) : डेटाबेस मैनेजमेंट सिस्टम एक सॉफ्टवेयर होता है, जिसके अंदर डेटाबेस तैयार किए जाते हैं । डेटाबेस मैनेजमेंट सिस्टम में डाटा को व्यवस्थित रूप से स्टोर किया जाता है, रिप्लाइ किया जाता है और एक्सेस किया जाता है।

डेटाबेस मैनेजमेंट सिस्टम (DBMS) वह सॉफ्टवेयर है जो डेटा को रखने और उसका विश्लेषण करने के लिए अंतिम उपयोगकर्ताओं (end user), अनुप्रयोगों (applications) और डेटाबेस के साथ सहभागिता (interact) करता है।

डेटाबेस मैनेजमेंट सिस्टम (DBMS) एक एप्लीकेशन प्रोग्राम है जो बड़ी मात्रा में जानकारी के प्रवेश (data entry), डेटा के कुशल भंडारण (storage), प्रसंस्करण (processing), पुनर्प्राप्ति (retrieval) प्रदान करता है।

एक डेटाबेस एक डेटा संरचना (Data Structure) है जो संगठित जानकारी संग्रहीत करता है।

अधिकांश डेटाबेस में कई टेबल होते हैं, जिनमें से प्रत्येक में कई फ़ील्ड हो सकते हैं। उदाहरण के लिए, एक कंपनी डेटाबेस में उत्पादों, कर्मचारियों और वित्तीय रिकॉर्ड के लिए टेबलों (tables) को शामिल किया जा सकता है। इन टेबलों में से प्रत्येक में अलग-अलग फ़ील्ड होंगे, जो टेबलों में संग्रहीत जानकारी से संबंधित होंगे ।

प्रारंभिक डेटाबेस अपेक्षाकृत "फ्लैट" थे, जिसका अर्थ है कि वे एक स्प्रेडशीट की तरह सरल पंक्तियों (rows) और स्तंभों (columns) तक सीमित थे। हालांकि, आज के रिलेशनल डेटाबेस उपयोगकर्ताओं को, विभिन्न टेबलों में संग्रहीत डेटा के संबंधों के आधार पर, जानकारी तक पहुंचने, अपडेट करने और खोज करने की अनुमति देते हैं। रिलेशनल डेटाबेस कई डेटाबेस पर आधारित क्वेरिस (Queries) चला सकते हैं। जबकि शुरुआती डेटाबेस केवल पाठ (Text) या संख्यात्मक (Numeric) डेटा को संग्रहीत कर सकते थे । आधुनिक डेटाबेस उपयोगकर्ताओं को अन्य डेटा प्रकार जैसे ध्वनि क्लिप, चित्र और वीडियो को संग्रहीत करने की भी अनुमति देता है।

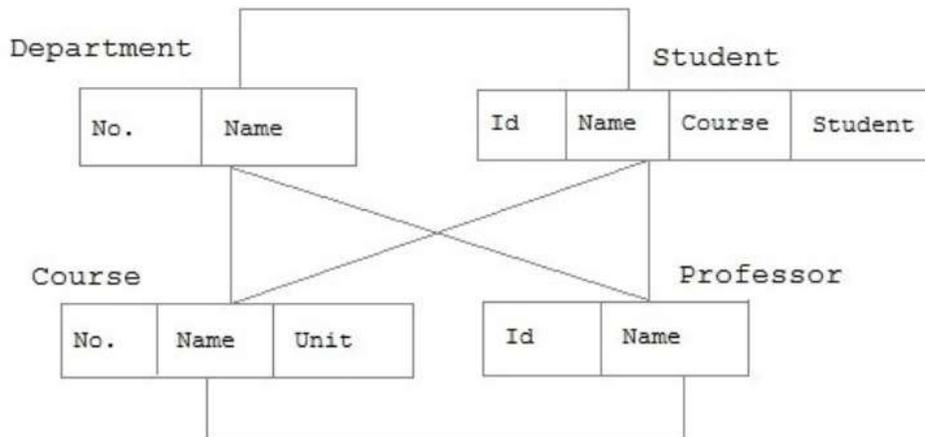
डेटाबेस मैनेजमेंट सिस्टम के कुछ उदाहरण : Dbase, Sybase, FoxPro, IMS, Microsoft Access, DB2, Oracle, SQL Server, MySQL etc.

डाटा मॉडल :

डेटाबेस मैनेजमेंट सिस्टम में डेटाबेस बनाए जाते हैं । डेटाबेस में डाटा के मध्य किस तरह का संबंध है, यह इस बात पर निर्भर करता है कि डेटाबेस किस डाटा मॉडल पर आधारित है । किसी डेटाबेस में उपस्थित डाटा के मध्य संबंधों को बताने के लिए और डेटा की प्रकृति बताने के लिए डाटा मॉडल का उपयोग किया जाता है । डेटाबेस मैनेजमेंट सिस्टम डाटा मॉडल पर आधारित होते हैं ।

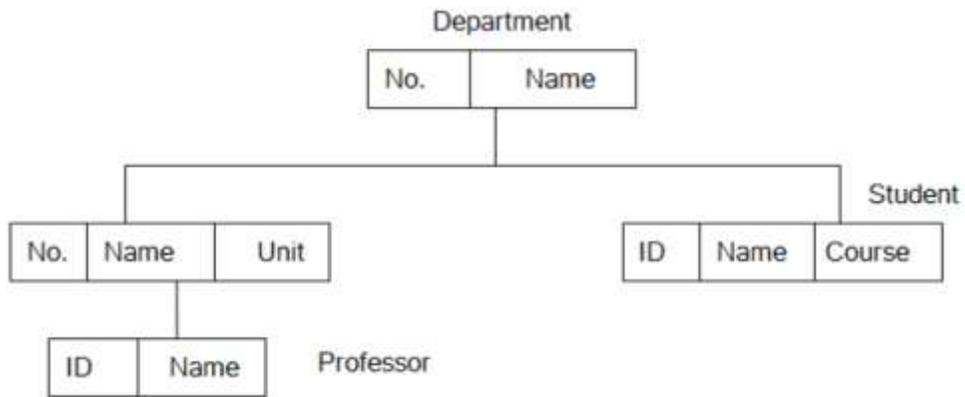
मुख्य रूप से तीन प्रकार के डाटा मॉडल होते हैं । अतः इन पर आधारित तीन प्रकार के डेटाबेस होते हैं जो निम्नलिखित हैं :

1. **नेटवर्क डेटाबेस (Network Database) :** नेटवर्क मॉडल पर आधारित डेटाबेस को नेटवर्क डेटाबेस कहते हैं इस प्रकार के डेटाबेस में डाटा को रिकार्ड (Record) के रूप में दर्शाया जाता है और डेटा के बीच संबंध लिंक (Link) के रूप में दर्शाया जाता है । इस डाटा मॉडल में डाटा को एक ग्राफ (Graph) के रूप में व्यवस्थित किया जाता है, जिसमें एक रिकॉर्ड से दूसरे रिकॉर्ड तक पहुंचने की कई रास्ते (Paths) हो सकते हैं ।

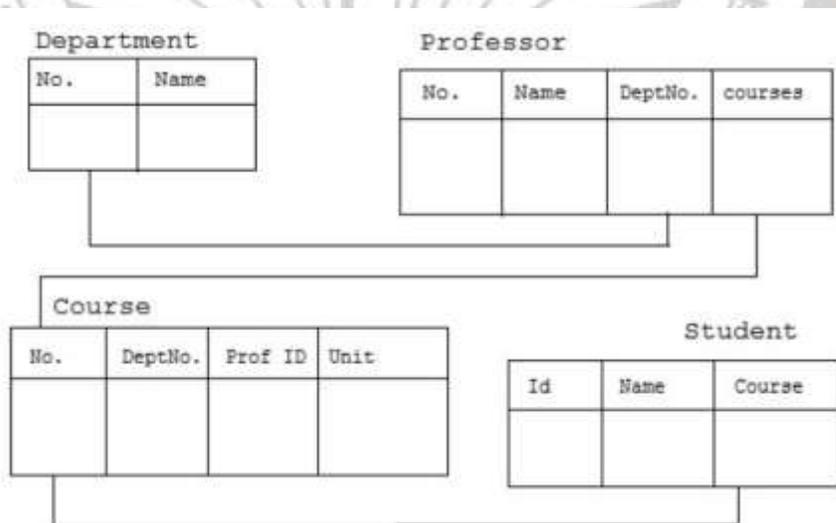


2. **हायरार्चिकल डेटाबेस (Hierarchical Database) :** हायरार्चिकल मॉडल पर आधारित डेटाबेस को हायरार्चिकल डेटाबेस कहते हैं । इस तरह के डेटाबेस में डाटा को ट्री स्ट्रक्चर (पेरेंट चाइल्ड) के रूप में व्यवस्थित किया जाता है और दर्शाया जाता है । ट्री स्ट्रक्चर (Tree Structure) में नोड्स (Nodes) लिंक के माध्यम से जुड़े हुए होते हैं । डाटा में पेरेंट चाइल्ड रिलेशनशिप (Parent-Child Relationship) होती है । इस

डाटा मॉडल में प्रत्येक चाइल्ड रिकॉर्ड का केवल एक पेरेंट होता है, लेकिन एक पेरेंट रिकॉर्ड की बहुत सारे चाइल्ड रिकॉर्ड हो सकते हैं ।



3. **रिलेशनल डेटाबेस (Relational Database)** : रिलेशनल मॉडल पर आधारित डेटाबेस को रिलेशनल डेटाबेस कहते हैं । इस डेटाबेस में डाटा टेबल्स (Tables) के रूप में संग्रहित होता है । टेबल में रो (Row) और कॉलम (Column) होते हैं । एक रो में एक रिकॉर्ड संग्रहित किया जाता है ।



Why use a Relational Database

हायरार्चिकल डेटाबेस में यह नुकसान है कि इसका स्ट्रक्चर सभी तरह के डेटाबेस में अप्लाई (apply) नहीं किया जा सकता यह डेटाबेस फ्लैक्सिबल (flexible) नहीं होता है ।

नेटवर्क डेटाबेस में यह नुकसान है कि इसमें रिकॉर्ड को अपडेट (update) और इंsert (insert) करना बहुत जटिल (complex) होता है।

रिलेशनल डेटाबेस का प्रयोग सबसे ज्यादा किया जाता है क्योंकि यह सबसे सरल है और आसानी से उपयोग किया जा सकता है । इस डेटाबेस में डाटा एक टेबल (Table) के अंदर स्टोर किया जाता है । टेबल में रो (Row) और कॉलम (Column) होते हैं । टेबल में एक रो एक एंटीटी (Entity) के रिकॉर्ड को दर्शाती है और कॉलम, एट्रीब्यूट्स (Attributes) / फील्ड्स (Fields) दर्शाते हैं । इस डेटाबेस में डाटा को इंsert (Insert), डिलीट (Delete) और अपडेट (update) करना बहुत आसान होता है । डाटा को इंsert , डिलीट और अपडेट करने के लिए एसक्यूएल क्वेरीज (SQL Queries) का उपयोग किया जाता है । रिलेशनल डेटाबेस में टेबल को रिलेशन भी कहते हैं इसीलिए इस डेटाबेस को रिलेशनल डेटाबेस कहा जाता है ।

कुछ प्रचलित रिलेशनल डीबीएमएस (Relational DBMS) के उदाहरण हैं : ओरेकल (Oracle), एसक्यूएल सर्वर (SQL Server), माय एसक्यूएल (MySQL), DB2 आदि ।

रिलेशनल डेटाबेस का सबसे मुख्य फायदा यह है कि इसमें डाटा टेबल के रूप में होता है जिसके कारण उपयोगकर्ता (User) इसे आसानी से समझ लेते हैं और एक्सेस (Access) कर लेते हैं । रिलेशनल मॉडल बहुत पावरफुल (Powerful) और सिंपल (Simple) है तथा बहुत ज्यादा फ्लैक्सिबल (Flexible) भी है ।

इसका यह नुकसान है कि ज्यादा डाटा होने पर यह जटिल बन जाता है और डाटा के मध्य की रिलेशनशिप भी जटिल और कठिन हो जाती है ।

रिलेशनल डेटाबेस मैनेजमेंट सिस्टम (RDBMS) : रिलेशनल डेटाबेस मैनेजमेंट सिस्टम एक ऐसा डेटाबेस मैनेजमेंट सिस्टम होता है, जो रिलेशनल डाटा मोडल पर आधारित है और जिसके अंदर रिलेशनल डेटाबेस तैयार किए जाते हैं ।

रिलेशनल डेटाबेस मैनेजमेंट सिस्टम के कुछ उदाहरण : Microsoft Access, Oracle, SQL Server, MySQL etc.

Overview of Database Design / डेटाबेस डिजाइन का अवलोकन :

डेटाबेस डिजाइन को इस प्रकार परिभाषित किया गया है :

अनुप्रयोगों (applications) के एक निर्धारित सेट के लिए एक संगठन (organisation) में उपयोगकर्ताओं की सूचना आवश्यकताओं (information needs) को समायोजित करने के लिए एक या अधिक डेटाबेस की तार्किक और भौतिक संरचना (logical and physical structure) डिजाइन करना ।

डेटाबेस डिजाइन एक डेटाबेस के विस्तृत डेटा मॉडल को प्रोड्यूस करने की प्रक्रिया है । डेटाबेस डिजाइन टर्म एक संपूर्ण डेटाबेस सिस्टम के डिजाइन के बहुत सारे भिन्न भागों का वर्णन करने के लिए प्रयोग की जाती है।

डेटाबेस डिजाइन के मुख्य स्टेप्स निम्नलिखित हैं :

1. Planning and Analysis
2. Conceptual design
3. Logical design
4. Physical design

1. Planning and Analysis (योजना और विश्लेषण) :

साइट अवलोकन (On Site Observation), साक्षात्कार (Interview), मैनुअल और प्रलेखन (Manual and Documentation), प्रश्नावली (Questionnaire) के आधार पर जानकारी प्राप्त करना और उसका विश्लेषण करना । जहाँ का भी डेटाबेस तैयार करना है वहाँ खड़े होकर अवलोकन (observe) करना अर्थात यह देखना कि वहाँ क्या क्या हो रहा है, किस तरह कार्य किया जा रहा है, कार्य को करने की क्या प्रक्रिया है, आदि । अवलोकन करने के बाद उस व्यक्ति का Interview लेना जिससे जानकारी प्राप्त करना है । आवश्यकताओं का संकलन साधारण: संगठन के यूजरों से पूछताछ करके किया जाता है । जानकारी को इकट्ठा करने का एक दूसरा तरीका प्रश्नावलियों का उपयोग है । अनुभव द्वारा प्रभावशाली प्रश्नवली बनाकर उन्हें लोगों में बाँट दिया जाता है और उनके उत्तर फॉर्म के रूप में एकत्र कर लिए जाते हैं ।

2. Conceptual design (वैचारिक डिजाइन या प्रारूप) :

डेटाबेस से सम्बंधित जानकारी को एकत्रित करने के बाद उस डाटा पार विचार किया जाता है कि जो जानकारी एकत्रित की गई है वह सही है या नहीं ।

एक बार डेटाबेस डिजाइनर उस डाटा के बारे में जानकारी प्राप्त कर लेता है, जो डेटाबेस में स्टोर होनी है, तब वह यह निर्धारित करता है कि कहां पर डाटा में निर्भरता (Dependency) है । कभी-कभी जब data बदल जाता है , तो वे अपने आप दूसरे डेटा को बदल सकते हैं , जो उचित नहीं होता है ।

डेटा मॉडल, वैचारिक डेटाबेस डिजाइन प्रक्रिया (Conceptual Database Design Process) का एक भाग है । दूसरा भाग कार्यात्मक मॉडल है (Functional Model) ।

डेटा मॉडल को प्लानिंग और एनालिसिस स्टेज से इनपुट मिलता है । डाटा मॉडल के दो आउटपुट होते हैं, पहला Entity-Relationship (ER Diagram) डायग्राम जो data-structure को चित्र के रूप में दिखाता है । ER Diagram की सहायता से एंड यूजर को डाटा मॉडल के विषय में बताना आसान हो जाता है । दूसरा आउटपुट है डाटा डिक्शनरी डॉक्यूमेंट । डाटा डिक्शनरी वह डॉक्यूमेंट है जो डाटा ऑब्जेक्ट, उनके मध्य संबंध और डेटाबेस के लिए आवश्यक नियमों की विस्तार से व्याख्या करता है ।

Conceptual design के टूल्स :-

- A - ER Diagram
- B - Relationship
- C - Normalization
- D - Anomaly check

3. Logical Design (लॉजिकल डिजाइन)

डेटाबेस का दूसरा चरण लॉजिकल डिजाइन होता है । डेटाबेस डिजाइन करते समय आवश्यक जानकारियों को इकट्ठा करने के बाद उस इंफॉर्मेशन को अलग-अलग बांट कर Data का लॉजिकल स्ट्रक्चर तैयार किया जाता है ।

लॉजिकल डिजाइन को डेटाबेस के विस्तृत वर्णन के बारे में सोच है जैसे डेटाबेस में क्या इंफॉर्मेशन स्टोर होगी , किस टाइप की इंफॉर्मेशन स्टोर होगी आदि ।

4. Physical Design (फिजिकल डिजाइन)

डेटाबेस का तीसरा चरण डेटाबेस डिजाइन को फिजिकली डिजाइन करना है अर्थात आवश्यक सूचनाओं को इकट्ठा करने उनके बारे में विस्तृत जानकारी निकालने के बाद उस डाटा को भौतिक रूप से प्रयोग में लाया जाता है ।

डेटाबेस की फिजिकल डिजाइन स्टोरेज मीडिया पर डेटाबेस के फिजिकल कॉन्फिगरेशन को specify करता है । फिजिकल डिजाइन डेटाबेस के सबसे नीचे के लेवल के डेटा का वर्णन करने के लिए प्रयोग किया जाता है । फिजिकल डिजाइन के अंतर्गत डेटाबेस के वास्तविक स्ट्रक्चर को डिजाइन किया जाता है ।

Data Normalization (Determining tables, Determining Fields, Determining Relationships)

नोर्मलाइजेशन क्या हैं ? (What is Normalization?) :

Normalization (मानकीकरण) डेटाबेस डिजाइन को सरल बनाता है । इसे डेटाबेस नोर्मलाइजेशन या डेटा नोर्मलाइजेशन के रूप में भी जाना जाता है । नोर्मलाइजेशन डेटाबेस डिजाइन का एक महत्वपूर्ण हिस्सा है , क्योंकि यह डेटाबेस की गति, सटीकता में सहायता करता है। नोर्मलाइजेशन एक ऐसी तकनीक है जिसका उपयोग रिलेशनल (relational) डेटाबेस को डिजाइन करने के लिए किया जाता है । नोर्मलाइजेशन दो चरणों में होती है जो डाटा से Repeating Groups को हटा कर टेबल के रूप में रखती है और उसके बाद रिलेशनल टेबल से एक जैसी यानी कि duplicate entries को हटाती है ।

नोर्मलाइजेशन का काम redundancy को कम करना होता है और redundancy को कम करने का अर्थ है एक information को एक ही बार स्टोर करना। एक ही Information को एक से ज्यादा बार स्टोर करने से स्टोरेज बढ़ता है। Normalized Relation से मतलब है कि जब भी

डेटाबेस में रिलेशनस को परिवर्तित (alter) किया जाए, तो इनफार्मेशन गुम नहीं होना चाहिए।

नार्मल फॉर्म में किसी डेटाबेस टेबल को बनाना नोर्मलाईजेशन कहलाता है । नोर्मलाईजेशन में किसी टेबल के प्रत्येक एट्रिब्यूट के बीच आपसी निर्भरता का विश्लेषण किया जाता है और फिर बड़े टेबल का प्रोजेक्शन लेकर उससे छोटे-छोटे टेबल बनाये जाते हैं । प्रोजेक्शन ऑपरेशन का अर्थ है किसी टेबल में उपस्थित एट्रिब्यूट में से कुछ एट्रिब्यूट को लेकर टेबल बनाना ।

डेटाबेस को नोर्मलाईजेशन करके, आप डेटा को टेबल और कॉलम में व्यवस्थित कर सकते हैं। आप सुनिश्चित करते हैं कि प्रत्येक टेबल में केवल संबंधित डेटा होता है। यदि डेटा सीधे संबंधित नहीं है, तो आप उस डेटा के लिए एक नई टेबल बनाते हैं।

उदाहरण के लिए, यदि आपके पास “ग्राहक” टेबल है, तो आप आमतौर पर उन उत्पादों के लिए एक अलग टेबल बना सकते हैं, जिन्हें वे ऑर्डर कर सकते हैं (आप इस टेबल को “Product” कह सकते हैं)। आप ग्राहकों के आदेशों के लिए एक और टेबल तैयार करेंगे (“ऑर्डर” टेबल कहा जाता है)। और यदि प्रत्येक ऑर्डर में कई आइटम हो सकते हैं, तो आप आमतौर पर प्रत्येक ऑर्डर आइटम को स्टोर करने के लिए एक और टेबल बनाते हैं (“ऑर्डरइटम” टेबल कहा जाता है)। इन सभी टेबलों को उनकी प्राथमिक कुंजी (Key) से जोड़ा जाएगा, जो आपको इन सभी टेबलों में संबंधित डेटा ढूँढने की अनुमति देता है (जैसे किसी दिए गए ग्राहक द्वारा सभी ऑर्डर) ।

नोर्मलाईजेशन के लाभ (Benefits of Normalization) :

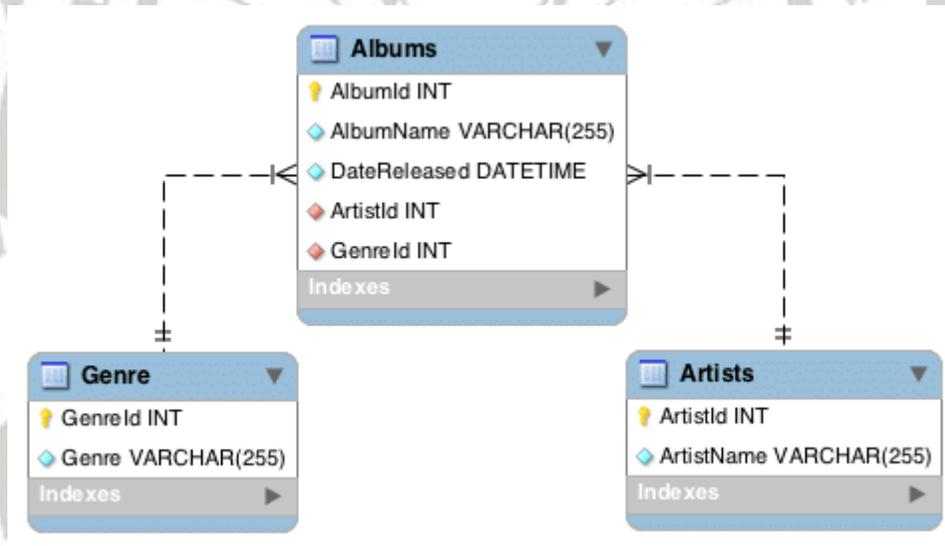
- यह डेटा रिडंडेंसी को कम करता है (डुप्लिकेट डेटा)।
- यह शून्य मूल्यों (null values) को कम करता है।
- यह अधिक कॉम्पैक्ट डेटाबेस में परिणाम देता है। (कम डेटा रिडंडेंसी / शून्य मानों के कारण)।
- यह डेटा संशोधन मुद्दों को कम करता है ।
- यह क्वेरी को सरल बनाता है।

- यह डेटाबेस संरचना को स्वच्छ और समझने में आसान बनाता है।
- आप मौजूदा डेटा को जरूरी रूप से प्रभावित किए बिना डेटाबेस का विस्तार कर सकते हैं।
- इसमें इंडेक्स खोजना, सॉर्ट करना और बनाना तेजी से हो सकता है, क्योंकि टेबल संकुचित होते हैं, और डेटा पंक्ति पर अधिक पंक्तियां फिट होती हैं।

एक नॉर्मलाइजेशन डेटाबेस का उदाहरण (Example of a Normalized Database) :

एक रिलेशनल डेटाबेस डिज़ाइन करते समय, आमतौर पर एक स्कीमा बनाने से पहले डेटा को नॉर्मलाइजेशन करता है। डेटाबेस स्कीमा संगठन और डेटाबेस की संरचना को निर्धारित करता है - मूल रूप से डेटा कैसे संग्रहीत किया जाएगा।

Normalization डेटाबेस स्कीमा का एक उदाहरण यहां दिया गया है:



यह स्कीमा डेटा को तीन अलग-अलग टेबलों में विभाजित करता है। प्रत्येक टेबल उस डेटा में काफी विशिष्ट है जो इसे स्टोर करती है - एल्बमों के लिए एक टेबल है, कलाकारों के लिए एक है, और दूसरा जो डेटा के लिए विशिष्ट डेटा रखता है। हालांकि, रिलेशनशिप मॉडल हमें इन

टेबलों के बीच रिलेशन बनाने की इजाजत देता है, हम यह पता लगा सकते हैं कि कौन से एल्बम किस कलाकार से संबंधित हैं, और किस शैली में वे संबंधित हैं।

Normalization के प्रकार (Types of Normalization) :

1. First Normal Form (1NF):-

First Normal Form को 1NF से भी दर्शाते हैं। एक Relational Table 1NF में होती है, जब कॉलम की सभी values Atomic (एकल मान) होती है। प्रत्येक कॉलम में प्रत्येक रो के लिए सिर्फ एक वैल्यू होनी चाहिए। रिलेशनल डेटाबेस में बनाया गया टेबल by default 1NF में होता है। हम इसे इस तरह से भी समझ सकते हैं कि कोई table 1NF में होती है यदि

- हर एक Cell में Atomic Value होनी चाहिए।
- कॉलम में entries/values का प्रकार एक जैसा ही होना चाहिए। एक कॉलम में सभी वैल्यूज का डोमेन (Domain) एक जैसा होना चाहिए।

2. Second Normal Form (2NF):-

एक टेबल या रिलेशन तब 2nd Normal Form में होता है जब वह 1st Normal Form की सभी जरूरतों को पूरी करता हो और सभी Non-Key Attributes पूरी तरह से Primary Key पर निर्भर हों।

3. Third Normal Form (3NF):-

कोई टेबल या रिलेशन तब 3rd Normal Form में होता है जब वह 2nd Normal Form की सभी जरूरतों को पूरी करता हो तथा उनमें Transitive Function Dependency नहीं होनी चाहिए।

4. बॉयस-कोड नार्मल फॉर्म / Boyce Codd Normal Form (BCNF):-

3NF, जो आज डेटाबेस में ज्ञात अधिकांश विसंगतियों को समाप्त करता है, वाणिज्यिक डेटाबेस और CASE टूल में Normalization का सबसे अच्छा मानक है। इसमें उपस्थित कुछ शेष

विसंगतियों को बॉयस-कोड नार्मल फॉर्म (BCNF) द्वारा समाप्त किया जा सकता है। BCNF को 3NF का एक मजबूत स्वरूप माना जाता है।

परिभाषा : एक टेबल R बॉयस-कोड नार्मल फॉर्म (BCNF) में है यदि प्रत्येक नॉन-ट्रिविअल FD, $X \rightarrow A$ के लिए, X एक सुपरकी है।

3NF की तुलना में BCNF, Normalization का एक मजबूत रूप है क्योंकि यह 3NF के लिए दूसरी कंडीशन को समाप्त कर देता है, जिसमें FD के दाईं ओर एक प्राइम एट्रिब्यूट (Prime Attribute) होने की अनुमति है। इस प्रकार, एक टेबल में FD के हर बाईं ओर एक सुपर की होना चाहिए। पिछली परिभाषाओं द्वारा हर टेबल जो BCNF में है वो 3NF, 2NF और 1NF में भी है।

Primary/Foreign Key, One-to-Many, Many-to-Many, One-to-One Relationship

Primary Key / प्राइमरी की :

Primary key एक टेबल में प्रत्येक रिकॉर्ड को यूनिक (अद्वितीय) रखने के लिए प्रयोग की जाती है। इस key के attribute कभी बदलना नहीं चाहिए जैसे एक व्यक्ति का एड्रेस (पता) प्राइमरी Key का हिस्सा नहीं होना चाहिए क्योंकि यह बदल सकता है। परंतु Employee code नहीं बदला जा सकता, जब तक की वह व्यक्ति उस संस्था को ना छोड़ दें। प्राइमरी की किसी भी टेबल में यूनिक की होती है जो पूरे रिकॉर्ड को रिप्रेजेंट करती है। Primary Key Column को Null नहीं रखा जा सकता है।

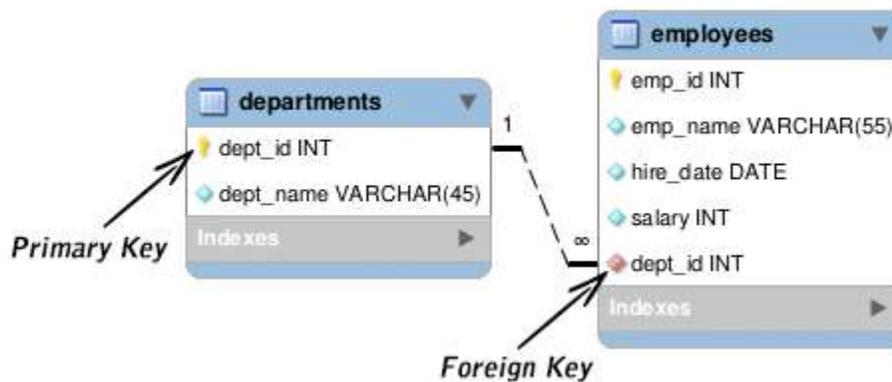
एक Table में केवल एक Primary Key हो सकती है। Primary Key में एक या अधिक फ़ील्ड होते हैं जो Table में संग्रहीत प्रत्येक रिकॉर्ड को विशिष्ट रूप से पहचानते हैं। अक्सर, एक Unique पहचान संख्या होती है, जैसे एक आईडी नंबर, एक सीरियल नंबर, या एक कोड, जो Primary Key के रूप में कार्य करता है। उदाहरण के लिए, आपके पास ग्राहक Table है जहां प्रत्येक ग्राहक के पास एक Unique ग्राहक आईडी नंबर होता है। ग्राहक आईडी फ़ील्ड, ग्राहक Table की Primary Key है। जब Primary Key में एक से अधिक फ़ील्ड होते हैं, तो यह आमतौर पर पूर्व-मौजूदा फ़ील्ड से बना होता है, जो एक साथ लेते हैं और Unique मान प्रदान करते हैं। उदाहरण के लिए, आप लोगों के बारे में एक Table के लिए, Primary Key के रूप

में अंतिम नाम, पहला नाम, और जन्म तिथि के combination(संयोजन) का उपयोग कर सकते हैं।

Foreign Key / फॉरेन की :

फॉरेन की का प्रयोग एक टेबल की, दूसरी टेबल के साथ रिलेशनशिप को स्थापित करने के लिए किया जाता है अर्थात एक टेबल की लिंक दूसरी टेबल के साथ बनाने के लिए किया जाता है। जैसे हम College के नाम से एक डेटाबेस को Create करना चाहते हैं। इस डेटाबेस में स्टूडेंट (student) तथा कोर्स (course) के नाम से 2 टेबल बनाते हैं।

एक Table में एक या अधिक Foreign Key भी हो सकती है। एक Foreign Key में वे मान होते हैं जो किसी अन्य Table की Primary Key में मानों के अनुरूप होते हैं। उदाहरण के लिए, आपके पास employees Table है जिसमें प्रत्येक employee का एक डिपार्टमेंट आईडी नंबर होता है, जो departments Table में रिकॉर्ड के अनुरूप होता है। डिपार्टमेंट आईडी फ़ील्ड ,employee Table की एक Foreign Key है।



Difference between Primary key and Foreign key :

प्राइमरी और फॉरेन की के बीच अंतर :

Primary Key	Foreign Key
प्राइमरी की एक कॉलम या कॉलम का सेट होता है जो एक टेबल में एक रो को यूनिक बनाता है ।	फॉरेन की एक कॉलम या कॉलम का सेट होता है जो दूसरी टेबल के एक प्राइमरी की या एक कैंडिडेट की को रेफर करता है ।
एक टेबल में सिंगल (सिर्फ एक) प्राइमरी की हो सकती है ।	एक टेबल में एक से अधिक फॉरेन की हो सकती हैं।
प्राइमरी की यूनिक होती है । प्राइमरी की नल नहीं हो सकती ।	फॉरेन की नल हो सकती हैं ।
प्राइमरी की Entity (एंटीटी) इंटीग्रिटी से सम्बंधित है ।	फॉरेन की referential (रेफरेंसीअल) इंटीग्रिटी से सम्बंधित हैं ।
प्राइमरी की को पैरेंट की (Parent Key) के नाम से भी जाना जाता है ।	फॉरेन की को चाइल्ड की (Child Key) के नाम से भी जाना जाता है ।

Connectivity and Cardinality of a Relationship :

एक Relationship की कनेक्टिविटी Relationship में संबद्ध entity के instances (उदाहरणों) के Mapping मानचित्रण का वर्णन करती है। कनेक्टिविटी की वैल्यू "एक" या "अनेक" होती हैं।

एक Binary Relationship की कार्डिनैलिटी (Cardinality) प्रत्येक दो entities के लिए संबंधित occurrences की वास्तविक संख्या है। Relationship के लिए निम्न प्रकार की कनेक्टिविटी हैं:

1. एक-से-एक (One-to-One Relationship)
2. एक से अनेक (One-to-Many Relationship)

3. अनेक -से-अनेक (Many-to-Many Relationship)

One-to-One Relationship : One-to-One (1:1) Relationship तब होता है जब एक Entity Set A का सिर्फ एक उदाहरण (instance) , एक Entity Set B के एक उदाहरण (instance) के साथ जुड़ा हुआ है । उदाहरण:

कंपनी में अधिकारियों को प्रत्येक को अपना कार्यालय सौंपा गया है। प्रत्येक अधिकारी के लिए एक अद्वितीय कार्यालय मौजूद है और प्रत्येक कार्यालय के लिए एक अद्वितीय अधिकारी मौजूद है ।

One-to-Many Relationship : One-to-Many (1: N) Relationship तब होता है जब Entity Set A के एक उदाहरण (instance) के लिए, Entity Set B के शून्य, एक या कई उदाहरण (instances) हैं । लेकिन Entity Set B के एक उदाहरण (instance) के लिए, Entity Set A का केवल एक उदाहरण (instance) है ।

उदाहरण :

एक विभाग में कई कर्मचारी होते हैं।

प्रत्येक कर्मचारी का एक विभाग होता है।

Many-to-Many (M:N) Relationship : Many-to-Many (M:N) Relationship तब होता है जब Entity Set A के एक उदाहरण (instance) के लिए, Entity Set B के शून्य, एक या अनेक उदाहरण (instance) हो । और Entity Set B के एक उदाहरण (instance) के लिए, Entity Set A के शून्य, एक या अनेक उदाहरण (instance) हो ।

उदाहरण :

कर्मचारियों को एक ही समय में दो से अधिक परियोजनाओं को सौंपा जा सकता है।

परियोजनाओं को कम से कम तीन कर्मचारियों को सौंपा जाना चाहिए।

एमएस एक्सेस 2007 का परिचय (Introduction to MS Access 2007) :

डाटाबेस डेवलपमेंट के लिए MS Access 2007 एक महत्वपूर्ण और शक्तिशाली प्लेटफार्म है । MS Access 2007 माइक्रोसॉफ्ट कंपनी द्वारा तैयार किया गया एक डेटाबेस एप्लीकेशन सॉफ्टवेयर है जिसके माध्यम से किसी भी कंपनी, संस्था आदि के बारे में हम ज्यादा से ज्यादा सूचनाओं को इकट्ठा कर सकते हैं । तथा बाद में कोई भी विशिष्ट जानकारी जो की उस संस्था से सम्बंधित हो, उसे प्राप्त कर सकते हैं । MS Access को हम “collection of database “ भी कह सकते हैं ।

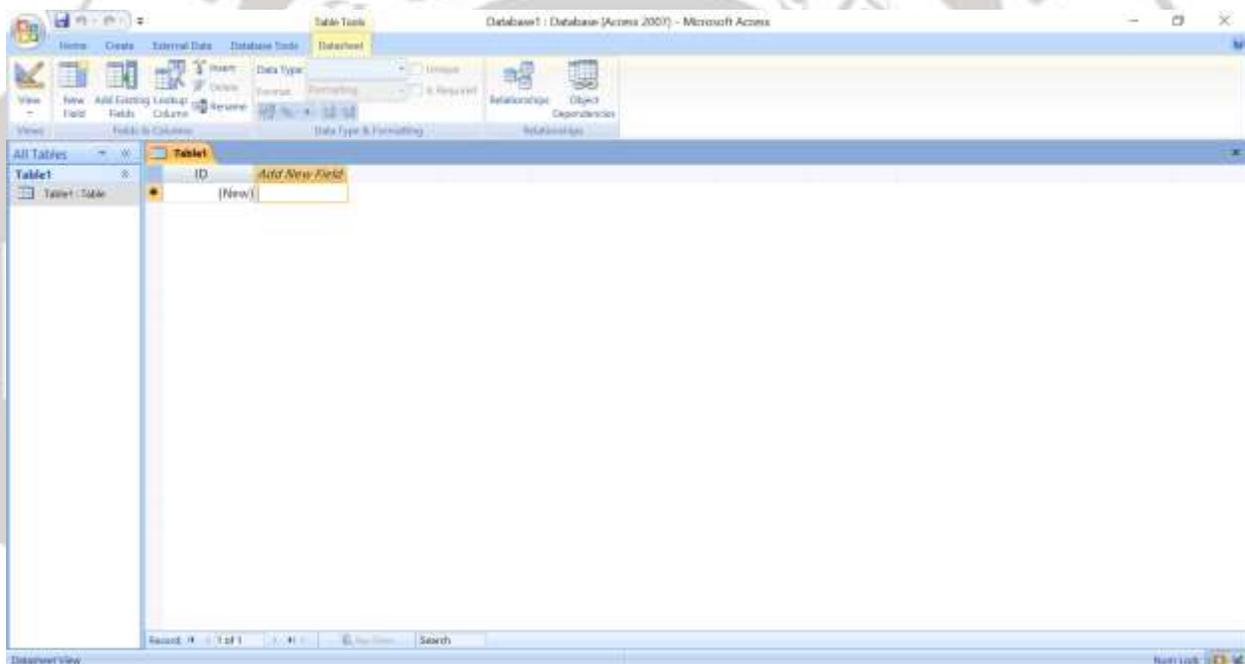
एक्सेस एक डाटाबेस मैनेजमेंट सिस्टम (डीबीएमएस) है जो एक ग्राफिकल यूजर इंटरफेस और सॉफ्टवेयर डेवलपमेंट टूल्स के साथ रिलेशनल माइक्रोसॉफ्ट जेट डाटाबेस इंजन को जोड़ती है।

- यह एक्सेस जेट डेटाबेस इंजन के आधार पर डेटा को अपने प्रारूप में स्टोर करता है।
- रिलेशनल डेटाबेस की तरह, माइक्रोसॉफ्ट एक्सेस आपको संबंधित जानकारी को आसानी से लिंक करने की अनुमति देता है ।
- यह अन्य प्रोग्राम और डेटाबेस में स्टोर डेटा को सीधे इम्पोर्ट या लिंक भी कर सकता है।
- जैसा कि इसके नाम से तात्पर्य है, एक्सेस कई लोकप्रिय पीसी डेटाबेस प्रोग्रामों सहित, डेस्कटॉप पर, सर्वर पर, मिनीकंप्यूटर पर या मेनफ्रेम पर, और संग्रहीत डेटा के साथ कई लोकप्रिय पीसी डेटाबेस प्रोग्राम समेत अन्य स्रोतों से डेटा के साथ सीधे काम कर सकता है।
- आप डेटा प्रोसेसिंग फ़ाइलों, स्प्रेडशीट्स या डेटाबेस फ़ाइलों से डेटा को डेटा निर्यात और आयात कर सकते हैं।
- एक्सेस SQL सर्वर, ओरेकल और डीबी 2 समेत ओपन डाटाबेस कनेक्टिविटी (ओडीबीसी) मानक का समर्थन करने वाले सबसे लोकप्रिय डेटाबेस के साथ काम कर सकता है।
- सॉफ्टवेयर डेवलपर्स एप्लीकेशन सॉफ्टवेयर विकसित करने के लिए माइक्रोसॉफ्ट एक्सेस का उपयोग कर सकते हैं।

एमएस एक्सेस वातावरण के साथ काम करना / (Working with MS Access Environment):

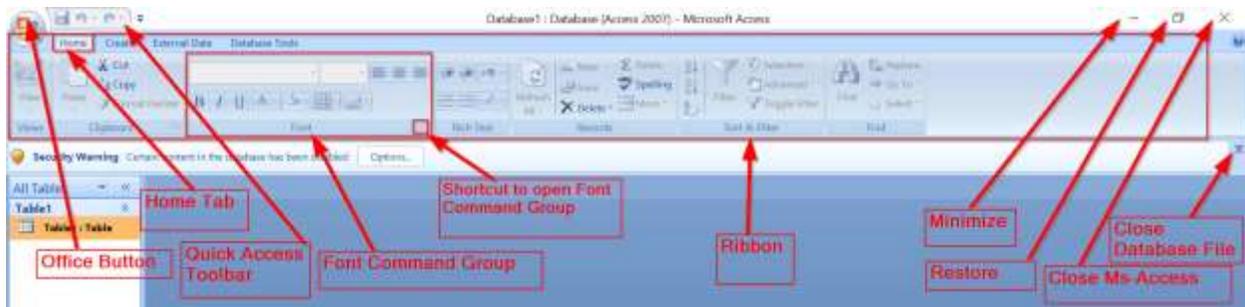
एमएस एक्सेस का ग्राफिकल यूजर इंटरफ़ेस / Graphical User Interface of Ms-Access:

एक्सेस 2007, 2010 और 2013 का का GUI (इंटरफ़ेस) काफी कुछ मिलता जुलता है। यह एक्सेस 2003 से काफी अलग है। हम यहाँ एम.एस.एक्सेस 2007 के इंटरफ़ेस पर चर्चा करेंगे। यह रिबन और क्विक एक्सेस टूलबार जैसी सुविधाओं का उपयोग करता है- जहाँ आपको बैकस्टेज व्यू में सामान्य कार्यों को करने के लिए कमांड मिलेंगे।



रिबन (The Ribbon) :

MS Access 2007 पारंपरिक मेनू के बजाय एक टैबड रिबन सिस्टम का उपयोग करता है। रिबन में कई टैब होते हैं, प्रत्येक में कमांड के कई समूह होते हैं। एक्सेस में सबसे आम कार्यों को करने के लिए आप इन टैब का उपयोग कर सकते हैं।



रिबन को कैसे मिनीमाइज और मैक्समाइज करें / How to minimize and maximize the Ribbon :

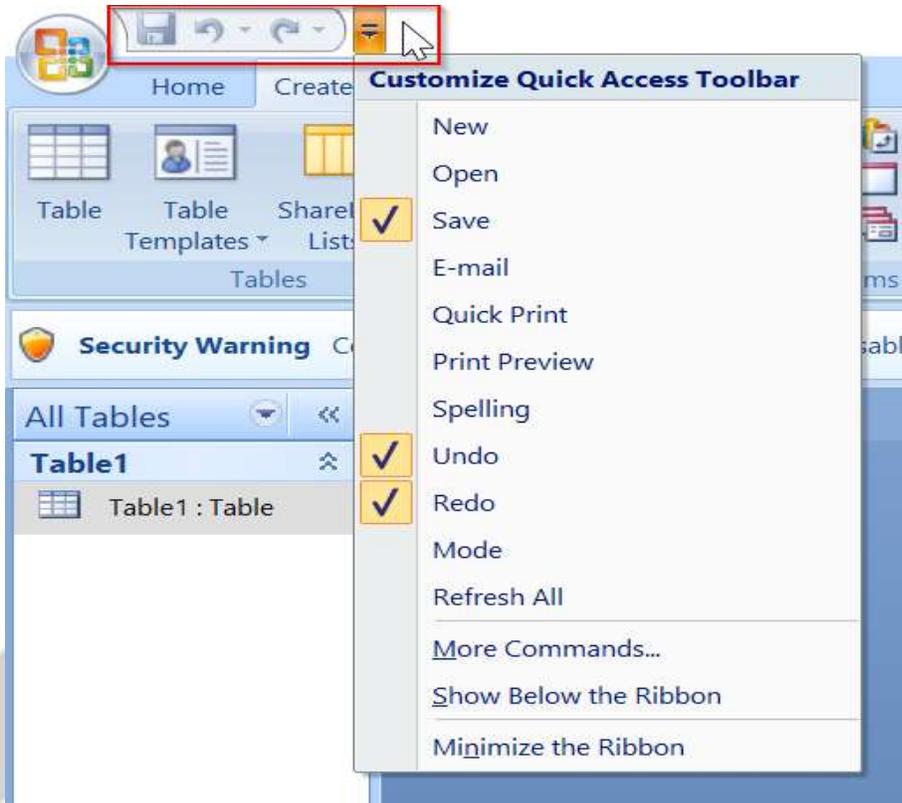
रिबन को आपके वर्तमान कार्य का जवाब देने के लिए डिज़ाइन किया गया है; हालांकि, यदि आप पाते हैं कि यह बहुत अधिक स्क्रीन स्पेस लेता है तो आप रिबन को Minimize कर सकते हैं।

इसे Minimize करने के लिए रिबन के निचले-दाएं कोने में तीर पर क्लिक करें। या किसी भी टैब पर डबल क्लिक करें। रिबन Minimize हो जाएगा। रिबन को फिर से दिखने के लिए एक टैब पर क्लिक करें। उपयोग में नहीं होने पर यह फिर से गायब हो जाएगा।

रिबन को Maximize करने के लिए, एक टैब पर क्लिक करें, फिर निचले-दाएं कोने में पिन आइकन पर क्लिक करें। या किसी भी टैब पर डबल क्लिक करें। रिबन हर समय दिखाई देगा।

The Quick Access toolbar

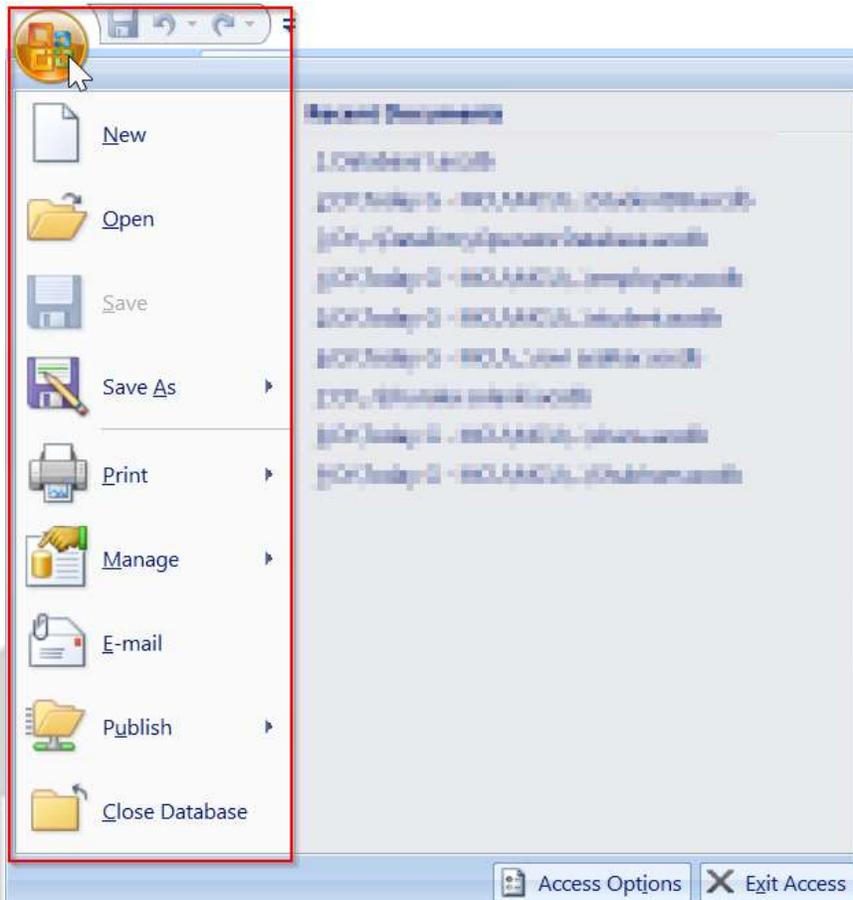
रिबन के ऊपर स्थित क्विक एक्सेस टूलबार, आपको सामान्य कमांड तक पहुंचने देता है इससे कोई फर्क नहीं पड़ता कि आप किस टैब पर हैं। डिफॉल्ट रूप से, यह Save, Undo और Repeat command दिखाता है। यदि आप चाहें, तो आप अतिरिक्त कमांड जोड़कर इसे कस्टमाइज कर सकते हैं।



Backstage view

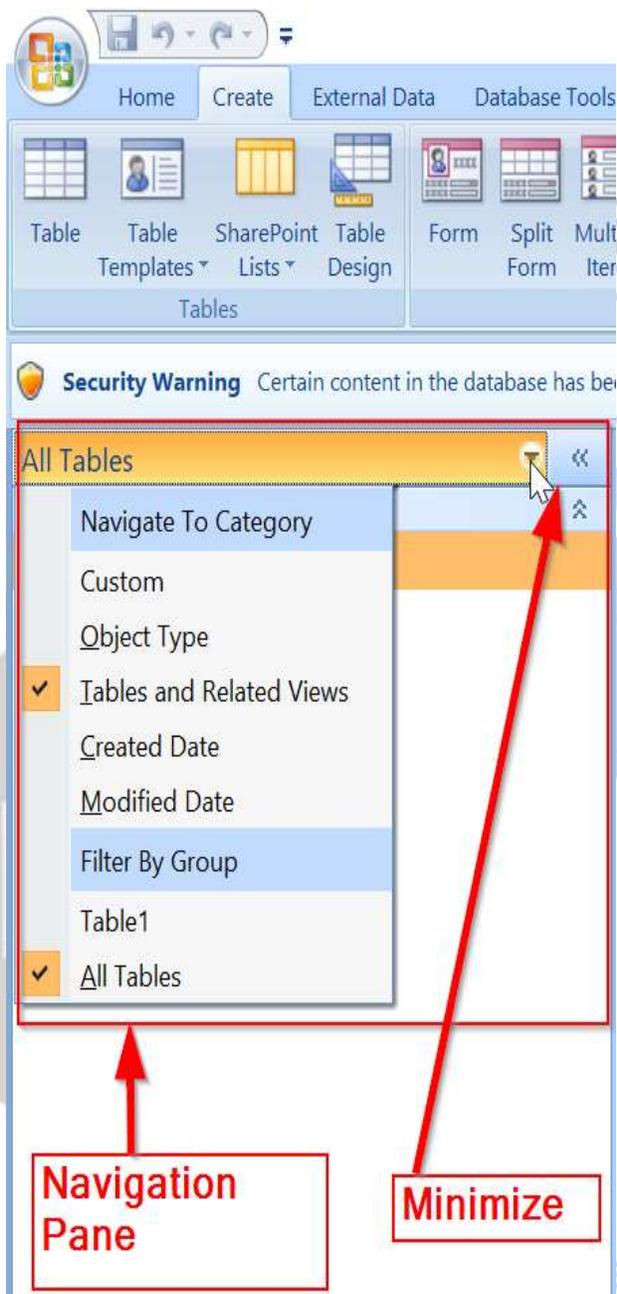
बैकस्टेज व्यू आपको अपने डेटाबेस को Save, Open और Print करने के लिए विभिन्न विकल्प देता है।

ऑफिस बटन पर क्लिक करें। Backstage view दिखाई देगा।



The Navigation Pane :

नेविगेशन पेन (फलक) एक list है जिसमें आपके डेटाबेस में प्रत्येक ऑब्जेक्ट होता है। आसान देखने के लिए, वस्तुओं को प्रकार के अनुसार समूह में व्यवस्थित किया जाता है। आप नेविगेशन पेन का उपयोग करके वस्तुओं को Open, Rename और Delete कर सकते हैं।



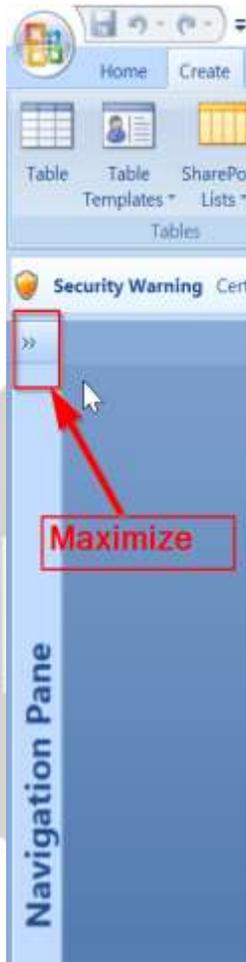
How to minimize and maximize the Navigation pane:

नेविगेशन पेन को मिनीमाइज कैसे करें :

नेविगेशन पेन (फलक) को आपके सभी ऑब्जेक्ट्स को प्रबंधित करने में आपकी सहायता के लिए डिज़ाइन किया गया है; हालांकि, अगर आपको लगता है कि यह आपकी स्क्रीन स्पेस को बहुत अधिक लेता है तो आप इसे Minimize (मिनीमाइज) कर सकते हैं।

नेविगेशन पेन को Minimize करने के लिए, ऊपरी-दाएं कोने में डबल तीर पर क्लिक करें।

नेविगेशन पेन को Minimize किया जाएगा। इसे Maximize (मैक्सिमाईज) करने के लिए दोबारा तीर पर फिर से क्लिक करें।



Introduction to MS Access Objects / एम.एस.एक्सेस ऑब्जेक्ट्स का परिचय :

Components of database management system / डेटाबेस मैनेजमेंट सिस्टम के घटक :

रिलेशनल डेटाबेस मैनेजमेंट सिस्टम के निम्नलिखित कंपोनेंट्स होते हैं जैसे टेबल, क्वेरी , फॉर्म , रिपोर्ट आदि । एम.एस-एक्सेस में इन्हें एक्सेस ऑब्जेक्ट कहा जाता है ।

1. **टेबल (Table)** : इस डेटाबेस में डाटा, टेबल्स (Tables) के रूप में संग्रहित और प्रदर्शित होता है। टेबल में रो (Row) और कॉलम (Column) होते हैं। एक रो में एक रिकॉर्ड संग्रहित किया जाता है। टेबल में एक रो एक एंटीटी (Entity) के रिकॉर्ड को दर्शाती है और कॉलम, एट्रीब्यूट्स (Attributes) / फील्ड्स (Fields) दर्शाते हैं। रिलेशनल डेटाबेस में टेबल को रिलेशन भी कहते हैं। एक रो और एक कॉलम जहां पर मिलते हैं उस बॉक्स को सेल (Cell) कहते हैं। सेल के अंदर डाटा प्रदर्शित किया जाता है। डेटाबेस का एक टेबल एक एंटीटी सेट (Entity Set) को प्रदर्शित करता है। टेबल की एक रो एक एंटीटी (Entity) या एक रिकॉर्ड को प्रदर्शित करती है। रिकॉर्ड को टपल भी कहा जाता है। टेबल का एक कॉलम, एक एट्रीब्यूट या एक फील्ड को प्रदर्शित करता है। एक डेटाबेस में कई टेबल हो सकते हैं। उदाहरण, किसी कॉलेज के सभी छात्रों की जानकारी का टेबल बनाना जिसमें छात्रों के रोल नंबर, नाम, कक्षा, मोबाइल नंबर, पूरा पता आदि हो।

Student				
rollno	sname	class	mobile	address
1	Raju Kumar Gu	DCA	9827000001	Khandwa
2	ShyamKumar	PGDCA	9827000002	Indore
3	Rahim Kumar	DCA	9827000002	Indore
4	Kishore Gupta	DCA	9827000003	Bhopal
5	Ramu	PGDCA	9827000004	Khandwa
6	Raj	PGDCA	9827000001	Khandwa
7	Ramu	DCA	9827000001	Khandwa
8	Raj	PGDCA	9827000004	Khandwa
9	Ravi Gupta	DCA	9827000003	Bhopal
10	Suresh	PGDCA	9827000003	Bhopal

2. **क्वेरी (Query)** : किसी टेबल या डेटाबेस से जरूरत के अनुसार डेटा निकालने के निर्देश (Command) को क्वेरी कहते हैं। डेटाबेस से पूछा गया प्रश्न क्वेरी कहलाता है। किसी क्वेरी के उत्तर में जो रिकॉर्ड डेटाबेस से निकाला जाता है उसे उस क्वेरी का डेटासेट या रिजल्टसेट (DataSet or ResultSet) कहते हैं। उदाहरण, इंदौर में रहने वाले सभी छात्रों के नाम और पिता के नाम की सूची निकालने की क्वेरी।

IndoreQueryWithJoin4

Field:	sname	fname	address	
Table:	Student	StudentFamily	Student	
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:			"indore"	
or:				

3. **फॉर्म** : यद्यपि आप टेबल्स में डेटा दर्ज और संशोधित कर सकते, लेकिन टेबल्स में डेटा को दर्ज करना तथा संशोधन करना आसान नहीं होता है क्योंकि टेबल में हजारों रिकॉर्ड हो सकते हैं। इस समस्या को दूर करने के लिए, फॉर्म प्रस्तुत किए जाते हैं। टेबल्स की तरह फार्म में भी डेटा दर्ज किया जाता है। फार्म में दर्ज किया गया डेटा अंततः टेबल में जाकर ही संग्रहित होता है। फार्म में डेटा दर्ज, संशोधित और प्रदर्शित भी किया जाता है। फॉर्म को

आकर्षक भी बनाया जा सकता है और इसमें डाटा वेलिडेशन भी लगाया जा सकता है ताकि टेबल में गलत डाटा संग्रहित न हो पाए ।

The screenshot shows a web application window titled "StudentForm1" with a sub-header "Student information Form 1". The date and time are "07 April 2020" and "02:42:40 AM". The form fields are as follows:

rollno:	2	DOB:	24-Sep-16
स्टूडेंट नाम:	ShyamKumar	Notes:	
class:	PGDCA	City:	Indore
mobile:	9827000002	percentage:	55
address:	Indore	AdmissionDate:	11-01-2020
aadhar_no:	123456781235		

A "Send" button is located at the bottom center of the form.

4. **रिपोर्ट्स** : क्वेरी द्वारा डेटाबेस या टेबल से लाए गये रिकॉर्ड को जब आप कागज पर प्रिंट करना चाहते हैं, तो उसे रिपोर्ट कहते हैं। जरूरत के अनुसार एक ही रिपोर्ट में कई टेबलों का डाटा रखा और प्रिंट किया जा सकता है । प्रिंट से पहले रिपोर्ट को आवश्यकता अनुसार फॉर्मेट भी किया जा सकता है और फ़ील्ड्स को व्यवस्थित भी किया जा सकता है । एक रिपोर्ट में एक या एक से ज्यादा पेज हो सकते हैं । डेटाबेस में संग्रहित विभिन्न प्रकार के डाटा का विश्लेषण करने के लिए भी रिपोर्ट का उपयोग किया जाता है ।

StudentForm2

Student 07 April 2020
02:43:37 AM

sname	Gender	rollno	DOB	class	mobile	address	aadhar_no	City
Ramu		7	14-Feb-19	DCA	9827000001	Khandwa	1234567812359	Khandwa
Raj	m	8	22-Aug-19	PGDCA	9827000004	Khandwa	1234567812359	Khandwa
Ravi Gupta	m	9	11-Sep-19	DCA	9827000003	Bhopal	1234567812357	Bhopal
Suresh	f	10	11-Sep-19	PGDCA	9827000003	Bhopal	1234567812357	Bhopal
Ramesh	f	11	11-Sep-19	DCA	9827000003	Bhopal	1234567812357	Bhopal
Uma	f	12	11-Sep-19	PGDCA	9827000003	Bhopal	1234567812357	Bhopal
Sudhir	M	13	24-Sep-16	PGDCA	9827000002	Indore	123456781235	Indore
Rani	M	14	24-Sep-16	DCA	9827000002	Indore	123456781235	Indore
Rajesh	m	15	22-Aug-19	PGDCA	9827000004	Khandwa	1234567812359	Khandwa
Raju Kumar Gupta	m	1	12-Jun-18	DCA	9827000001	Khandwa	123456781234	Khandwa

5. **मैक्रो (Macro)** : मैक्रो विभिन्न प्रकार के कार्यों को स्वचलित बनाने की सुविधा प्रदान करते हैं। आप मैक्रो का उपयोग किसी फॉर्म, रिपोर्ट या कंट्रोल में फंक्शनलिटी (Functionality) जोड़ने के लिए कर सकते हैं।

6. **मोड्यूल (Module)** : मैक्रो के समान ही मोड्यूल, विभिन्न प्रकार के कार्यों को स्वचलित बनाने की सुविधा प्रदान करते हैं। आप इनका उपयोग किसी फॉर्म, रिपोर्ट या कंट्रोल में फंक्शनलिटी (Functionality) जोड़ने के लिए कर सकते हैं। मैक्रो को बनाने के लिए मैक्रो एक्शन की सूची में से चुनाव करना होता है। जबकि मोड्यूल, विजुअल बेसिक फॉर एप्लीकेशन (Visual Basic for Applications) में लिखे जाते हैं।