अध्याय – 14

रिलेशनल डाटाबेस की अवधारणायें

रिलेशनल डाटाबेस में बहुत सी टेबलस होती है। हर टेबल का एक यूनिक नाम होता है। व उसमें बहुत से कॉलम (स्तंभ) होते है। हर स्तम्भ का भी एक यूनिक नाम होता है। रिलेशनल डाटाबेस का नाम मेथिमेटिक्स रिलेशन से निकाला गया हो क्योकि दोनों में निकट व्यवहारता है।

टेबल (रिलेशन) :– आर डी बी एम एस में डाटा एक प्रकार के डाटा बेस आबजेक्ट में स्टोर होता है। जिसे हम टेबल कहते है। दूसरे शब्दों में टेबल सम्बधित डाटा एन्ट्रीरीज का संग्रहण हो जिसमें की पक्तियाँ एवम् स्तंभ होते है। निम्नलिखित एक स्टूडेन्ट टेबिल का उदाहरण है।

Student Table

Roll_no	Name	Age	Address	class
101	Harish	10	Ajmer	5th
105	Kailash	20	kota	10th
109	Manish	18	Ahmadabad	9th
120	Ronak	14	Udaipur	8th
135	Shanker	13	Jaipur	7th

चित्र–1 स्टूडेन्ट टेबल रिलेशन

फिल्ड (Field) :– किसी टेबिल का Field उसका एक स्तंभ होता है। जो कि उस टेबल में किसी रिकार्ड की specific इर्फोमेशन को रखता है। जैसे की ऊपर दी हुईstudent टेबल मेंRoll_no,name, address और class फिल्ड्स है।

रिकार्ड :– को हम टेबिल की एक पक्ति भी कहते है। तथा यह एक टेबिल की वह individual entry है जो उस टेबिल में है। जैसे कि

Student table

105	**kailash**	**20**	**kota**	**10th**

स्तंभ :–किसी एक टेबिल की वह वर्टिकल एन्ट्री है जो किसी विशिष्ट फिल्ड से सम्बनधित सभी इन्फोमेशन रखता है। जो कि student टेबिल का एक स्तंभ Roll_no है। जो कि निम्न इन्फोमेशन रखता है।

Roll_no
101
105
109
120
135

डोमेन :– किसी फिल्ड की परमिटेड वैल्यू सैट को उसका डोमेन कहते है। उदाहरण स्वरूप field name के लिए डोमेन सभी नामों का सैट है।

डाटा बेस स्कीमा (Database schema) :–डाटा बेस स्कीमा किसी डाटा बेस की लॉजिकल डिजाईन है। जो कि शायद ही बदलती है। जैसे कि student टेबिल का schema है।

Student (Roll_no, name, age, address, class)

डाटा बेस इन्सटेन्स (Database instance) :–किसी डाटा बेस में समय के किसी भी क्षण डाटा के समूह को डाटा बेस इन्सटेन्स कहते है। उदाहरणार्थ निम्नलिखितStudent टेबिलStudent डाटाबेस का एक इन्सटेन्स है। जब हम टेबिल में कोई नई एन्ट्री करे या कुछ टेबिल से delete करेंतो यह किसी भी क्षण बदल सकता है।

Student

RollNo	Name	Age	Address	Class
101	Harish	10	Ajmer	5^{th}
105	Kailash	20	Kota	10^{th}
109	Manish	18	Ahmadabad	9^{th}
120	Ronak	14	Udaipur	8^{th}
135	Shanker	13	Jaipur	7th

प्राइमेरी की :–किसी टेबिल में एक या अधिक फिल्डस (attribute) का ऐसा set जो कि उस टेबिल की किसी भी पक्ति अथवा टपल्स को uniquely identify करता हो तो इस attributes के सैट को collectively लेने पर यह उस टेबिल की Primary key कहलाती है। जो एक प्रकार का constraints भी है। student टेबिल की primary key ,Roll_no फिल्ड है क्योकि student टेबिल में इसे फिल्ड के द्वारा सभी छात्रों को uniquely identify किया जा सकता है। एवंम् Roll_no फिल्ड की सहायता से किसी छात्र का रिकार्ड टेबिल से निकाला जा सकता है। जैसे कि अगर Roll_no फिल्ड की वैल्यू 105 लेने पर जो रिकार्ड टेबिल से निकलेगा वह छात्र kailash का होगा।

डाटा कन्सट्रेन्टस(Data constraints) :– किसी टेबिल के स्तंभों पे इस तरह के नियम लागू करना है जो उस टेबिल में डाटा की एन्ट्री की सीमा को निर्धारित करता है की उस टेबिल में केवल उसी प्रकार को डाटा एन्टर हो जो उस डाटाबेस कीconsistency, reliabilityएवंम् accuracy को सुनिश्चित कर सके। एवंम् डाटाबेस में किसी प्रकार का बदलाव जब अधिकृत डाटा बेस यूर्जस करे तब भी डाटाबेस में किसी प्रकार का डाटाconsistency loss ना हो।

डाटा कन्सट्रेन्टस कॉलम (स्तंभ) लेवल ओर टेबिल लेवल हो सकते है। कॉलम लेवल एवंम टेबिल लेवल कन्सट्रेन्टस में मुख्य अन्तर यह है कि कॉलम लेवल कन्सट्रेन्टस एक कॉलम में लगाये जाते है। जबकि टेबिल लेवल कन्सट्रेन्टस पूर्ण टेबिल में लगाये जाते है। डाटा कन्सट्रेन्टस के निम्न उदाहरण है जैसे कि

(1) studentकीClass nullनही हो सकती है।

(2) किन्ही दो छात्रों केRoll_noएक समान नही होगें।

(3) studentरिलेशन की हर एक Classरिलेशन में एकmatching class जरूर रहेगी ।

एक रिलेशन वाले कन्सट्रेन्टस

निम्नलिखित कन्सट्रेन्टस एक रिलेशन वाले कन्सट्रेन्टस है।

1) Not null

2) Unique

3) Check (<predicate>)

1) Not null कन्सट्रेन्टस :– यह कन्सट्रेन्टस किसी भी टेबिल में किसी फिल्ड या एट्रीब्यूट कीnullवेल्यू की एन्ट्री कोप्रतिबंधित करता है। अर्थात यदि किसी फिल्ड के लिए अगर यह कन्सट्रेन्टस लगा हुआ है। और उस टेबिल में कोई आपरेशनजो उस टेबिल में बदलाव कर अगर वेल्यूnullडालने की कोशिश करता है तो वहॉ पेerror जनरेट हो जाती है।

उदाहरण के तौर पर हमstudentटेबिल मेंclass एट्रीब्यूट की वेल्यू nullनहीं चाहते है इसी प्रकार Roll_noएट्रीब्यूटकी वेल्यू भीnullनही होनी चाहिए क्योकि वह उस टेबिल की एक प्राइमेरी की है।

2) Unique कन्सट्रेन्टस :–यह सुनिश्चित करता है कि कोई से दो टपल्स या पक्तियों किसी रिलेशन में सभी प्राइमेरी की एट्रीब्यूट पर बराबर नही हो सकती । अर्थात दोनों टपल्स के लिए हर एट्रीब्यूट की वेल्यू एक समान नही होगी।Unique कन्सट्रेन्टस केन्डीडेड की फोर्म करता है। जो कि किसी रिलेशन में एक से ज्यादा भी हो सकती है

3) Check कन्सट्रेन्टस :–Check कन्सट्रेन्टस को हम डोमेन एवंम् रिलेशनल दोनों डिक्लेरेशन पर लगा सकते है। जब किसी रिलेशन डिक्लेरेशन पर लगा हो तो सभी

टपल्सCheck क्लाज द्वारा specified condition को पूरा करेगा अर्थात्Check क्लाज यह सुनिश्चत करता है कि किसी स्तंभ की सभी वेल्यूज उस पर लगी शर्त को पूरा करेगा ।

उदाहरण स्वरूपstudent टेबिल में classफिल्ड की वेल्यू1^{st},2^{nd},3^{rd},----,9^{th},10^{th},11^{th},12^{th}होगी।

check (class in('1^{st}', '2^{nd}','3^{rd}', ----- '9^{th}','10^{th}','11^{th}','12^{th})

एंटिटी इन्टीग्रीटी कन्सट्रेन्टसः–यह सूनिश्चित करता है कि किसी भी टेबिल में कोई दो रिकार्डस या पक्तियों या टपल्स डूप्लीकेट नही हो सकते है। इसके अलावा वह फिल्ड जो प्रत्येक रिकार्ड की उस टेबिल में पहचान कर रहा है वह एक यूनिक फिल्ड है तथा इस फिल्ड की वेल्यू कभी भीnullनही होगी ।

एंटिटी इन्टीग्रीटी कन्सट्रेन्टस प्राइमेरी की के द्वारा लगाया जा सकता है। हर एंटिटी के लिए अगर हम प्राइमेरी की को परिभाषित करते है तो वह स्वतः ही एंटिटी इन्टीग्रीटी की पूर्ति करता है। उदाहरणार्थ

Student table

RollNo	Name	Address	Age	Class
110	Komal	jaipur	17	12th
120	Ronak	Udaipur	14	8th
105	kailash	kota	20	10th
107	hari	chittorgarh	10	5h

उक्तStudentटेबिल की प्राइमेरी की अगरRoll_noफिल्डस है तो इस फिल्ड में प्रत्येक छात्र काRoll_noअलग–अलग होगा। साथ ही किसी छात्र के लिए उसकी वेल्यू nullनही होगी अर्थात् सभी का अपना अलगRoll_noहोगाअलग –अलगRoll_noकी वजह से इस टेबिल में कोई दो पक्तियॉ एक समान नहीं होगी

रेफरेनशीयल इन्टीग्रीटी(Referential integrity) :–अगर हम किसी प्रकार से यह सुनिश्चित करना चाहते है कि किसी रिलेशन में कुछ ऐट्रीब्यूट के लिए उनकी वेल्यू वही हो जो किसी अन्य रिलेशन में कुछ ऐट्रीब्यूट के लिए है। अर्थात दोनों टेबिल में कुछ ऐट्रीब्यूट वेल्यू एक समान हो तो यह शर्त रेफरेनशीयल इन्टीग्रीटी कहलाती है। रेफरेनशीयल इन्टीग्रीटी की सुनिश्चितता फोरेन की के द्वारा की जा सकती है।

फोरेन की(Foreign key)इन्टीग्रीटी कन्सट्रेन्टसः– इस कन्सट्रेन्टस को समझने लिए निम्न प्रदर्शित टेबिल को उदाहरण के लिए हम लेते है। यहॉ दो टेबिल studentएवंमClassके नाम से है। एवंम् किसी क्षण उनमे एन्टर वेल्यूस भी प्रदर्शित है।

Student

RollNo	Name	Age	Address	Class
101	Harish	10	Ajmer	5^{th}
105	kailash	20	Kota	10^{th}
109	Manish	18	Ahmadabad	9^{th}
120	ronak	14	Udaipur	8^{th}
135	shanker	13	jaipur	7th

Classes

Class_name	Class_room	Strength
12^{th}	F-1	95
10^{th}	F-2	80
9^{th}	F-3	70
4th	F-4	110

चित्र –2 फोरेन की स्टूडेन्ट रिलेशन टेबिल

उक्त टेबिल Studentमें इस टेबिल कीप्राइमेरी की Roll_noफिल्ड है। जबकि Classesटेबिल की प्राइमेरी कीClass_name फिल्ड है। यहॉ पर हमने यह माना है कि सभी स्टूडेन्ड की एक हीclassविद्यमान है। जैसे की12^{th}की एकclass10^{th}की एकclass 9^{th} की एकclassइसी प्रकार अन्य ,अर्थात एक हीclassके sectionsअलग –अलग नही है। इसलिएclassesटेबिल की प्राइमेरी कीClass_name है।

Studentटेबिल अपने ऐट्रीब्यूट(Roll_no, address, age, name, class) के बीच में एक ऐसा ऐट्रीब्यूट भी रखता हैजो किसी अन्य टेबिल की प्राइमेरीkeyहै।उदाहरण स्वरूपStudentटेबिल मेंClassऐट्रीब्यूट टेबिल की प्राइमेरी की है। अतः Studentटेबिल में इस ऐट्रीब्यूटclassको हम इस टेबिल कीprimary keyकहेगे। जो कि टेबिल Classesको रेफर करेगी।

रिलेशन Studentटेबिल को हम रेफरेनशिंग रिलेशन कहेगें जब कि Classesको रेफरेनशड रिलेशन आफ फोरेन keyकहेगे।किसी ऐट्रीब्यूट को primary key होने के लिए उसका टाइप domain वही होना चाहिऐ जो दूसरे रिलेशन केऐट्रीब्यूट का है तथाforeign key में ऐट्रीब्यूटस की संख्या भी दूसरे रिलेशन के ऐट्रीब्यूटस के बराबर होनी चाहिए अर्थात वेकाम्पीटेबल होने चाहिए।

Foreign key के द्वारा हम निम्नलिखित सुनिश्चित कर सकते है–

(1) Classesटेबिल से हम कोई रिकार्ड तब तक नही हटा(delete)सकते जब तक इसकी रिलेटेड टेबिलStudent मेंमेचिंग रिकार्ड उपलब्ध है।

(2) आप Classes टेबिल में प्राइमेरीkeyकी वेल्यू को तब तक नही बदल सकते जब तक उस रिकार्ड से रिलेटेड रिकार्ड दूसरी टेबिल में उपलब्ध है।

(3) हम Studentटेबिल के Classफिल्ड में कोई नई वेल्यू जब तक नही डाल सकते जब तक वह वेल्यू Classesटेबिल केप्राइमेरी की फिल्ड(class_name) में उपलब्ध नही है।

(4) लेकिन आप foreign key field मेnull वेल्यू डाल सकते है। उक्त आपरेशनस के दौरान अगर हम चाहते है कि यह रिजेक्ट ना हो तो हमें classesका प्रयोगSQLमें करते है।

SQL का परिचयः–

स्ट्रकचर क्यूरी लेग्वेजSQLरिलेशनल ऐलजेबरा एवंम् रिलेशनल केलकूलस के कोमबीनेशन का उपयोग करती है।SQL आर डी बी एम एस की एक स्टेन्डर लेंग्वेज है जो किसी रिलेशनल डाटा बेस को ऑर्गेनाइज करने उसमें उपस्थित डाटा को प्रबंधन करने एवंम् रिलेशनल डाटा बेस में से डाटा रिट्रीवल के लिए उपयोग में लाई जाती है।

डी बी एम एस के विक्रेताओं जैसे कि ऑरेकलIBM, DB2, Sybase and Ingressअपने डाटाबेस के लिएSQLका एक प्रोग्रामिंग लेंग्वेज के तौर पर उपयोग करते है।इसका मूल संस्करण sequelकहलाता था जिसकोIBMने विकसित किया था।

इसके कई संस्करण होते है जैसेSQL-86, SQL89(extended standard), SQL92 and SQL1999 औरवर्तमानसंस्करणSQL-2003, SQLकेवल मात्र एक डाटाबेसqueryलेंग्वेज नहीं है ब्लकि अपने आप में एक स्टेन्र्ड है। जिसकेनिम्नलिखित भाग है।

(1) डेटा डेफीनेशन लेग्वेंज(Data Definition Language)

(2) डेटा मेनीपूलेशन लेग्वेंज(Data Manipulation Language)

(3) डेटा कन्ट्रोल लेग्वेज (Data Control Language)

ज्यादातर कामर्शियल रिलेशनल डाटा बेस जैसेIBM, Oracle, Microsoft, Sybase आदिSQLका उपयोगकरते है। सभी डाटाबेस SQLके सभी संस्करण में उपस्थित फिचर्स को सपोर्ट नहीं करते है अतः हमेशा अपने डाटाबेससिस्टम के अनुसार उसके संस्करण को देखते हुये ही फिचर्स का उपयोग करना चाहिए।

SQLके लाभः–SQLसभी डाटाबेस सिस्टम चाहे कामर्शियल(Oracle, IBM, DB2, Sybase) हो याऑपन सॉर्स(MySQL ,Postgres) के लिए लाभदायक है। इसके लाभ निम्नलिखित है।

(1) उच्च गति(**High speed**) :–इस प्रकार की लेंग्वेज है जो बडे से बडे ऑर्गेनाइजेशन के डाटाबेस से बहुत हीकुशलता(efficiently)के साथ तथा जल्दी –जल्दी डाटा निकाल सकती है अतः SQLएक उच्च गति की लेंग्वेज हैं।

(2) सीखने की सुविधा(**Easy to learn**):–SQLलेंग्वेज को सीखना बहुत ही आसान है। क्योंकि इसमें प्रोग्राम का किसी प्रकार कालंबा कोड नहीं होता हैं। अर्थात् इसमें ज्यादा कोडिंग की आवश्यकता नहीं होती हैं।

(3) अच्छी तरह से परिभाषित मानक लेंग्वेज(**Well defined standard**) :–SQL एक मानक(standard)भाषा हैजिसकों स्टेन्डराइजANSI & ISOने किया है।

डाटा डेफीनेशन लेंग्वेज(**Data Definition Language**) :–**DDL**,SQLका एक भाग हैं। जिसकी सहायता से हमडाटा बेस स्कीमा को स्पेसीफाई कर सकते है।

DDLकेवल डाटा बेस स्कीमा(schema)को हीspecifiedनहीं करती बल्कि हर रिलेशन के बारे में भी स्पेसीफीकेशनरखती है। कुछ निम्न है।

(1) हर रिलेशन का स्कीमा के लिए

(2) हर ऐट्रीब्यूट की वेल्यूज का डोमेन

(3) कन्सट्रेन्ट्र के लिए

(4) इनडेक्स के लिए

(5) किसी स्कीमा के ऑर्थोराइजेशन एंवम् सिक्योकिटी के लिए

(6) हर रिलेशन के फिजीकल स्टोरेज के लिए

SQL के बेसिक डोमेन टाइप्सः–

मुख्य रूप से उपयोग करने के लिएSQL के सभी विल्ट इन डोमेन टाइप्स निम्न होते है।

(1) **न्यूमेरिक डेटा टाइप्सः**– इसमें निम्न डेटा टाइप्स शामिल है।

(i)IntयाInteger:–बडें साईज के इन्टीजर्स के लिए है। इसके लिए 4 बाईट का स्टोरेज आवश्यक है।

(ii)Small Int:–छोटे साईज के इन्टीजर्स के लिए है। इसके लिए आवश्यक स्टोरेज बाईट 2 है।

(iii)Tiny Int:–अत्यधिक छोटे साईज के साईज या अनसाइन इन्टीजर्स के लिए

(iv) Float(M.D) :–फलोटिगं पाइन्ट नम्बरर्स के लिए । यह केवल साइन नम्बर्स के लिए है यहॉ पर Mउस नम्बर मेंकुल डिजिटस है तथाDडेसिबल नम्बर की संख्या है।

(v) Double (M,D) :–यह डेटा टाइप्स डबल प्रिसीजन वाले फलोटिंग पॉइन्ट नम्बर के लिए है तथा यह REALका समानार्थक शब्दहै।

(2) **स्ट्रींग टाइप्स (String Type)**:–MySQLमें निम्न स्ट्रींग डेटा टाइप्स है

(i)CHAR(C) :– यह डेटा टाइप फिक्सड लम्बाई की स्ट्रींग के लिए है। यहॉ परCइस स्ट्रींग की लम्बाई है जो यूजरदेता है। अगर इस लम्बाई से कम लम्बाई की कोई स्ट्रींग अगर स्टोर करते हैं तो बचे हुए में spaceस्टोर होता है।

(ii)VARCHAR(C):–यह वेरीयेबल लम्बाई की स्ट्रींग के लिए है यहॉ परCस्ट्रींग कीmaximum लम्बाई है जिसकोयूजर स्पेसीफाई करता है।

(3) DateऔरTimeडेटा टाइप्स :– MySQLमें निम्नलिखितTimeऔरDateडेटा टाइप्स है।

(i)Date:–इस डेटा टाइप्स में कोईDate,YYYY-MM-DDके प्रारूप में स्टोर होती है। तथा वह1000-01-01एवंम् 9999-12-31के मध्य हो सकती है।उदाहरण स्वरूपStudentटेबिल में किसी छात्र की ageअगर 1st july, 1980 हो तो यह1980-07-01 के प्रारूप मेंस्टोर होगी ।

(ii)DATETIME:– इस डेटा इाईप के द्वाराDate एंवमTimeके मेल को स्टोर कर सकते है। जिसका प्रारूपYYYY-MM-DD HH:MM:SSहै।यह Date एंवम्Time 1000-01-01 00:00:00 एंवम्9999-12-31 23:59:59 के मध्य हो सकता है। उदाहरणस्वरूप मध्यान्त2:35दिनाँक1st july, 1980 को1980-07-01 14:35:00इस तरह स्टोर कर सकते हैं।

(iii)Time:–यह समय कोHH:MM:SSके प्रारूप में स्टोर करता है।

(iv)Year:–यह वर्ष को2 डिजीट या 4 डिजीट के प्रारूप में स्टोर करता है।

डेटा मेंनीपुलेशन लेंग्वेज(Data Manipulation language):–यह SQLका वह भाग हैं।जिसे हम क्यूरी लेंग्वेज भी कहते है। अतः DMLएंवम् क्यूरी लेग्वेज समानार्थक शब्द है।DMLका उपयोग डाटा जो किरिलेशन में स्टोर है को मेनीपूलेशन (इन्सर्ट ,डिलीट ,अपडेट और रिट्रीवल) करने में करते है।

DML निम्नलिखित मेंनीपूलेशन कमाण्डस् रखता है।

(i) SELECT

(ii) UPDATE

(iii) INSERT

(iv) DELETE

डेटा कन्ट्रोल लेंग्वेज(Data Control Language):–डाटा कन्ट्रोल लेग्वेज SQLकी उपकमाण्डस् का संग्रह है जो डाटा बेस में डाटा की सुरक्षा तथा डाटा को मेंनीपूलेशन के अधिकारों से सम्बंधित है।

DCLमें निम्न कमाण्डस् आती है।

(i) COMMIT

(ii) ROLLBACK

(iii) GRANT

(iv) REVOKE

DDLरिलेडेट कमाण्डस एंवम् उनके सिन्टेक्स :– टेबिल या रिलेशन डिफाइन करने के लिएSQL में CREATE Tableकमाण्डस का प्रयोग किया जाता है। जिसका सिन्टेक्स निम्नानुसार है।

Table createकरने का Syntex :–MySQLमें एक टेबिल बनाने का generic syntaxहै

CREATE TABLE table_Name (F1 D1, F2 D2,…… ,Fn Dn<Integrity Constraints1,……<ICk>);

इस Syntexमें प्रत्येक टेबिल के फिल्ड या ऐट्रीब्यूट का नाम है। तथाDiप्रत्येकFi के अन्तर्गत आने वाली वेल्यूस का डोमेनटाइप है। इसके अलावा हम टेबिल पे कई तरह के कन्सट्रेन्ट लगा सकते है। जैसे कि PRIMARY KEY, FOREIGN KEY आदि है।उदाहरण स्वरूप अगर हम studentके नाम से कोई टेबिल बनानी हो जिसका schema है।

Student(Roll_No, Name, Age, Address, Class)तथा अन्य टेबिल Classesकाschema है।Classes(Class_Name , CRoomNo, Cstrength)अतःSQLमें इन टेबिल के लिए Syntexहोगा

CREATE TABLE Student (Roll_No int , Name CHAR(20), Age int Address VARCHAR(30), Class CHAR(10));

studentटेबिल पे अगर कन्सट्रेन्टस लगाने हो तोCREATE TABLE Student

(Roll_No int NOT NULL AUTO_INCREMENT, Name CHAR(20), Age int NOT NULL,Address VARCHAR(30), Class CHAR(10) NOT

NULL, primary key(Roll_no), foreign key(class) references classes(class_name));

यहाँ पर NOT NULLका प्रयोग उस ऐट्रीब्यूट की वेल्यूNULLना हो इसलिए किया गया हो अगरuserइस फिल्ड मेंNULLवेल्यू डालना चाहेगा तो SQL errarदेगा ।oreign keyकन्सट्रेन्टस का उपयोग ऐसे आपरेशनको रोकने के लिए है जो studentतथा Classesटेबिल में लिंक तोडते हैतथा AUTO_INCREMENTका प्रयोग Roll_noफिल्ड में वेल्यू को एक से आगे से बढाने के लिए किया जाता है। इसकी by defaultवेल्यू एक हाती है। अगर हम चाहते है कि यह sequenceकिसी और वेल्यू से प्रारम्भ हो तो हम इसSyntexका प्रयोग करते है।

ALTER table student AUTO_INCREMENT=100

Classesटेबिल के लिए Syntex

CREATE Table Classes

(Class_Name CHAR(10) NOT-NULL, CRoomNo CHAR(10),

PRIMARY KEY (Class_Name));

उक्त टेबिल को MySQL Promptकी सहायता से बना सकते है जैसे कि

root@host# MYSQL-u root -p

enter passsword : *****

MySQL> use School_Management;

यहॉ परuseकमाडण्स का उपयोग School_Managementके नाम से जो हम सर्वप्रथम डाटा कैसे बनायेगे।

उसमें प्रवेश केलिए करते है।डाटाबेस बनाने के लिए

MySQL> CREATE DATABASE School_Management;

तथा इसके बाद MySQL> use School_Management

MySQL> CREATE TABLE Student

(Roll_No Int NOT NULL AUTO_INCREMENT,Name CHAR(20), Age int NOT NULL,Address VARCHAR(30),Class CHAR(10) NOT NULL,PRIMARY KEY (Roll_No), FOREIGN KEY (Class) REFERENCES Classes(Class_Name));

-> Query ok, o row affected

MySQL>MySQL में किसी भी कमाडण्स को टर्मिनेट करने के लिए अन्त मेंसेमीकालन लगाते है। किसी टेबिल को डाटाबेस से हटाने का Syntexनिम्न है।

Generic syntax:–

DROP TABLE table_name;

Studentटेबिल को हटाने के लिए

DROP TABLE Student;

ALTER tableकमाडण्स :– इन कमाडण्स का उपयोग किसी टेबिल मे नया स्तंभ जोडने(add),delete करने याMODIFY करने में करते है। जिनके Syntexनिम्न है।

(1) नया स्तंभ(**column**)जोडने के लिए:–

ALTER TABLE table_name ADD column_name datatype;

डदाहणार्थClasses टेबिल में एक नयाcolumn *classes* strengthजोड. सकते है।

Alter table classes add class_ strength int

(2) कोई स्तंभ टेबिल को हटाने के लिए:–

ALTER TABLE table_name DROP COLUMN column_name;

(3) किसी स्तंभ का डेटा टाइप्स बदलने के लिए:–

ALTER TABLE table_name MODIFY column_name datatype ;

उदाहरण :–

ALTER TABLE Student MODIFY Age Date;

अन्य उदाहरण:–

CREATE TABLE Teacher (Tname VARCHAR(20), DOB Date, Salary FLOAT(5,2), Address VARCHAR(30), phone int PRIMARY KEY (Tname));

CREATE TABLE Teaches (Tname VARCHAR(20),Class_name CHAR(10),PRIMARY KEY(Tname, Class_name), foreign key(Tname) REFERENCES Teacher(Tname),

FOREIGN KEY(Class_Name)REFECENCES Classes (Classs_Name));

Checkकन्सट्रेन्ट Syntex:–

Checkकन्सट्रेन्टSyntexका उदाहरण निम्नलिखित है।

CREATE TABLE Student

(Roll_No int NOT NULL AUTO_INCREMENT,Name CHAR(20),Age int NOT NULL,Address VARCHAR(30),Class CHAR(10) NOT NULL,PRIMARY KEY(Roll_No),

Check(Class in ('1st', '2nd', '3rd', '4th', '5th', '6th', '7th', '8th', '9th', '10th', '11th', '12th')));

Create Index कमाण्डस :–यह कमाण्ड किसी टेबिल पर इन्डेक्स क्रिऐट करने के काम आती है। हम इन्डेक्स को नही देख सकतेपर यह टेबिल में डाटा तेजी से सर्च करने में मदद करता है इसका Syntexनिम्न है।

CREATE INDEX Index_name ON table_name (column_name)

CREATE INDEX Index on Student(Address)

डेटा मेनीपूलेशन कमाण्डस और उनके रिनट्रेक्स :–

SQL DMLक्माण्डस निम्नलिखित है।

(i) INSERT

(ii) DELETE

(iii) UPDATE

(iv) SELECT

(i) INSERTकमाण्ड :– CREATE TABLEकमाण्ड के उपयोग द्वारा किसी टेबिल को बनाने पर एक खाली टेबिलबनती हैं अर्थात उस टेबिल में किसी प्रकार की कोई वेल्यू या रिकार्ड या टप्लस नहीं होते है।किसी टेबिल में रिकार्ड या डाटा डालने के लिए हम SQL INSERT INTOकमाण्डस का उपयोग करते है।

MySQL Syntax :-

INSERT INTO
Table_Name(Column_Name1,Column_Name2,……,Column_Namen)V ALUES(Val e1,value2,……valuen);

स्ट्रींग टाइप्स के डाटा के लिए सभी वेल्यूज को सिंगल या डबल quotes (" ")में लेंगें।

MySQL कमाण्ड Promptसे डाटा का Insertion:–

उदारहण स्वरूप अगर हमेंStudentटेबिल में नई वेल्यूस या डाटा डालना है तो

MySQL> USE School_Management;

MySQL>INSERT INTO Student (Name, Age, Address, Class)

VALUES(“HARI”,15,”Chittorgarh”,”10th”);

इस उदाहरण में हमने Roll_Noनहीं लिया क्योकि टेबिल बनाते वक्त उसको हमने auto incrementदिया है। इसलिए MySQL automaticallyइसकी वेल्यू देगा। अन्य Syntex है।

INSERT INTO Student VALUES (“prakash”,18,”jaipur”,”12th”);

इनसर्ट के अन्यSyntexमें हम एक टपल्स को स्पेसीफइड करने के बजाय हम SELECTके उपयोग के द्वारा एक साथकई टप्लस इनर्सट कर सकते है।

(ii) SQL DELETEकमाण्डसः–DELETE कमाण्डस के द्वारा हम एक पूरा टप्लस DELETE करते है। इसके द्वाराहम किसी एट्रीब्यूट की वेल्यू को DELETEनहीं कर सकते है।

Syntax:-

DELETE FROM T ,WHERE P ;

यहाँ Tएक रिलेशन है जिसमें से टपल्स DELETEकरना है। तथा वह predicate (condition)है। जिसके अनुसारटपल्सDELETEहोगें।

DELETE FROM Student,WHERE Roll_No=105;

उक्तdelete statementके द्वारा हम वह टपल्सDELETEकरेगें जिस छात्रका Roll_No=105 है।

सारे टपल्स DELETEकरने के लिए हम निम्नSyntexलिखते है।

DELETE FROM Student ;

DELETE FROM Classes;

Class =10thके सभी छात्रों के डाटाDELETEकरने के लिए

DELETE FROM Student,WHERE Class =”10th”;

(iii) SQL UPDATEकमाण्डस :– UPDATE कमाण्ड का उपयोग हम किसी टपल्स के विशिष्ट एट्रीब्यूट की वेल्यू को बदलने के लिए करते है। अर्थात अगर हम पूरे टपल्स को नहीं बदलना चाहते है। केवल इसमें किसी ऐट्रीब्यूट की वेल्यू को ही बदलना चाहते है तो हम UPDATEकमाण्ड का उपयोग करते है। जिसका Syntex निम्न है।

Syntax:

UPDATE table_name SET first_field=value1, second_field=value2

[WHERE clause];

हम एक साथ कई फिल्ड की वेल्यू कोUPDATEकर सकते है।

उदाहरण

MySQL> UPDATE Student, SET Class="11th",WHERE Roll_No=12;

इस statementके द्वारा हम उस छात्र की classको बदलना चाह रहे है जिसकाroll_no 12 है। उसकी class को हम 10^{th}कीबजाय 11^{th}कर रहे है।

छात्र का**address**बदलना

UPDATE Student ,SET Address="Ajmer",WHERE Roll_No=102;

Student

RollNo	Name	Age	Address	Class
11	Hari	15	Jaipur	10th
101	Prakash	16	Kota	11th
102	Basu	11	Udaipur	6th
120	Viveka	9	Jaipur	4th

उक्तupdate statement के बादstudent table का instances निम्नलिखित होगा ।

Roll_No	Name	Age	Address	Class
11	Hari	15	Jaipur	10th
101	Prakash	16	Kota	11 th
102	Basu	11	Ajmer	6 th
120	Viveka	9	Jaipur	4 th

(iv) SQL SELECT statement

किसीSQL query expression में मुख्य रूप से निम्नलिखित तीन क्लॉजेसclauses होते है।अर्थात कोई भीSQL query जो हम रिलेशन डाटा बेस के लिए लिखते है। उसकी बुनियादी संरचना(basic structure)मेंउक्त 3 क्लाज होगें।

- SELECTक्लॉज में हम उन ऐट्रीब्यूटस को लिखते जो हमें हमारे आउट पुट रिलेशन में चाहिए।

- FROM क्लॉज में हम उन रिलेशन को लिखते है। जिनको हमेंquery expressionमें उपयोग करना है।FROMक्लाज मे लिखे रिलेशनस का Cartesian productहोता है।
- WHEREक्लॉज में हमpredicate लिखते है। जोFROM clause के रिलशनस के ऐट्रीब्यूटस को लिप्त रखता है।अर्थात जिसकीBoolean वेल्यू(true or false) होती है।

SQL query का निम्न फोर्म (form)होता है।

SELECT AT1, AT2, AT3 , …,ATn,

FROM r1, r2, r3,…rrn,

WHERE P;

यहाँATi एक ऐट्रीब्यूट को प्रदर्शित करता है। और r_i एक रिलेशन को ।P एक predicateहै।

SELECT clause:SELECTक्लाज उन ऐट्रीब्यूट की लिस्ट रखता हैं जो उस रिलेशन से retrieveकरने है।

उदाहरण :–SELECTक्लाज को समझने के लिए हमSchool Management डाटा बेस में से एक टेबिल को लेते है। जोकि निम्नलिखित हैं–

Roll_No	Name	Age	Address	Class
101	Bhagat singh	20	Ajmer	12th
105	Chandra	17	Kota	11 th
12	Shekhar	16	Jaipur	10th
17	DinDayal	15	Udaipur	9 th
90	Rohit	9	Ajmer	5th

SELECTकमाण्ड काSyntaxहैं।

SELECT field_names, FROM realtion_names;

Field namesमें एक साथ कई फिल्डfetchकर सकते है। इसके अलावा हम स्टार(*) भी फिल्ड के नाम के बजायलगा सकते हैं तबSELECTफिल्डFROM क्लाज के रिलेशन के सारे फिल्ड Returnकरेगा ।

उदाहरण

SELECT Roll_No, Age,
FROM Student;

रिजल्ट

RollNo	Age
101	20
105	17
12	16
17	15
90	9

SELECT * FROM Student ;

Output->

RollNo	Name	Age	Address	Class
101	Bhagat	20	Ajmer	12th
105	Chandra	17	Kota	11 th
12	Shekhar	16	Jaipur	10th
17	DeenDayal	15	Udaipur	9 th
90	Rohit	9	Ajmer	5th

Duplicateफिल्ड वेल्यूज को हटाने के लिए SELECT में DISTINCT keywordका उपयोग करते है।

SELECT DISTINCT AddressFROM Student ;

यहाँ Studentटेबिल केAddressफिल्ड को fetchकरने पर उसमें Duplicateवेल्यू Ajmer एक ही बार आयेगी ।

रिजल्ट

Address
Ajmer

Kota
Jaipur
Udaipur

SELECTक्लाज में हमColumn के नाम के साथ साथ निम्नलिखित अर्थमेटिक एक्सप्रेशनभी उल्लेखितकर सकते है।

Description	Operator
Addition	+
Subtraction	-
Division	/
Multiplication	*

डदाहरण के लिए हम Teacherटेबिल को लेते है।

Teacher

Tname	Address	Salary	Phoneno
Radha krishan	Kerla	5000	1234567899
Lalaji	Udaipur	750	9413962123 9999999999
Sarvopalli	Rampur	2000	9312171080 3333333333
leadgevan	Maharastra	9000	5189310510 2121212121

SELECT Tname, Salary *10,

FROM Teacher;

इसqueryको रिजल्ट मेंSalary ऐट्रीब्यूट की वेल्यूज 10 से मल्टीपलाई हो कर मिलेगी।

रिजल्ट

Tname	Salary*10
Radha krishnan	50,000
Lalaji	7500

Sarvopalli	20,000
Leadgevan	90,000

SELECT statement का उपयोग WHEREक्लाज के साथ :–WHEREक्लाज मे एक या ज्यादा कन्डीशन्स को लिखते है। जब यह शर्त (condition)संतुष्ट होगी तभी रिलेशन से कोई पंक्ति निकाल सकते है।

उदाहरण स्वरूप Teacherटेबिल के लिए निम्न queryलिखते है।

SELECT Tname, Salary,FROM TeacherWHERE Salary>2000;

तो निम्नResultप्राप्त होगा ।

Tname	Salary
Radha krishnan	5000
Harivansh	9000

WHERE clauseनिम्नलिखित लॉजिकलconnectionsउपयोग करता है।

(i) AND

(ii) OR

(iii) NOT

WHEREक्लाज कम्पेरिजन (Comparision)आपरेटर्स भी उपयोग करता है।

Description	Operators
Less than	<
Less than or equal to	<=
Greater than or equal to	>=
Greater than	>
Equal to	=
Not equal to	!= or <>

उदाहरण :–

SELECT Tname,FROM Teacher,WHERE Salary>2000 AND Address=”Kerala”;

रिजल्ट

Tname
Radha krishnan

(3) डाटा कन्ट्रोल लेंग्वेज कमाण्डसor DCLकमाण्डस :–DCL कमाण्डस डाटाबेस की सुरक्षा से सम्बन्धित है । यहकमाण्डस यूजर्स को विशेषाधिकार प्रदान करती है। ताकि यूजर्स डाटाबेस में कुछ आब्जेक्टस को accessकर सके ।

SQL GRANTकमाण्डस :– रिलेशनल डाटाबेस में एक यूजर के द्वारा बनाये(create)गये आब्जेक्टस कोदूसरे यूजर्स जब तक नही पहुँच(access)सकते जब तक उनको बनाने वाला यूजर्स दूसरे यूजर्स को इसकी सहमती नहीं देता। यह सहमती(permission)GRANTकमाण्ड के उपयोग द्वारा दी जा सकती हैं।

GRANT statement कई तरह के विशेषाधिकार कई आब्जेक्टस टेबिल व्यूपर प्रदान करता हैं ।

Syntax:-

GRANT [type of permission] ON [database name] . [table_name] TO'[user name] '@'localhost'; identified by password;

यहाँपर type of permissionनिम्नलिखित होती है।

- CREATE- नई टेबिल या डाटाबेस बनाने की सहमती देता है।
- DROP . टेबिल या डाटाबेस को हटाने(delete)की सहमती देता है।
- DELETE – टेबिल से पक्ति या टापल्स हटाने(delete)की सहमती देता हैं
- INSERT- टेबिल में पक्ति या टापल्स जोडनें(insert) की सहमती देता है।
- SELECT – इस कमाण्डस के द्वारा टेबिल या डाटाबेस को(read)करने की सहमती देता है।
- UPDATE. टेबिल की पक्तियों कोupdate की सहमती देता हैं

उपर दिये syntaxमें हम डाटा बेस या टेबिल के नाम के स्थान पर अगरasterisk (*) लगाते है। तो यहsyntaxकोई भीडाटाबेस या टेबिल के पहुँच(access)की सहमती(permission)देगा।

Syntax:-

GRANT ALL Privileges ON *.* TO 'new_user @ 'localhost';

यहाँasterisk (*) डाटाबेस व टेबिल को बताता हैं। यह कमाण्ड यूजर कोread, edit ,execute और सभी आपरेशनस कीसहमती सभी डाटाबेस और टेबिल के लिए देता है।

उदाहरण :– MySQL मेंGRANT कमाण्डस को समझते है।

नया यूजर बनानाः–

MySQL में Defaultयूजर्स Rootहोता है। जिसकी सभी डाटाबेस पर फुल पहुँच (full access)होता है। MySQLमें नया यूजर बनाने को syntax है।

Syntax

MySQL> CREATE USER 'new_user' @'localhost'

IDENTIFIED BY 'Password';

उदाहरण –MySQL> CREATE USER '14EEACS350' @'localhost' IDENTIFIED BY '123456';

उदाहरण :–उक्त कमाण्ड द्वारा एक नयायूजर '14EEACS350' के नाम से बनेगा जिसका पासवर्ड'123456' होगा।हॉलाकि इस यूजर को डाटा बेस के साथ कुछ भी करने की सहमती नही है। अभी तक तो इस नये यूजर कोMySQLमें लॉगिनभी नहीं हो सकता है। इसलिए सर्वप्रथम नये यूजर कोसहमती देनी आवश्यक है।

उदाहरण :–GRANT ALL PRIVILEGES ON

School_Management.* TO '14EEACS350'@'localhost'

IDENTIFIED BY '123456';

इस कमाण्ड के द्वारा नये यूजर(14EEACS350)को School Managementडाटाबेस की सभी टेबिल पर सभी पहुँच(access)प्राप्त होगें।एक बार यूजर के लिए सहमतीfinalहोने के बाद निम्न कमाण्ड चलाये

FLUSH PRIVILEGES;

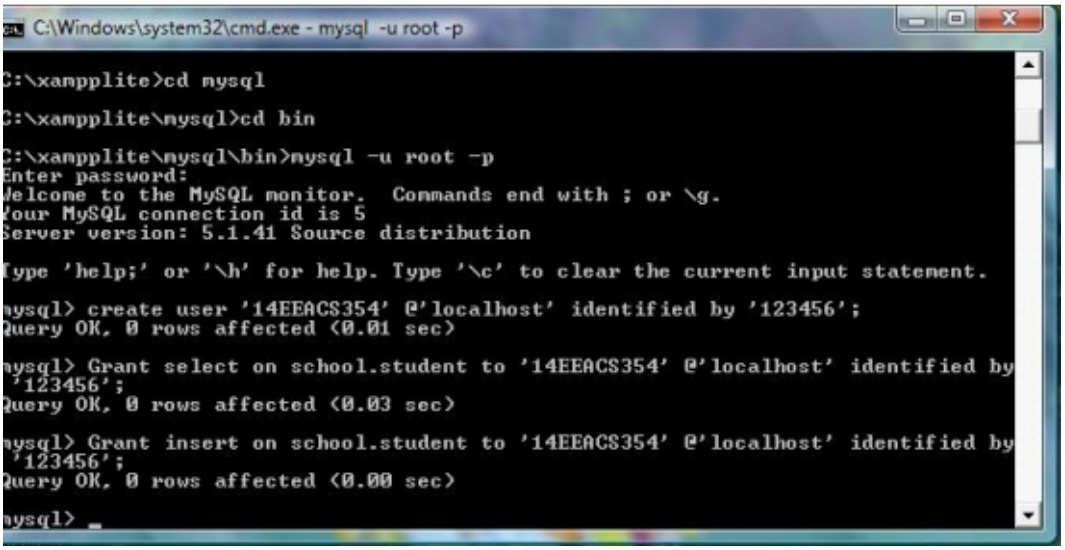
तकि सभी बदलाव प्रभावी हो सके। अगर हम सारे विशेषाधिकार'14EEACS350' यूजर को नही देना चाहते है। तो केवलreadके अधिकार के लिए

GRANT SELECT ON School_Management.Student TO '14EEACS350' @ 'localhost' IDENTIFIED BY '123456';

कमाण्ड चलाये इस कमाण्ड द्वारा 14EEACS350' को केवल देखने का अधिकार वो भी केवल Student टेबिल को प्राप्त होगा।Student टेबिल में INSERTके विशेषधिकार के लिए

GRANT INSERT ON School_Management.Student TO '14EEACS350' @ 'localhost' IDENTIFIED BY '123456';

कमाण्ड का उपयोग करे । अन्य विशेषधिकार भी प्रदान किये जा सकते है।



चित्र–3

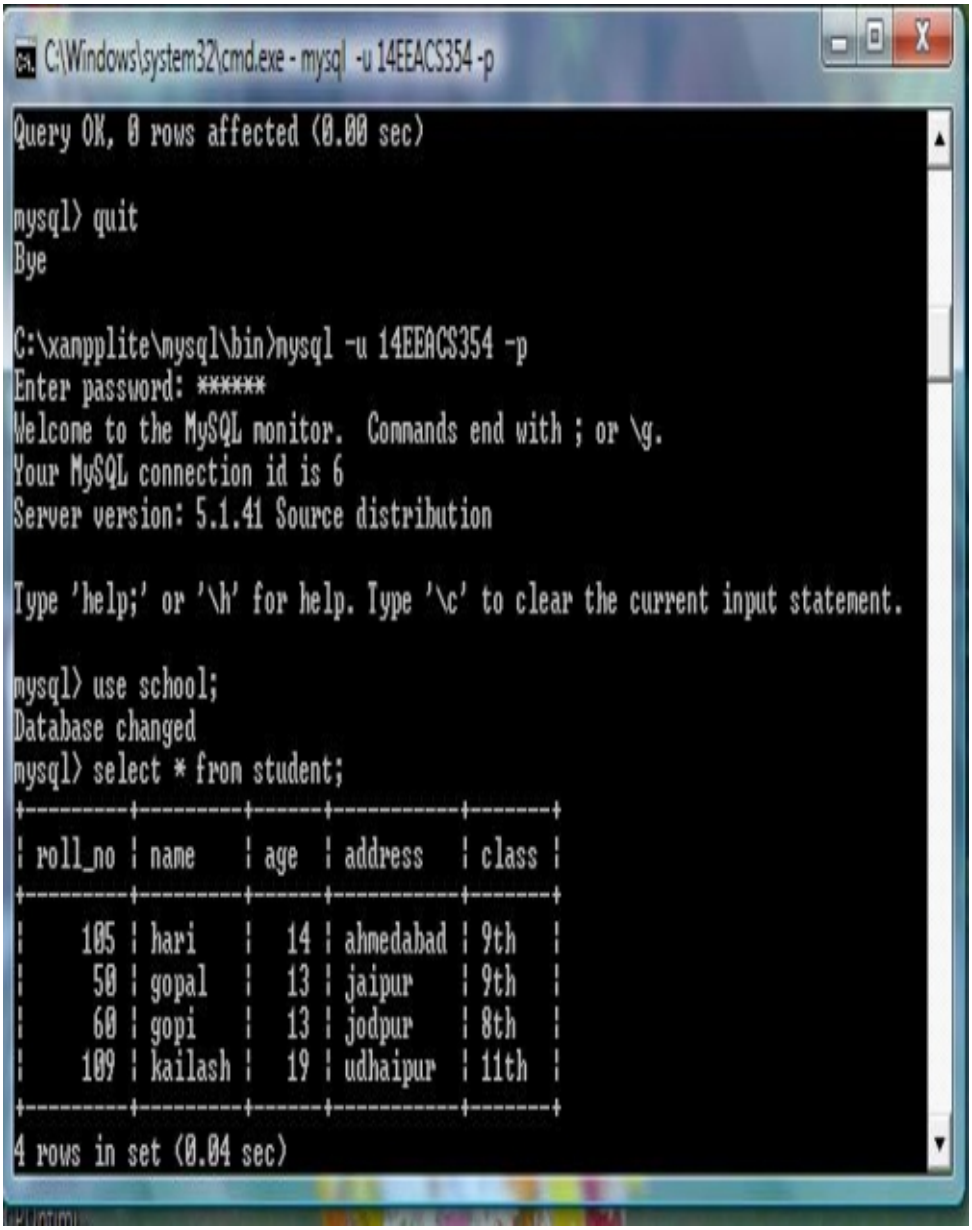
क्योकि हमने केवलStudent टेबिल पर14EEACS350' यूजर को केवल select एवंम्insert विशेषाधिकार प्रदानकिया है। अतः यह यूजरStudentटेबिल में कोई बदलाव(update)अथवा हटाना(delete)कमाण्ड नहीं चलसकता है।इसको देखने के लिए हम एक बार Quitकमाण्ड द्वारा logoutहाते है। एवम् नये यूजर के लिए निम्नलिखित कमाण्ड द्वारा वापिस login करते है।

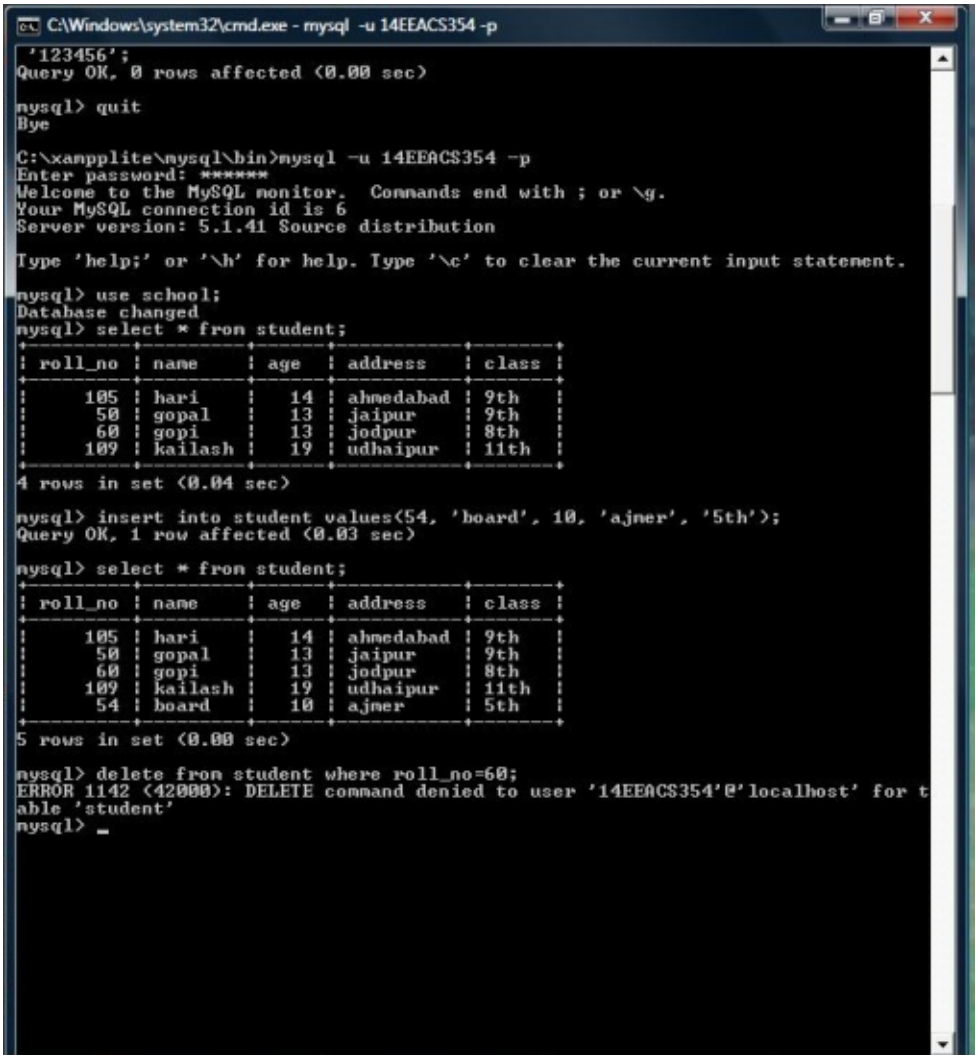
MySQL-u[new username]-P;

MySQL> MySQL-u 14EEACS350 -P;

Enter password :123456;

यहाँStudentटेबिल पर भिन्न भिन्न कमाण्ड चलाते है।जिनके screen shots है।


C:\Windows\system32\cmd.exe - mysql -u 14EEACS354 -p
Query OK, 0 rows affected (0.00 sec)
mysql> quit
Bye
C:\xampplite\mysql\bin>mysql -u 14EEACS354 -p
Enter password: ******
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 6
Server version: 5.1.41 Source distribution
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> use school;
Database changed
mysql> select * from student;
roll_no	name	age	address	class
105	hari	14	ahmedabad	9th
50	gopal	13	jaipur	9th
60	gopi	13	jodpur	8th
109	kailash	19	udhaipur	11th
4 rows in set (0.04 sec)



2 REVOKEकमाण्ड :–

REVOKEकमाण्डस के द्वारा किसी आब्जेक्ट (टेबिल) से सहमती वापिस लेने के लिए करते है। जिसकीSyntax GRANTके समान ही है।

Syntax:-

REVOKE [type of permission] ON [database_name].

[table_name] FROM '[user name] '@'localhost';

उदाहरण के तौर पर हमDROP के द्वारा किसी यूजर को हटाने(delete)करने के लिए कर सकते है।

उदाहरण :–DROP USER '14EEACS350' @'localhost';

REVOKE DELETE ON Student FROM 14EEACS350;

(3)COMMITकमाण्डः–COMMITकमाण्ड का उपयोग सभी प्रकार के जो बदलाव डाटाबेस पे किये गये है। उनबदलावों के डाटाबेस में स्थायी करने के लिए किया जाता है।MySQL में हर समय auto commit mode enabledहोता है। जिसका अर्थ कि जैसे ही हम कोईupdate कमाण्डसके द्वारा टेबिल कोmodfiyकरते है। तो यह बदलावMySQL मेंDISK पर होता है। जिससे की यह स्थायी हो सके ।

START TRANSACTIONके द्वारा हम auto commit को disabledकर सकते है।

SQL आपरेटर्स का परिचय :–SQL आपरेटर्स एक प्रकार के रिजर्व (सुरक्षित) शब्द है। जो किसी SQLक्यूरी के WHEREक्लाज में उपयोग किये जाते है। जिनमें मुख्य ऑपरेशन निम्न है।

(i) कम्पेरिजन(Comparison)

(ii) अर्थमेटिक(Arithmetic)

(iii) लोजिकल(Logical)

(iv) शर्त को निगेट करने के कार्य मे आने वाल आपरेटर्स

(i) कम्पेरिजन आपरेटर्स :– आपरेटर्स विवरण निम्नलिखित है।

आपरेटर्स	विवरण
(=)	यह आपरेटर्स दो आपरेटर्स की वेल्यू की समानता या असमानता को चेक करता है। समान होने पर शर्तtrueहोगी।
(<> or !=)	यह आपरेटर्स दो आपरेटर्स की वेल्यू की समानता या असमानता को चेक करता है। अगर वेल्यू समान नही है तो शर्त trueहोगी।
(>)	अगर बॉये तरफ के आपरेन्ड की वेल्यू दॉये तरफ के आपरेन्ड से ज्यादा हो तो शर्त trueहोगी।
(<)	अगर बॉये तरफ के आपरेन्ड की वेल्यू दॉये तरफ के आपरेन्ड से कम हो तो शर्त trueहोगी।
(>=)	अगर बॉये तरफ के आपरेन्ड की वेल्यू दॉये तरफ के आपरेन्ड से ज्यादा या समान हो तोशर्त true होगी।
(<=)	अगर बॉये तरफ के आपरेन्ड की वेल्यू दॉये तरफ के आपरेन्ड से कम या समान हो तो शर्त trueहोगी।

(ii) SQLअर्थमेटिक आपरेटर्स :– आपरेटर्स विवरण निम्नलिखित है।

आपरेटर्स	विवरण
(+)	आपरेटर्स की दोनों आपरेन्ड की वेल्यू को जोडता (add)है।
(-)	बॉयी तरफ के आपरेन्ड की वेल्यू में से दाँयी तरफ की आपरेन्ड की वेल्यू का घटाव (subtract)करता है।
(*)	दोनों तरफ की आपरेन्ड की वेल्यू को गुणा करना ।

(% or मॉडूलस) बाँयी तरफ के आपरेन्ड की वेल्यू को दॉयी तरफ के आपरेन्ड की वेल्यू से भाग(divide)देता है। तथाउनका शेष रिटर्न करता है।

SQLअर्थमेटिक व कम्पेरिजन आपरेटर्स को समझने लिए हम एक बार नीचे दी गईStudent टेबिल, Teacherटेबिलऔर Classesटेबिल का उदाहरण लेते है।

Student

Roll_No	Name	Age	Address	Class
109	Omprakash	9	Jodhpur	4th
111	Prakash	15	Jaipur	10 th
91	Suman	11	Jaipur	6th
75	Shanker	13	Ajmer	8th

Teacher

Tname	Address	Salary	PhoneNo
Radha krishnan	Kerla	3000	1234567899
Rajesh	Jodhpur	5000	1111162123
Lalaji	Ajmer	9000	1111171080
Hariom	Kerla	40000	1111110510

Classes

Class_name	CroomNo	CStrength
12th	F-1	90
6th	F-17	65
9th	S-21	110

उदारहण 1

SELECT * FROM Student,WHERE Age=15;

Output ->

Roll_No	Name	Age	Address	Class
111	Prakash	15	Jaipur	10 th

क्योकिStudentटेबिल में केवल एक ही पक्ति ऐसी है। जिसमें age15 हैं।

उदारहण 2

SELECT * FROM Student, WHERE Age>11;

Output ->

Roll_No	Name	Age	Address	Class
111	Prakash	15	Jaipur	10 th
75	Shanker	13	Ajmer	8th

उदारहण 3

SELECT * FROM Student, WHERE Age <= 13;

Output ->

Roll_No	Name	Age	Address	Class
109	Omprakash	9	Jodhpur	4th
91	Suman	11	Jaipur	6th
75	Shanker	13	Ajmer	8th

(iii) SQLलॉजिकल आपरेटर्स :– आपरेटर्स विवरण निम्नलिखित है।

आपरेटर्स	विवरण
(AND)	WHERE क्लाज में विभिन्न शर्तो को अलाऊ करता है। अर्थातa AND bदोंनों शर्तो aएंवम् b trueहोने परइसका परिणामtrueहोगा।
(OR)	यह WHEREक्लाज में विभिन्न शर्तो को संयुक्त करता है। अर्थात a OR bमें कोई भी शर्त aया bके trueहोनेपर इसका परिणाम trueहोगा।

उदाहरण के लिए हमTeacher टेबिल को लेते है।

(i) SELECT Tname, SalaryFROM Teacher WHERE Salary <= 5000 and Salary >=4000;

उक्त क्यूरी मेंSalary <= 5000 के लिए दो पक्ति में वेल्यू5000 से कम एव समान है। अतः शर्त trueतथा Salary >=4000 के लिए 3 पक्ति में वेल्यूस 4000 से अधिक है। पर दोनों शर्त केवल एक पक्ति के लिएtrue है। अतः इसकापरिणाम होगा।

Tname	Salary
Rajesh	5000

(ii) SELECT Tname, Salary,FROM Teacher,WHERE Salary <= 5000 OR Address = "Kerla";

Output ->

Tname	Salary
Radha krishnan	3000
Rajesh	5000
Hariom	40000

उक्त परिणाम में तीन पक्तिया प्राप्त हुए है। क्योंकि <=5000 के लिए दो पक्ति में वेल्यू इसके कम या समान है। जबकी address=" Kerla" भी दों पक्तियों के लिएtrue है। हॉलाकि एक पक्ति मेंSalaryकी वेल्यू 5000से अधिक पर यह शर्त यहाँ cheek नही होगी अतः परिणाम उक्त प्राप्त होगा क्योकि ORदोनो शर्तो को संयुक्त कर परिणाम देगा ।

आपरेटर विवरण

Not(!) यह आपरेटर आपरेन्ड की वेल्यू को इन्वर्स करता है।

उदाहरण :– SELECT *FROM StudentWHERE !(Roll_No= 111);

Output ->

RollNo	Name	Age	Address	Class
109	Omprakash	9	Jodhpur	4th
91	Suman	11	Jaipur	6th
75	Shanker	13	Ajmer	8th

यहRollNo 111 के अलावा बाकी सभीStudent की सूचना देगा

आपरेटर विवरण

BETWEEN यह आपरेटर्स वेल्यूस के मध्य वेल्यूस को सर्च करता है।

उदाहरण :–

SELECT Tname , SalaryFROM TeacherWHERE Salary BETWEEN 3000 and 5000;

Output ->

Tname	Salary
Radha krishnan	3000
Rajesh	5000

आपरेटर विवरण

ANY इस आपरेटर के द्वारा हम वेल्यू को किसी लिस्ट में अन्य वेल्यू से कम्पेयर (compare)करने काम में लेतें है।

Syntax:- ANY आपरेटर के निम्नलिखितsyntaxहो सकते है।

<ANY, <=ANY, >=ANY, =ANY and <>ANY

उदाहरण :–अगर हमें किसी वेल्यू 5 को लिस्टसे compareकरना हो तो ANYके द्वारा इसcomparision कापरिणाम निम्नानुसार होगा ।

Expression	परिणाम	टिप्पणी
(1)	5 >ANY {0, 5 ,6}True	क्योंकि यहाँ 5, 0 से बड़ा है।
(2)	5 =ANY {0, 5 , 6} True	क्योंकि यहाँ 5, 5 से समान है।
(3)	5 <ANY{0, 5, 6}Trueक्योंकि यहाँ 5, 6 से छोटा है।	
(4)	5 >ANY{5, 6} False नम्बर से बड़ा नही	क्योंकि यहाँ 5 किसी भी

अर्थात ANYआपरेटर कम से कम एक के लिए शर्त सही होने परTrueदेता है।

आपरेटर विवरण

ALL इस आपरेटर के द्वारा हम एक वेल्यू को किसी अन्य लिस्ट की सभी वेल्यूस से(compare)करने के काम लेतेहै।

अर्थातALL आपरेटर लिस्ट की सभी वेल्यूस के लिए शर्त सही होने पर हीtrue returnकरता है।

उदाहरण :–

	Expression परिणाम		टिप्पणी
(1)	5<ALL{5,6}	False	यहाँ 5 केवल 6 से ही छोटा है।
(2)	5>ALL{0, 3, 4}	True	यहाँ 5 सभी से बड़ा है।
(3)	5<=ALL{5, 6}	True	क्योंकि यहाँ 5ए 6 से छोटा है। तथा 5 के समान है।

उक्त आपरेटर का उपयोग हम आगेsubqueries लिखने में करेंगें

आपरेटर	विवरण
LIKE	इस आपरेटर का उपयोग किसीstringमें pattern matching के लिए करते है।

निम्न दो विशेष करेक्टर के द्वाराPatternको विवरणदे सकतें है।

(1) Underscore (–):– यहाँ(–)करेक्टर कोई भी करेक्टर कोmatchकरता है।

(2)Percent (%) यहाँ(%) करेक्टर कोई भीsub stringकोmatchकरता है।

कुछ डाटाबेस में pattern matchingकेस सेन्सीटीव होती है। जबकि MySQLमें यह(by default) केस इनसेन्सीटीव होती हैअर्थात करेक्टर uppercase (B) करेक्टरlowercase (b) कोmatchकरता है।तथाlowercase,Uppercaseको मैंच करता है।

उदाहरण(1):– अगर हमेंStudent टेबिल से हमें वो सभीName निकालने है। जिनका प्रारम्भSसे होता हैं तो इसकाSQL Syntaxहोगा ।

MySQL> SELECT *FROM Student WHERE Name LIKE 'S%';

Output ->

Roll_No	Name	Age	Address	Class
91	Suman	11	Jaipur	6th
75	Shanker	13	Ajmer	8th

उदाहरण(2) :–वे नाम जिनका अंतshसे होता है।

MySQL> SELECT * FROM Student WHERE Name LIKE '%sh';

Output ->

Roll_No	Name	Age	Address	Class
109	Omprakash	9	Jodhpur	4th
111	Prakash	15	Jaipur	10 th

उदाहरण(3):– वे नाम जो 5 करेक्टर से मिल के बने हो

MySQL> SELECT *FROM Student WHERE Name LIKE '-----';

Output->

Roll_No	Name	Age	Address	Class
91	Suman	11	Jaipur	6th

किसी भीpattern में अगर हम विशेषpattern करेक्टर (%, _) को भी भामिल करना चाहते है। तो हमेंescape character (\) Backslashको उपयोग करते है।इसescape characterको हमें विशेषpattern करेक्टर के तुरंतपहले लगाते है।

उदाहरण :–LIKE 'BJ\%BHARAT%' उस सभी स्ट्रीग कोmatchकरेगा जिनको प्रारम्भ' BJ\%BHARAT%' से होता है।

LIKE 'BJ\\BHARAT%' उन सभी स्ट्रीग को मेच करेगा जिनका प्रारम्भBJ\%BHARAT% से होगा ।

SQL NOT LIKEके उपयोग द्वाराmismatchesको भी ढुंढा जा सकता है।MySQLमेंExtended regular expression के उपयोग द्वारा भी pattern matchingकी जाती है।इसके लिएMySQLमेंREGEXP(RLIKEऔर NOT REGEXP(NOT RLIKE) आपरेटर्स का उपयोग करते है।

"[…]"एक करेक्टर क्लास है जो ब्रेकेटस के अन्दर कोई भी करेक्टर कोmatchकरेगी

उदाहरण :–"[p q r]" "p", "q" or "r"को मैच करती है।

"[a-z]" किसी भी लेटर को मैच करती है।

आपरेटर	विवरण
IS NULL	इस आपरेटर का उपयोग किसी वेल्यू को एक NULLवेल्यू से कम्पेयर करने के काम लेते है।

अगर किसी टेबिल में किसी ऐट्रीब्यूट की वेल्यू उस वक्तAbsenceहै तो उस ऐट्रीब्यूज के लिए इस Absence information को बताने के लिए NULLवेल्यू का उपयोग करते है।

उदाहरण :–के लिए अगर Teacher टेबिल मे किसी टीचर का PhoneNo नही हैं तो वहाँ पर वेल्यूNULLआयेगी औरइसकोtest करने के लिएMySQLमें की–वर्डNULLका उपयोग करेगें ।

SELECT Tname FROM Teacher WHERE PhoneNo IS NULL;

आपरेटर्स एवंमNULLवेल्यू :–

आपरेटर्स	वेल्यू 1	वेल्यू 2	**Output**
+, -, *, or /	वेल्यू 1	null	NULL
+, -, * or /	null	वेल्यू	NULL
>,<, >=,<=,<>	null	वेल्यू	unknown
<,>,,>=,<=,<>,=	वेल्यू	null	unknown
AND	true	unknown	unknown
AND	false	unknown	false
AND	unknown	unknown	unknown
OR	true	unknown	true
OR	false	unknown	unknown
OR	unknown	unknown	unknown
NOT	unknown	unknown	unknown

अर्थात यदि उक्त टेबिल का उपयोग करते हुए यदि WHERE clause में predicateका रिजल्ट false या unknown आता है। तो कोई भी टप्ल रिजल्ट में नहीं आयेगा।

नोटः– सभी aggregateफक्शन(count(*))के अलावा null वेल्यू को जब वेउनके इनपुट लिस्ट में आती हैं, तो उनकोcalculation में नहींलेतें है। अर्थात नजर अंदाज करते हें।

SET Operators:–SQL के SET आपरेशन निम्नलिखित है। जो कि रिलेशनसके ऊपर आपरेटकरते है।

(i) UNION

(ii) UNION ALL

(iii) INTERSET

(iv) EXCEPT

UNION SET आपरेटर्स

इस आपरेटर का उपयोग दो या दो से अधिक SELECT statementके रिजल्ट सेट कोसंयुक्त(combine)करने के काम आता है। यह आपरेटर्स duplicate पक्तियो को हटा(remove)कर रिजल्ट देता है।

नोटः– इस आपरेटर्स के लिए यह आवश्यक है किUNIONहोने वाले दोनोंSELECT statementमें फिल्डस (fields) कीसंख्या एक समान हो तथा उनके डाटा टाइप्स भी एक समान होने चाहिए।

Syntax:- MySQL UNIONआपरेटर syntaxहै।

SELECT ex1, ex2,… ,ex_nFROM tables[WHERE condition]

UNION [DISTINCT]

SELECT ex1, ex2,… ex_nFROM tablesWHERE condition;

यहाँ परDISTINCTकी–वर्ड आवश्यक नहीं है। क्योंकिUNIONआपरेटरduplicate पक्तियों कोstatementमें सेहटा देता है।

उदाहरण :–

SELECT ClassFROM Student

UNION

SELECT Class_nameFROM Classes;

Output ->

Class
4th
10th
6th
8th
12th
9th

यह उदाहरण केवल एक फिल्ड के लिए है अर्थात SELECT statementकेवल एक फिल्डreturnकरता है। दोनोंफिल्डस के यहाँ डाटा टाइप्स समान है। रिजल्ट सेट केcolumn का नाम होगा। यहाँ पहले SELECT statementकेreturnका नाम होगा। क्योंकि हम जानतें है। कि MySQL UNIONआपरेटर्सduplicateको हटाता है। अतः6^{th} यहाँ एक ही

बार आया हैं। और अगर हम duplicate रखना चाहते है तो हम UNION ALL का उपयोग करेंगें।

Syntax :- UNION ALLकाSyntax UNIONजैसा है केवल UNIONके स्थान पर UNION ALL काउपयोग करेगें।

SQL INTERSET आपरेटरः– इस आपरेटर का उपयोग दो या दो से अधिक डाटा सेटस के intersectionके लिए कामआता है। अर्थात यदि दोनों डाटा सेटों में कोई रिकार्ड विद्यमान है तोINTERSETआपरेटर के रिजल्ट में हमें वह रिकार्डप्राप्त होगा अन्यथा अगर कोई रिकार्ड केवल एक ही डाटा सैट में है तो वह रिजल्ट में नहीं आयेगा।

उदाहरणः–

SELECT ClassFROM Student

INTERSET

SELECT Class_name,FROM Classes;

Output ->

Class
6th

नोटः– लेकिन MySQLमेंINTERSET आपरेटर उपलब्ध नहीं है।

MySQLINआपरेटर के द्वारा हम INTERSETआपरेशन कर सकते है।

Syntax:-

MySQL IN आपरेटर syntax

Expression IN(value1 , value2 , …$value_n$);

यहाँ expression वह वेल्यू है जिसको हम test करना है value1,value2,…valuen वह वेल्यूस है जिसमें हमे test वेल्यू को test करता है यदि कोई भी वेल्यू test वेल्यू से matchहोती है। तो IN आपरेटर trueकरता है।

SQL EXCEPT आपरेटरः–SQL EXCEPTआपरेटर दो SELECT statementको संयुक्त करने के उपयोग में आताहैं। और यह इनको संयुक्त इस प्रकार करता हैं कि पहलेSELECT statementकी वह पक्तियाँ जो दूसरेSELECT statementमें नहीं है, को परिणाम स्वरूप देता है।

Syntax:-

SELECT column namesFROM tables[WHERE clause]

EXCEPT

SELECT column namesFROM tablesWHERE clause;

MySQLEXCEPTआपरेटर को भी उपलब्ध नही करवाता है इसके लिए हम NOT IN आपरेटर का उपयोग कर सकते है।

SQL(functions)फंक्शन :–SQLकई प्रकार के उपयोगी built in फंक्शन उपलब्ध करवाता है। जिनका विवरण निम्न प्रकार है।

(1)DateऔरTimeफंक्शन :– में काम आने वाले Dateऔर Timeफंक्शन तथा उनका विवरण नीचे दिया गया है।

फंक्शन का नाम	विवरण
ADDDATE()	Dateको जोड़ता है।
ADDTIME()	Timeको जोड़ता है।
CURDATE()	यह वर्तमान Datereturn करता है।
CURTIME()	यह वर्तमान Time return करता है।
DATE_SUB()	यह दो Dates (subtracts) को घटाता हैं।
NOW()	यह वर्तमान Dateव Timeदेता है।
STR_TO_DATE()	यह स्ट्रींग को Dateमें परिवर्तित करता है।

उदाहरण :–

Syntax:- SELECT ADDDATE(expr, days)

MySQL> SELECT ADDDATE('1980-07-01',32);

Output -> 1980-08-02

उदाहरण :–

Syntax:- SELECT ADDTIME(expr2, expr1)

यहाँexpr1कोexpr2में जोड़कर रिजल्ट देगा ।

MySQL>SELECT ADDTIME('1999-12-31 23:59:59.999999', '11:1:1.000002');

Output -> 2000-01-02 01:01:01.000001

उदाहरण :–

CURDATE syntax :- SELECT CURDATE();

यहाँ वर्तमानYYYY-MM-DD फार्मेट में देगा।

(2) स्ट्रींग(String)फंक्शन :–Stringके उपयोगी स्ट्रींग फंक्शन निम्नानुसार है

नाम	विवरण
ASCII()	यह फंक्शन बांये छोर(left most)के करेक्टर कीnumericवेल्यू देगा।
BIN(N)	यहNकी बाइनरी वेल्यू का स्ट्रींग रिप्रजेन्टेशन (representationदेता है।
BIT_LENGTH(str)	यहstrस्ट्रींग की लम्बाई (length)बीटस में देगा।
CHAR(N)	यह हर आरग्यूमेन्टN को इन्टीजर मानकर उसका स्ट्रींग रिप्रजेन्टेशन देगा। यह स्ट्रींग उन करेक्टर्स सेमिलकर बनेगी जो इन्टीजरCHAR(N) में आरग्यूमेन्ट के तौर पर है।
CHAR_LENGTH(Str)	यहStrस्ट्रींग की लम्बाई करेक्टरस में नापता है।
CONCAT(Str1,Str2,…)	यह आरग्यूमेन्ट स्ट्रींगको concatenate करके प्राप्त स्ट्रींग कोreturn के तौर परदेता है।
FIELD(Str,Str1,Str2,…)	यहStrका इन्डेक्स दी हुई लिस्ट(Str1,Str2,…) में से रिर्टन करता है। अगरStrनहीं मिलती तो 0 रिर्टन करता है।
LOAD_FILE(file_name)	यह फंक्शन फाइल को readकरके उसके कन्टेन्टस को स्ट्रीग रूप में देता है। इसकेउपयोग के लिए serverपर उपस्थित फाईल का पूर्णpath name उल्लेखित करना होता है।
REPLACE(Str, from_Str, to_Str)	यहStrस्ट्रींग को , उसमें सेfrom_Str की सारीoccurrences को to_Strसेreplaceकर के रिर्टन करता है।

उक्त फंक्शन के अलावा की अन्य कई स्ट्रींग फंक्शन MySQLमें हैं ।

उदाहरणः– (1)

MySQL> SELECT ASCII('3')

Output ->

ASCII('3')
51

उदाहरणः– (2)

MySQL> SELECT BIN(2)

Output -> 1 0

उदाहरणः– (3)

MySQL> SELECT BIT_LENGTH('BHARAT')

Output ->

BIT_LENGTH('BHARAT')
48

उदाहरणः– (4)

MySQL> SELECT CONCAT('BH','A','GAT',' ','SI','NGH')

Output -> BHAGAT SINGH

उदाहरणः– (5)

MySQL> UPDATE Student SET Address= LOAD_FILE('pathname') WHERE Roll_No=105 ;

उदाहरणः– (6)

MySQL> SELECT CONCAT(Roll_No, Name, Class) FROM Student

Output ->

CONCAT(Roll_No, Name, Class)
105hari9th
50gopal9th
60gopi8th
109kailash11th

उदाहरणः– (7)

MySQL> SELECT CHAR(66,72,65,82,65,84)

Output->

CHAR(66,72,65,82,65,84)
BHARAT

RANDफंक्शन :– MySQLमें0व1 के बीच में कोई भी नम्बरrandomly निकालने के लिए हमRANDफंक्शनको उपयोग करते है।

उदाहरण :–

MySQL > SELECT RAND(), RAND();

Output-> RAND() RAND()

0.03014567845357
0.93969467893221

SQRTफंक्शनः– यह फंक्शनकिसी नम्बर कासquare root निकालने के काम आता है।

उदाहरण :–

MySQL> SELECT SQRT(64)

Output ->

SQRT
8

अगर हमTeacher टेबिल में Salary field कासquare root निकालना है तो हम उसे निम्न प्रकार उपयोग करेंगें।

MySQL>SELECT Tname, SORT(Salary)FROM Teacher

Output->

Tname	Salary
Radha Krishnan	54.7722557505166
Rajesh	70.7106781188548

Numericफंक्शनः–इस फंक्शनको उपयोग गणितीय आपरेशन में काम में लेते है। कुछ उपयोगी फंक्शननिम्नलिखितहै।

फंक्शन	विवरण
(i)ABS(V)	यह फंक्शनV की पूर्ण वेल्यू देता है।
(ii)GREATEST(n1,n2,…)	यह फंक्शनदी हुई पैरामीटर लिस्ट में से अधिकतम(greatest)वेल्यू देता है।
(iii)INTERVAL(N,n1,n2,n3,----)	यह फंक्शनN की वेल्यू को लिस्ट(,n1,n2,n3,----) से कम्पेयर करता है। औरअगरN<n1है तो0। N<n2है तो 1। N<n3है तो 2 और इसी प्रकार आगे नम्बर रिर्टन करता है।
(iv)LEAST(N1,N2….)	यह GREATESTका विपरीत है।

उदाहरण :–(1) MySQL> SELECT ABS(-6);

Output ->

ABS(-6)
6

उदाहरण :–(2) MySQL> SELECT GREATEST(4,3,7,9,8,0,10,50,70,11)

Output ->

GREATEST(4,3,7,9,8,0,10,50,70,11)
70

उदाहरण :–(3) MySQL> SELECT INTERVAL(4,3,5,8,11,12,17 ,18)

Output ->

INTERVAL(4,3,5,8,11,12,17,18)
1

Aggregate फंक्शनः– MySQLमेAggregateफंक्शनइनपुट के तौर पे वेल्यूस का संग्रहलेते हैऔर आउट पुट में एक वेल्यू देते है।MySQLमेंनिम्न 5 प्रकार केbuilt in Aggregateफंक्शनहोते है।

AVERAGE: Avg()

MAXIMUM: Max()

MINIMUM: Min()

TOTAL: Sum()

COUNT: Count()

यहाँSum एवंम्Averageफंक्शनकी इनपुट वेल्यूस आवश्यक रूप से नम्बर होने चाहिए।जबकि दूसरे आपरेटर्स ,स्ट्रींग के उपर भी काम कर सकते है।

Avg() function फंक्शन :–यह फंक्शनटेबिल के किसी फिल्ड की वेल्यूस का औसत निकालने के उपयोग में आता है।

उदाहरण :–

:- MySQL> SELECT Avg(Salary)FROM Teacher;

Output->

Avg(Salary)
14250.0000

ऊपर दिये उदाहरण मेंAverageफंक्शनSalaryफिल्ड में वेल्यूस का औसत रिर्टन करता है।

Sum() Function फंक्शन :– यह फंक्शनकिसी फिल्ड की सभी वेल्यूस काSumदेता हैं।

उदाहरण :–

MySQL> SELECT Sum(Salary)FROM Teacher;

OUTPUT->

Sum(Salary)
57000

Max() functionफंक्शन :–यह फंक्शनवह रिकार्ड जो किसी रिकार्ड सैट में अधिकतम है को देता है।

उदाहरण :–

MySQL> SELECT Max(Salary) FROM Teacher

Output->

Max(Salary)
40000

Min() functionफंक्शन :–यह फंक्शन निम्नतम वेल्यू वाला रिकार्ड देता है।

उदाहरण :–

MySQL> SELECT Min(Salary) FROM Teacher

Output->

Min(Salary)
3000

Count() functionफंक्शन :–यह फंक्शन टेबिल में रिकार्ड की संख्या गणना के लिए काम में आता है। अर्थात रिकार्डसकी कुल संख्या पता करने के काम आता हैं(counting the number of records)

उदाहरण :–

SELECT Count (*) FROM Student

Output ->

Count(*)
4

उदाहरण :–

SELECT Count (*) FROM Student WHERE Class=”9th”

Output ->

Count(*)
2

SQL ORDER BY क्लॉज:–SQLकेSELECT statementके द्वारा चुनी गई पक्तियों का क्रम कुछ भी हो सकताहै। अगर इन पक्तियों को हमे किसी क्रम में देखना है तो हमेंSQL के ORDER BYक्लॉज का उपयोग उस स्तंभ यास्तंभों के साथ करते है। जिनकों हम क्रय में देखना चाहते है। अर्थातORDER BY के द्वारा हम किसी स्तंभ की वेल्यूस कोबढ़ते(ascending)या घटते(descending)क्रम में देख सकते हैं ।

Syntax:-

SELECT field1,field2….filedn

FROM T1,T2…Tn

ORDER BY field1, field2….filedn [Asc[Desc]];

यहाँ हम क्यूरी के रिजल्ट को एक से ज्यादा फिल्ड पर SQRTकर सकते है। जिसके लिएAscयाDesc है।

उदाहरण :–

MySQL> SELECT Roll_NO, Age FROM Student ORDER BY Age Desc;

Output:-

Roll_No	Age
109	19
105	14
50	13
60	13

उदाहरण :–

MySQL> SELECT Roll_NO, Age FROM Student ORDER BY Age Desc, Roll_No Desc;

Output:-

RollNo	Age
109	19
105	14
60	13
50	13

उदाहरण :–

Query :-उन छात्रों का नाम बताइये जोClassroomनम्बरF-17 में बैठता है।

MySQL> SELECT Name FROM Student, Classes WHERE Class=Class_name AND CRoomNo =”F-17”ORDER BY Name Asc;

इस उदाहरण में हमने विभिन्न टेबिल का उपयोग किया है।क्योंकि हमें विभिन्न टेबिल से सूचनाओं(information)कीआवश्यकता हैं। जैसें कि छात्र का नाम हमStudentटेबिल से प्राप्त होगा जबकि Classroom फिल्डClassesटेबिल मेंहै। अतः हमें दोनों टेबिल को जोड़ते (joine)हुये आवश्यक क्यूरी लिखनी है।इस क्यूरी केWHEREक्लाज में हमने दोनों टेबिल को उनकीprimary keyवforeign keyके द्वारा जोड़ा हैं क्योंकियह दोनों टेबिल एक समान

है। अर्थात दोंनों टेबिल इनके द्वारा लिक्ड़ है। किसी क्यूरी में हम दो या दो से अधिक टेबिल को कॉमन फिल्ड के द्वारा जोड़ते है।

यहाँ हमने SQL ORDERक्लाज का भी उपयोग किया है। क्योंकि हम रिजल्ट को बढ़ते क्रम में चाहते है। अर्थात छात्रों का नाम इनके ऐलफाबेटिकल आर्डर में प्राप्त हो ।

Output->केवल वे छात्र जो F-17 में बैठते है।

Name
--

SQL GROUP BY क्लॉजः–MySQLका यह क्लॉज अत्यन्त उपयोगी इस क्लॉज के उपयोग द्वारा कई महत्वपूर्ण क्यूरीलिखी जा सकती है। GROUP BYक्लास के द्वारा किसी स्तंभ(column)या स्तंभों या ऐट्रीब्सूटस की वैल्यूस का समूहबनाया जा सकता है। अर्थात क्लॉज में दिय गये ऐट्रीब्यूट का उपयोग समूह(GROUP)बनाने के लिए करते है। GROUP BYक्लॉज में दिये गये एट्रीब्यूट या एट्रीब्यूटस की वेल्यूज जिन टप्लस या पक्तियों की लिए एक समान है। वेसभी टप्लस या पक्तियों एक समूह में आयेगी।

GROUP BYक्लॉज को हम निम्न उदाहरण द्वारा समझ सकते है। इसके लिए Studentटेबिल को लेतें है। जिसमेंकिसी क्षण निम्न रिकार्डस है।

Roll_No	Name	Age	Address	Class
1	Ajay	9	Jaipur	4th
2	Vijay	17	Kota	12th
10	Hari	11	Udaipur	7th
17	Shanker	13	Jaipur	8th
21	Om	21	Ajmer	12th
51	Mayank	15	Ajmer	9th
90	Anju	18	Ajmer	11th
53	Suman	12	Ajmer	10th
64	Kamal	10	Kota	4th
500	Komal	16	Udaipur	9th
700	Aryabhatt	11	Jaipur	7th
900	Bodhayan	13	Jodhpur	8th

Figure 5 Student table

उदाहरण :–

क्योरी:–प्रत्येक Classमें पढ़ने वाले छात्रों की संख्या बताइयें।

अगर हम यह क्योरी निम्न प्रकार से लिखगें तो परिणाम गलत प्राप्त होगा।

MySQL> SELECT Count(*) FROM Student

Output->

Count (*)
12

क्योंकि इस Syntaxके द्वारा कुल पढ़नें वालो छात्रों की संख्या प्राप्त होगी अर्थात Studentटेबिल में जितने छात्रों का रिकार्ड उपलब्ध है। वही उस स्कूल में पढ़नें वाले छात्र है। अतः यह Syntax,Studentटेबिल में कुल टपल्स कितने है। उनकी संख्या देगा।सही परिणाम के लिए Count aggregateफंक्शन केGROUP BYसाथ क्लाज का उपयोग करना पड़ता है। जिसकाSyntaxनिम्नानुसार है।

MySQL> SELECT Class, Count(Roll_No) FROM Student GROUP BY Class;

Studentटेबिल में समूह बनने के बाद टेबिल के बादStudentटेबिल कुछ इस प्रकार दिखेगी क्योंकि GROUP BYक्लाज में Classऐट्रीब्यूट के द्वारा समह बनाया जा रहा है। अतः एक समान Classवाली पक्तियाँ एक समूह में दिखेगी।

Output ->

Class	Roll_No	Age	Name	Address
11th	90	18	Anju	Ajmer
12th	2	17	Vijay	Kota
12th	21	21	Om	Ajmer
10th	53	12	Suman	Ajmer
9th	51	15	Mayank	Ajmer
9th	500	16	Komal	Udaipur
8th	17	13	Shanker	Jaipur
8th	900	13	Bhodhayan	Jodhpur
7th	10	11	Hari	Udaipur

7th	700	11	AryaBhatta	Jaipur
4th	1	9	Ajay	Jaipur
4th	64	10	Komal	Kota

Class	Count(Roll_No)
10th	1
11th	1
12th	2
9th	2
8th	2
7th	2
4th	2

जिस columnके द्वाराGROUPबनाया जाता है। उसcolumnपर कोईcalculation Count, Avg, Max, Min आदि aggregate functionके द्वारा कि जा सकती है। अतः इस क्यूरी मेंCount ,aggregate फंक्शन को हर समूहके टपल्स जिनकीClass एक समान है कि लिए लगाया गया है। क्योंकि SELECTक्लॉज मे दो ही फिल्ड़ है। अतःपरिणाम निम्नानुसार प्राप्त होगा ।

Class	Count(Roll_No)
10th	1
11th	1
12th	2
9th	2
8th	2
7th	2
4th	2

नोटः– किसीSELECTक्लॉज में aggregateफंक्शन के बाहर जो भी एट्रीब्यूटस आतें है। वे GROUP BYक्लास केअन्दर भी लिखना है। जैसेSELECTक्लॉज में Classएट्रीब्यूट आने पर वह GROUP BYक्लॉज में भी आया है।

उदाहरण :–

क्योरी:- उनClassके नाम बताइये (प्रत्येकClass)जिनमें पढ़ने वाले छात्रों की संख्या 1 से अधिक है।

Syntax:-

MySQL> SELECT Class, Count(Roll_No)FROM StudentGROUP BY ClassHAVING Count(Roll_No)>1;

Output->

Class	Count(Roll_No)
12th	2
9th	2
8th	2
7th	2
4th	2

यहाँ रिजल्ट में प्रत्येक GROUPसे वही टपल्सSELECTहुये है जिनका Count1 से अधिक है। SQLमें एक किसीटपल्स के लिए कोई शर्त पूर्ण होती है या नहीं यह देखने के लिए WHEREक्लॉज का उपयोग करते है। जबकि GROUP BY क्लॉज के द्वारा बनाये गये समूहों में उपस्थित टपल्स के लिए शर्त को देखने(test)के लिएHAVING क्लॉज काउपयोग करते हैं।WHEREक्लॉज एवंमHAVINGक्लॉज में यह मुख्य अन्तर है।

SQLमेंHAVINGक्लाज में दर्शाये गये predicateको लागूGROUP BYक्लाज के द्वाराGROUPबनाने के बादकरते है। इसलिए इसके साथaggregateफंक्शन भी उपयोग कर सकते है।

नोट :– अगर किसी क्योरी मेंWHERE, HAVING, GROUP BY आते है तो सबसे पहलेWHEREक्लॉज मेंलागू होगा उसके बाद जिन टपल्स के लिए शर्त पूर्ण होगी वे GROUP BY क्लॉज के द्वारा एक समूहों में रखेगें औरअन्त में हर समूह के लिएHAVINGक्लॉज को लागू करेंगें जिन समूहों के लिएHAVINGक्लॉज संतोशजनक नहींरहता वे समूह परिणाम में से हट जाते है।

SQL RENAME आपरेंशन :– SQLमें इस तरह की व्यवस्था है कि हम किसी भी रिलेशन का और ऐट्रीब्यूट का नामबदल सकते है। इसके लिए SQLमे'AS'क्लॉज के उपयोग निम्न प्रकार से करते है।

Old_relation/ attribute_name as new_name

'AS' क्लॉज का उपयोगSELECTव FROMदोनों क्लोजों में किया जा सकता है।

उदाहरण :–

MySQL> SELECT Avg(Salary) AS AvSalary FROM Teacher;

AvSalary
14250.0000

उदाहरण :–

MySQL> SELECT Class, Count(Roll_No) AS totalFROM Student GROUP BY Class;

Output->

Class	total
10th	1
11th	1
12th	2
9th	2
8th	2
7th	2
4th	2

SQL JOIN:–SQLकेJOINkeywordका उपयोग दो या दो से अधिक टेबिलस से डाटा की क्यूरी करने के कामलिए होता है। आपरेशन दो रिलेशन को इनपुट के तौर पर लेते है। और एक रिलेशन आउट पुट के तौर पर देते है। SQLमें दो रिलेशन कोJOINकरने के कई तंत्र(mechanisms)है।

जैसं कि (1) Cartesian product mechanism (2) Inner join (3) Outer join (left, right, full)

ऊपर दिये प्रत्येक join type के लिए एकJoin condition भी जुड़ी हुई होती है। अतः एकJoin expression इन दोनों(join typeऔरJoin expression)से मिलकर बनती है।जिसे हमFROM क्लॉज में उपयोग करते है।JOINको समझने के लिए हम Studentटेबिल नम्बर 5 एवंमClassटेबिल नम्बर 6 जिसमें किसी क्षण निम्न रिकार्ड को लेते है।

Class_name	CRoomNo	CStrength
12th	F-1	90
11th	F-2	75
10th	F-5	99
9th	S-21	110
8th	S-10	70
7th	F-10	85
6th	F-17	65
5th	F-7	60
4th	F-9	55
3th	F-8	50
2th	S-15	35
1th	S-9	60

Studentएवंम् Classटेबिल को JOINकरने के लिए एक क्योरी लिखते है।

MySQL> SELECT Roll_No, Class, CStrengthFROM Student AS St, Classes AS S,WHERE St.Class=S.Class_name;

इस क्योरी मेंFROMक्लॉज मेंStudent टेबिल को RENAMEकर केStएवंम्Classको Sकिया गया है। इनदोनों रिलेशनका Cartesian product होगा जिसमें Stटेबिल के हर टपल्स काSटेबिल के हर टपल्स से JOINहोगा। अतः प्राप्त रिलेशन में कुल टपल्स होगें।

N1 * N2 =12*12=144

यहाँN1 Stटेबिल में टपल्स की संख्या व N2 Sटेबिल में टपल्स की संख्या है।यहाँ पर रिजल्ट टेबिल में टपल्स प्राप्त होगें वो WHEREक्लॉज की शर्तें को पूर्ण करने वाले ही होंगें।

OUTER JOIN आपरेंशन :– निम्न प्रकार के होते है।

(1) LEFT OUTER JOIN (2) RIGHT OUTER JOIN (3) FULL OUTER JOIN

Outer Joinsके साथ निम्नjoin condition का उपयोग करते है।

1) Natural

2) ON (Predicate)

3) Using (A1,A2,…….An)

Left outer join और ON Join condition:–Left join को समझने के लिए निम्न दो टेबिल लेते है।

Classes

Class_name	CRoomNo	CStrength
12th	F-11	90
9th	S-21	110
7th	F-10	85
4th	F-9	55

Admission

Class_name	Roll_No	admission_date
12th	79	2000/7/15
9th	89	2010/8/13
6th	69	2012/7/21
5th	49	2013/7/03

Syntax:- Select Classes, Class_Name, Roll_No, CStrengthFrom Classes Left Outer Join Admission on Classes.class_Name=Admission.Class_Name

यहाँ रिलेशन का नाम उसके ऐट्रीब्यूट के साथ लिखते है। क्योंकि एक समान नाम का ऐट्रींब्यूट एक से ज्यादा रिलेशन में है।अतः अस्पश्टता(ambiguity)को दूर करने के लिए ऐसा किया है।

Output

Class_name	Roll_No	CStrength
12th	79	90
9th	89	110
7th	Null	85
4th	Null	55

Left Outer Join के रिजल्ट में दोनों रिलेशन केMatching tuplesतथाLeftवाले रिलेशन(classes)केunmatched tuples उपस्थित होते है।

Right Outer Joinऔरon Join Condition:–

Syntax:

Classes Right Outer Join admission on classes.Class_Name=Admission.Class_Name

Output

Class_Name	CRoomNo	CStength	Class_Name	Roll_No	Admission_date
12th	F-11	90	12th	79	2000/7/15
9th	S-12	110	9th	89	2010/8/13
Null	Null	Null	6th	69	2012/7/21
Null	Null	Null	5th	49	2013/7/03

Right Outer Join आपरेशन Left Outer Join के समान ही है। किन्तु इसमें Join operation के दाँयी (right)और वाले रिलेशन के unmatched tuples भी आते है। leftरिलशन के एट्रीब्यूटस के लिए वेल्यूस Nullरखेगें।

Full outer join और On condition:–

Syntax:

Classes full outer join admission on classes. Class_Name= Admission.Class_Name

Class_Name	CRoomNo	CStrength	Class_Name	Roll_No	Admission_date
12th	F-11	90	12th	79	2000/7/15
9th	S-12	110	9th	89	2010/8/13
7th	F-10	85	Null	Null	Null
4th	F-9	55	Null	Null	Null
Null	Null	Null	6th	69	2012/7/21
Null	Null	Null	5th	49	2013/7/03

यहाँ दोनों रिलेशन केunmatchedटपल्स भी आयेगें। साथ में दूसरे रिलशन केunmatched टपल्स के लिए Nullआयेगा।

Outer Join और Natural condition:–दो रिलेश्न का NaturalJoinकरने पर उन टपल्स की संख्या उन रिलेशनमें उपस्थित एक समान(common)ऐट्रीब्यूट के द्वारा प्राप्त होते है औरcommonट्रीब्यूटसresultरिलेशन में एक बार आते है। वह क्रम में सबसे पहले आते है।

उदाहरण :–Classes Natural right outer join admission

Class_Name	CRoomNo	CStength	Roll_No	Admission_date
12th	F-11	90	79	2000/7/15
9th	S-12	110	89	2010/8/13
6th	Null	Null	69	2012/7/21
5th	Null	Null	49	2013/7/03

अन्य outer joinभी Natural conditionशर्त के लिए ऊपर दिये अनुसार हम प्राप्त कर सकते है।

Inner join :–

उदाहरण :–

Classes inner Join Admission OnClasses.Class_Name= Admission.Class_Name;

Output->

Class_Name	CRoomNo	CStength	Class_Name	Roll_No	Admission_date
12th	F-11	90	12th	79	2000/7/15
9th	S-12	110	9th	89	2010/8/13

Inner join और natural condition:–

उदाहरण :–

Classes Natural Inner-Join Admission

Output ->

Class_Name	CRoomNo	CStength	Roll_No	Admission_date
12th	F-11	90	79	2000/7/15
9th	S-12	110	89	2010/8/13

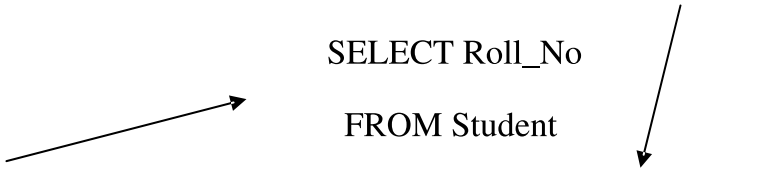
यहाँ केवल एक ही ऐट्रीब्यूट दोनों रिलेशन में समान है। अतः Joinकेवल उस ही ऐट्रीब्यूट के द्वारा होगा।

नोटः– Join condition USINGभी Natural Join के समान ही केवल joiningऐट्रीब्यूट USINGमें (A1,A2,…An) होते है। जबकि Naturalमें सारे ऐट्रीब्यूट जो दोनों रिलेशन में एक समान हैं होते है। यही Naturalएंवम् USING conditionका मुख्य अन्तर है।

SQL Sub queries:– एक Sub queriesइस प्रकार कीSQLक्योरी होती है जो किसी अन्य क्योरी के भीतर नेस्टेड(nested)होती है। इसके अलावाSub queries खुद भी अन्य Sub queriesके भीतर नेस्टेड(nested)हो सकतीहै।Sub queries को inner query क्योरी भी कहते है तथा जिसे क्योरी के भीतर Sub queriesहोती है उसे Outerक्योरी (बाहरी क्योरी) कहते है।

उदाहरण :–

Inner query



Sub queriesके द्वारा दी गई एक वेल्यू को कम्पेयर करने के लिए कम्पेरिशन आपरेटर (=,>=,<= etc)आदि का उपयोगकर सकते है।

उदाहरण के लिएTeacherटेबिल का उपयोग करते है।

SELECT Tname, SalaryFROM TeacherWHERE Salary= (SELECT Max(Salary) FROM Teacher);

Output ->

Tname	Salary
Hariom	40000

उदाहरण :–

उन Teacherके नाम बताइये जिनकी Salary सभीTeacherकी औसतSalaryसे कम है।

SELECT Tname, SalaryFROM TeacherWHERE Salary< (SELECT Avg(Salary) FROM Teacher);

Output ->

Tname	Salary
Radha krishnan	3000
Rajesh	5000
Lalaji	9000

महत्वपूर्ण बिंदु

- डाटा बेस स्कीमा किसी डाटा बेस की लॉजिकल डिजाईन है जो कि शायद ही बदलती है।
- किसी डाटा बेस में समय के किसी भी क्षण डाटा के समूह को डाटा बेस इन्सटेन्स कहते है।
- प्राइमेरी की :– किसी टेबिल में एक या अधिक फिल्डस (attribute) का ऐसा set जो कि उस टेबिल की किसी भी पर्क्ति अथवा टपल्स को uniquely identify करता है।
- रेफरेनशीयल इन्टीग्रीटी की सुनिश्चितता फोरेन की के द्वारा की जा सकती है।
- स्ट्रकचर क्यूरी लेग्वेज(SQL)रिलेशनल ऐलजेबरा एवंम् रिलेशनल केलकूलस के कोमबीनेशन का उपयोग करती है।
- AUTO_INCREMENTका प्रयोगफिल्ड में वेल्यू को एक से आगे बढाने के लिए किया जाता है।
- GRANT ALL Privileges ON *.* TO 'new_user @ 'localhost';

  यहाँasterisk (*) डाटाबेस व टेबिल को बताता हैं। यह कमाण्ड यूजर कोread, edit,execute और सभी आपरेशनस कीसहमती सभी डाटाबेस और टेबिल के लिए देता है।
- GROUP BYक्लॉज में दिये गये एट्रीब्यूट या एट्रीब्यूटस की वेल्यूज जिन टप्लस या पर्क्तियों की लिए एक समान है वेसभी टप्लस या पर्क्तियों एक समूह में आयेगी।
- SQLमें दो रिलेशन कोJOINकरने के कई त़ंत्र(mechanisms)है जैसे कि 1) Cartesian product mechanism (2) Inner join (3) Outer join (left, right, full)।

अभ्यासार्थ प्रश्न

वस्तुनिष्ठ प्रश्नः

प्रश्न 1. जो एक SQL खंड नहीं है?

(अ) Select (ब) From (स) where (द) condition

प्रश्न 2. SQL का पूर्ण रूप है

(अ) Structure Question language (ब) syntax question language

(स) Structure query language (द) Structure question language

प्रश्न 3. DDL के लिए है।

(अ) डेटा डेफिनेशन लेंग्वेज (ब) डबल डेटा लेंग्वेज

(स) डेटा डेटा लेंग्वेज (द) इनमें से कोई नहीं

प्रश्न 4. Count() है एक

(अ) स्ट्रिंग फंक्शन (ब) सांख्यिक फंक्शन (स) दोनों (द) मौजूद नहीं

प्रश्न 5. Like ऑपरेटर के लिए उपयोग किया जाता है, जैसे

(अ) Concatenating strings (ब) count string character

(स) string matching (द) all

अतिलघुत्रात्मक प्रश्नः

प्रश्न 1. SQL क्या है?

प्रश्न 2. आप SQL from clause द्वारा क्या समझते हो?

प्रश्न 3. SQL select clause के महत्व क्या है?

प्रश्न 4. SQL के प्रकार का नाम दे।

प्रश्न 5. अद्वितीय और प्राथमिक बाधाओं के बीच क्या अंतर है?

प्रश्न 6. डेटाबेस instances.को परिभाषित करो।

प्रश्न 7. हम SQL में order by clause का उपयोग कैसे करते है।

प्रश्न 8. SQL में NULL क्या है?

प्रश्न 9. आप aggregate फंक्शन द्वारा क्या समझते हो ?

प्रश्न10. हम SQL grantकमाण्ड का उपयोग क्यों करते है।

लघुत्रात्मक प्रश्नः

प्रश्न 1. SQL की बुनियादी संरचना क्या है?

प्रश्न 2. विभिन्न DMLकमाण्ड क्या हैं उन के लिए syntaxes दे।

प्रश्न 3. फॉरेन key क्या है? हम किसी टेबिल में एक फॉरेन key कैसे बनाते है।?

प्रश्न 4. SQL के group by clause का उपयोगउदाहरण केद्वारा समझाओं।

प्रश्न 5. Cartesian join और natural join में क्या अंतर है।

निबंधात्मक प्रश्नः

प्रश्न 1. SQL joins, टेबिल्स के उपयुक्त उदाहरण लेने के साथ समझाओ।

प्रश्न 2. aggregate फंक्शन क्या हैं? हम aggregate फंक्शन का उपयोग कैसे करें? प्रत्येक का एक उदाहरण दे।

प्रश्न 3. किसी एकल SQL क्वेरी में where, group by और having clause के उपयोग की व्याख्या, करें ? एक उपयुक्त उदाहरण दे।

प्रश्न 4. दिए गए निम्न स्कीमा पर विचार करें।

students(Roll_no, Sname, age, phone, address, class)

Classes(Class_name, CRoomNo, CStrength)

और निम्न के लिए कोई SQL सिंटैक्स लिखें।

1) कमरा नंबर F-12 में बैठे 5वींclass के छात्रों का नाम का पता लगाए ।

2) अजमेर में रहने वाले 10वीं कक्षा के छात्रों की संख्या का पता लगाएं।

प्रश्न 5. उप क्वेरीज (Sub queries) द्वारा आपका क्या मतलब है? उप क्वेरीज क्यों उपयोगी होती हैं। सेट comparison में उप क्वेरी का उपयोग समझाओ।

उत्तरमाला

उत्तर 1: द उत्तर 2: स उत्तर 3: अ

उत्तर 4: ब उत्तर 5: स