DEPARTMENTOFCOMPUTERSCIENCEAND ENGINEERING

DATABASEMANAGEMENTSYSTEMS

(R22A0584)

LABORATORYMANUAL&RECORD

B.TECH
(IIYEAR–ISEM)
(2024-25)

Name: _____
Roll No:_____
Section:_____
Year:_____

MALLAREDDYCOLLEGEOFENGINEERING&TECHNOLOGY

(AutonomousInstitution–UGC,Govt.ofIndia)

Recognizedunder2(f)and12(B) of UGCACT1956
(AffiliatedtoJNTUH,Hyderabad,ApprovedbyAICTE-AccreditedbyNBA&NAAC–'A'Grade-ISO9001:2015Certified)Maisammaguda,Dhulapally(PostVia.Hakimpet),Secunderabad–500100,TelanganaState,India

DEPARTMENTOFCOMPUTERSCIENCEANDENGINEERING

Vision

- To acknowledge quality education and instill high patterns of discipline making the student'stechnologicallysuperiorandethicallystrongwhichinvolvestheimprovement in the qualityof lifeinhuman race.

Mission

- To achieve and impart holistic technical education using the best of infrastructure, outstanding technical and teaching expertise to establish the students to competent and confident engineers.
- Evolvingthecenterofexcellencethroughcreativeandinnovativeteachinglearningpractices for promoting academic achievement to produce international competitive and world class professionals.

PROGRAMMEEDUCATIONALOBJECTIVES(PEOs)

PEO1–ANALYTICALSKILLS

1. To facilitate the graduates with the ability to visualize, gather information, articulate, analyze,solvecomplexproblems,andmakedecisions.Theseareessentialtoaddressthe challenges of complex and computation intensive problems increasing the irproductivity.

PEO2–TECHNICALSKILLS

2. To facilitate the graduates with the technical skills that prepare them for immediate employment and pursue certification providing a deeper understandingofthetechnologyinadvancedareasofcomputerscienceandrelated fields ,thus encouraging to pursue higher education and research based on their interest.

PEO3–SOFTSKILLS

3. Tofacilitatethegraduateswiththe soft skills thatinclude fulfilling the mission,setting goals, showing self-confidence by communicating effectively, having a positive attitude, getinvolvedin team-work,beingaleader, managingtheircareer and their life.

PEO4–PROFESSIONALETHICS

4. Tofacilitatethegraduateswiththeknowledgeofprofessionalandethicalresponsibilitiesby paying attention to grooming, being conservative with style, following dress codes, safety codes, and adapting themselves to technological advancements.

PROGRAMSPECIFICOUTCOMES(PSOs)

EngineeringGraduateswillbeableto:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals,and anengineering specialization to the solution of Complexengineeringproblems.

2. **Problemanalysis**:Identify,formulate,reviewresearchliterature,andAnalyzecomplex engineering problems reaching substantiated conclusions using first Principles of mathematics, naturalsciences, and engineering sciences.

3. **Design / development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the Specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmentalconsiderations.

4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including designofexperiments, analysisand interpretationofData and synthesis ofthe information to provide valid conclusions.

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modernengineeringandITtoolsincludingpredictionandmodelingtocomplexengineering activities with anunderstanding ofthe limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legaland cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and needfor sustainable development.

8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities andnorms ofthe engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leaderin diverse teams, and in multidisciplinarysettings.

10. **Communication:** Communicate effectively on complex engineering activities With the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clearinstructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of th engineering and management principles and apply these to one’s own work, as a member and leader ina team, to manage projects and in multidisciplinaryEnvironments.

12. **Life-** long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

INDEX

S.No	Date	Nameofthe Activity/Experiment	Grade/Marks	Faculty Signature

INDEX

INTRODUCTION

HierarchicalModel

Thismodelislikeahierarchicaltreestructure,usedtoconstructahierarchyofrecords inthe form of nodes and branches. The data elements present in the structure have Parent- Childrelationship. Closely related information in the parent-child structure is stored together as a logicalunit.Aparentunitmayhavemanychildunits,butachildisrestrictedtohaveonlyoneparent.

Thedrawbacksofthismodelare:

The hierarchical structure is not flexible to represent all the relationship proportions,whichoccur intherealworld.

Itcannot demonstrate the overall data modelfor the enterprise because of the non-availabilityof actualdataatthetimeofdesigningthedatamodel.

ItcannotrepresenttheMany-to-Manyrelationship.

NetworkModel

It supports the One-To-One and One-To-Many types only. The basic objects in this model areDataItems,DataAggregates,RecordsandSets.

It is an improvement on the Hierarchical Model. Here multiple parent-child relationships areused. Rapid and easy access to data is possible in this model due to multiple access paths to thedataelements.

RelationalModel

Doesnotmaintain physical connection between relationsDataisorganizedintermsofrowsandcolumnsinatable

Thepositionofarowand/or columninatableisof noimportanceTheintersectionofarowandcolumnmustgiveasingleva lue

FeaturesofanRDBMS

Theabilitytocreatemultiplerelationsandenter datainto themAnattractivequerylanguage

Retrievalofinformationstoredinmorethanonetable

AnRDBMSproducthastosatisfyatleastSevenof the12rulesofCoddtobeacceptedasafull-fledgedRDBMS.

RelationalDatabaseManagementSystem

RDBMSisacronymforRelationDatabaseManagementSystem.Dr.E.F.CoddfirstintroducedtheRelationalDatabaseModelin 1970.TheRelationalmodelallows data toberepresented in a simple row-column.Eachdatafieldisconsideredasacolumnandeachrecordisconsideredasarow.Relational Database is more or less similar to Database Management S ystem.In relational model there is relationbetweentheirdataelements.Dataisstoredintables.Tableshavecolumns,rows andnames. Tables can be related to each other if each has a column with acommontypeofinformation.ThemostfamousRDBMSpackagesareOracle,SybaseandInformix.

SimpleexampleofRelationalmodelisasfollows:

StudentDetailsTable

Roll_no	Sname	S_Address
1	Rahul	Satelite
2	Sachin	Ambawadi
3	Saurav	Naranpura

StudentMarksheetTable

Rollno	Sub1	Sub2	Sub3
1	78	89	94
2	54	65	77
3	23	78	46

Here,bothtablesarebasedonstudents'details.CommonfieldinbothtablesisRollno.Sowecansaybothtables are related with each other through Roll no column.

DegreeofRelationship

OnetoOne(1:1)

OnetoManyorManytoOne(1:M/M:1)Many to Many(M:M)

TheDegreeofRelationshipindicatesthelinkbetweentwoentitiesforaspecifiedoccurrenceofeach.

OnetoOneRelationship:

(1:1)11

StudentHasRollNo.

One studenthasonlyone Rollno. Forone occurrenceofthefirstentity, therecanbe,atthemostonerelatedoccurrenceof thesecondentity,andvice-versa.

OnetoManyorManytoOneRelationship:(1:M/M:

1) 1M

CourseContainsStudents

As per the Institutions Norm, One student can enroll in one course at a time however, in one course, there can be more than one student.

Foroneoccurrenceofthefirstentitytherecanexistmanyrelatedoccurrencesofthesecondentity and for every occurrence of the second entity there exists only one associated occurrence of the first.

ManytoManyRelationships:

(M:M)MM

StudentsAppearsTests

The major disadvantage of the relational model is that a clear-cut interface cannot be determined. Reusability of astructureis notpossible. TheRelational Databasenowaccepted model on which major database system are built.

Oraclehas introduced added functionality to this by incorporated object-oriented capabilities. Now it is known is as Object Relational Database Management System (ORDBMS). Object-orientedconceptisaddedinOracle8.

Some basic rules have to be followed for a DBMS to be relational. They are known as Codd's rules,designedinsuchawaythatwhenthedatabaseis readyforuseit encapsulatestherelational theory to its full potential. These twelve rules are as follows.

E.F.CoddRules

1. **TheInformationRule**

   Allinformationmustbestoreintableasdatavalues.

2. **TheRuleofGuaranteedAccess**

   Everyiteminatablemustbelogicallyaddress ablewiththehelp ofatablename.

3. **TheSystematicTreatmentofNullValues**

   TheRDBMSmustbetakencareofnullvaluestorepresentmissingorin applicable information.

4. **TheDatabaseDescriptionRule**

   Adescriptionofdatabaseismaintainedusingthesamelogicalstructureswithwhich data was defined by the RDBMS.

5. **ComprehensiveDataSublanguage**

   Accordingtotherulethesystemmustsupportdata definition,viewdefinition, data manipulation, integrityconstraints,authorizationandtransactionmanagement operations.

6. **TheViewUpdatingRule**

   Allviewsthataretheoreticallyupdatablearealsoupdatablebythesystem.

7. **TheInsertandUpdateRule**

   Thisruleindicatesthatallthedata manipulationcommandsmust beoperationalon sets of rows having a relation rather than on a single row.

8. **ThePhysicalIndependenceRule**

   Applicationprograms mustremainunimpairedwhenanychangesaremadein storage representation or access methods.

9. **TheLogicalDataIndependenceRule**

   Thechanges that aremadeshould not affect theuser's abilityto workwiththedata.Thechange can be splitting table into many more tables.

10. **TheIntegrityIndependenceRule**

    Theintegrityconstraintsshouldstoreinthesystemcatalogorinthedatabase.

11. **TheDistributionRule**

    Thesystemmustbeaccessormanipulatethedatathatisdistributedinothersystems.

12. TheNon-subversionRule

IfaRDBMSsupportsalower-levellanguagethenitshouldnotbypass any integrity constraints defined in the higher level.

ObjectRelationalDatabase ManagementSystem

Oracle8 and later versions are supported object-oriented concepts. A structure once created can be reused is the fundamental of the OOP's concept. So, we can say Oracle8 is supported Object Relational model, Object-oriented model both. Oracle products are based on a concept known as a client-servertechnology.Thisconceptinvolvessegregatingtheprocessingofanapplicationbetween two systems. One performs all activitiesrelated to the database (server)and the other performs activities that help the user to interact with the application (client). A client or front-end database application also interacts with the database by requesting and receiving informationfrom database server.It acts as an interface between the userand the database.

The database server or back end is used to manage the database tables and also respond to client requests.

Introductionto ORACLE

ORACLE is a powerful RDBMS product that provides efficient and effective solutions for major database features. This includes:

Largedatabasesandspacemanagement
control Many concurrent database
users

Hightransaction processing performance High
availability

Controlledavailability

Industryacceptedstandards Manage
able security

Databaseenforcedintegrity
Client/Server environment

Distributeddatabase
systems Portability

Compatibility

Connectivity

AnORACLEdatabasesystemcaneasilytakeadvantageofdistributedprocessingbyusingitsClient/Server architecture. In this architecture, the database system is divided into two parts:

Afront-endoraclientportion

Theclientexecutesthedatabaseapplicationthataccessesdatabaseinformationandinteractswiththeuser.

Aback-endoraserverportion

TheserverexecutestheORACLEsoftwareandhandlesthefunctionsrequiredforconcurrent,shareddataaccess to ORACLE database.

Facultysignature

ILLUSTRATION:ROADWAYTRAVELS

"RoadwayTravels"isinbusinesssince1977withseveralbusesconnectingdifferentplacesinIndia.Itsmain officeis located inHyderabad. Thecompany wants to computerize its operations in thefollowing areas:

Reservations
Ticketing
Cancellations

Reservations:

Reservationsaredirectlyhandledbybookingoffice.Reservationscanbemade60 daysinadvanceineithercashorcredit.Incasetheticket isnot available,awaitlistedticketis issued to the customer. This ticket is confirmed against the cancellation.

Cancellationandmodification:

Cancellationsarealsodirectlyhandedatthebookingoffice. Cancellationcharges will be charged.

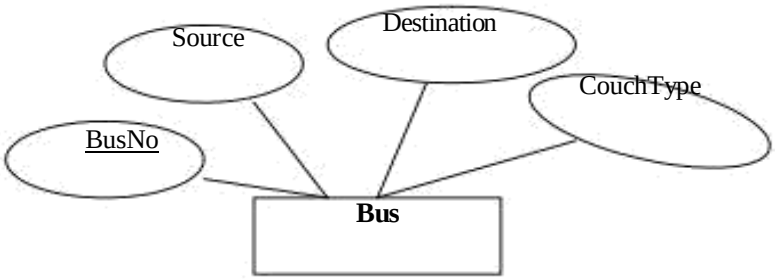
Waitlistedticketsthatdonotgetconfirmedarefullyrefunded.

WEEK-1

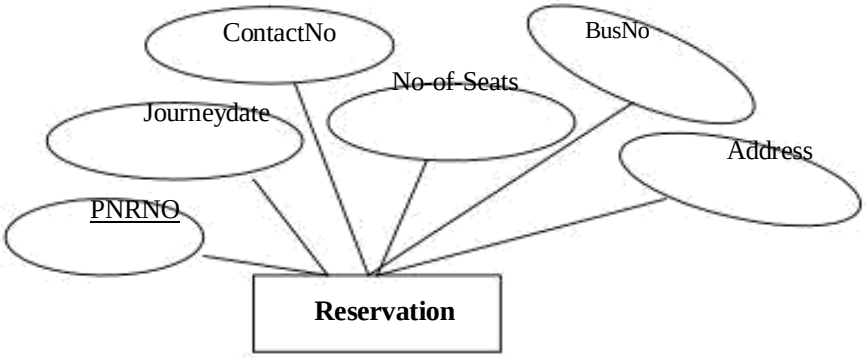
AIM:Analyzetheproblemandcome withthe entitiesinit.
IdentifywhatDatahastobepersistedinthedatabases.
TheFollowingaretheentities:

1. **Bus**
2. **Reservation**
3. **Ticket**
4. **Passenger**
5. **Cancellation**

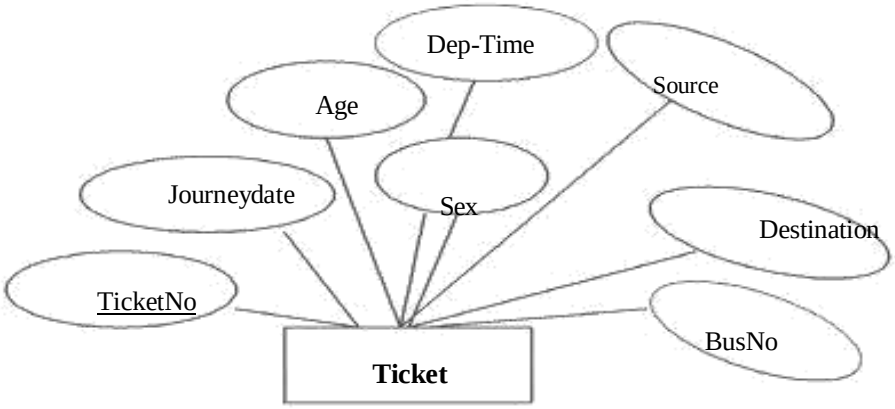
TheattributesintheEntities:B us:(Entity)



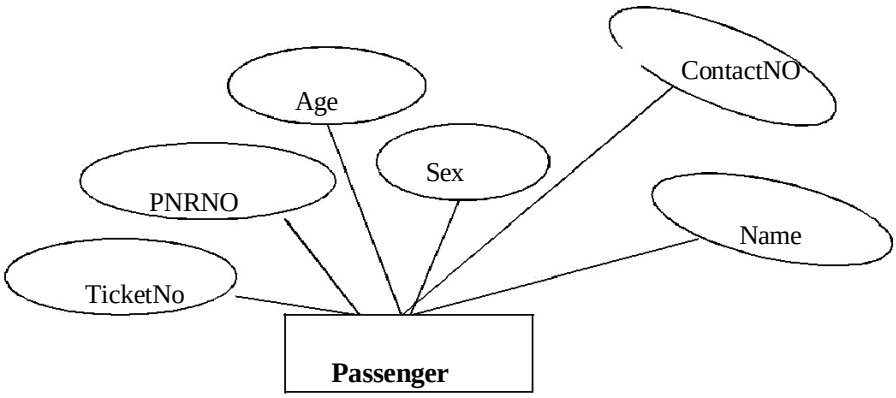
Reservation(Entity)



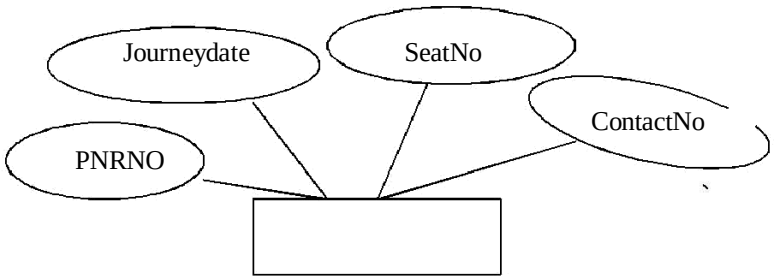
Ticket:(Entity)



Passenger:

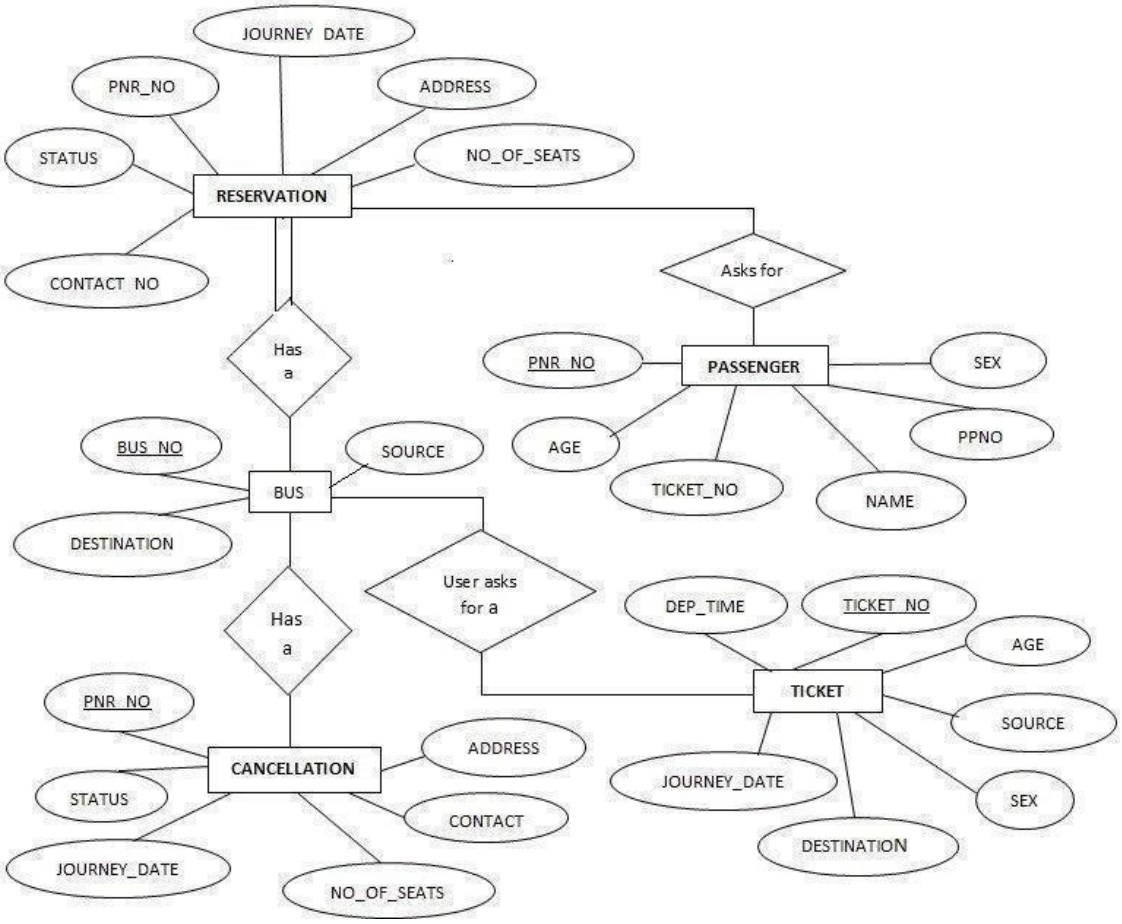


Cancellation(Entity)



Cancellation

ConceptdesignwithE-RModel:



CASESTUDY1:

- Considerthefollowinginformationaboutauniversitydatabase:
- ProfessorshaveanSSN,aname,anage,arank,andaresearchspecialty.
- Projectshaveaprojectnumber,asponsorname(e.g.,NSF),astartingdate,anendingdate,andabudget.
- GraduatestudentshaveanSSN,aname,anage,andadegreeprogram(e.g.,
- M.S.orPh.D.).
- Eachprojectismanagedbyoneprofessor(knownastheproject'sprincipalinvestigator).
- Eachprojectisworkedonbyoneormoreprofessors(knownastheproject'sco-investigators).
- Professorscanmanageand/orworkonmultipleprojects.
- Eachprojectisworkedonbyoneormoregraduatestudents(knownastheproject'sresearchassistants).
- Whengraduatestudentsworkonaproject,aprofessormustsupervisetheir workon the project. Graduate students can work on multiple projects, in which casetheywill havea(potentiallydi_erent)supervisorforeachone.
- Departmentshaveadepartmentnumber,adepartmentname,andamainoffice.
- Departmentshaveaprofessor(knownasthechairman)whorunsthedepartment.
- Professorsworkinoneormoredepartments,andforeachdepartmentthattheyworkin,atimepercentageisassociatedwiththeirjob.
- Graduatestudentshaveonemajordepartmentinwhichtheyareworkingontheirdegree.
    - Eachgraduatestudenthasanother,moreseniorgraduatestudent(knownasastudentadvisor)whoadviseshimorheronwhatcoursestotake.

WhatisSQLandSQL*Plus

Oracle was the first company to release a product that used the English-based Structured QueryLanguage or SQL. This language allows end users to manipulate information of table (primarydatabaseobject). TouseSQL you need not torequireanyprogramming experience. SQL is astandard language common to all relational databases. SQL is database language used for storingand retrieving data from the database. Most Relational Database Management Systems provideextension to SQL to make it easier for application developer. A table is a primary object ofdatabaseusedtostoredata.Itstoresdatainformofrowsandcolumns.

SQL*PlusisanOracletool(specificprogram)whichacceptsSQLcommandsandPL/SQLblocksand executesthem.SQL*Plusenables manipulationsofSQLcommandsandPL/SQLblocks.Italso performs additional tasks such as calculations, store and print query results in the form o reports,list columndefinitionsofanytable,accessandcopydatabetweenSQLdatabasesandsend messages to and accept responses from the user. SQL *Plus is a character based interactivetool,thatrunsinaGUIenvironment.Itisloadedontheclientmachine.

TocommunicatewithOracle,SQLsupportsthefollowingcategoriesofcommands:

1. DataDefinitionLanguage

Create,Alter,DropandTruncate

2. DataManipulationLanguage

Insert,Update,DeleteandSelect

3. TransactionControlLanguage

Commit,RollbackandSavepoint

4. DataControlLanguage

GrantandRevoke

Beforewetakea lookonabove-mentionedcommands wewillseethedata types available inOracle.

OracleInternalDatatypes

WhenyoucreateatableinOracle,afewitemsshouldbeimportant,notonlydoyouhavetogiveeach tableaname(e.g.employee,customer),you mustalsolistallthecolumnsorfields (e.g.First_name, Mname, Last_name) associated with the table.You also have to specify what type ofinformation thattable will hold to the database. For example, the column Empno holds numericinformation.AnOracledatabasecanholdmanydifferenttypesofdata.

DatatypeDescription

Char(Size)Storesfixed-lengthcharacterdatatostore alphanumericvalues,withamaximumsizeof2000bytes.Defaultandminimumsizeis1byte.

Varchar2(Size)Storesvariable-lengthcharacterdatatostore alphanumeric values,withmaximumsizeof 4000bytes.

char(Size)Storesfixed-lengthcharacterdataoflengthsizecharactersorbytes,dependingon the choice of national character set. Maximumsize if determined by the number of bytesrequired storing each character with an upper limit of 2000 bytes. Default and minimum size is 1characteror1byte,dependingonthecharacterset.

Nvarchar2(Size) Storesvariable-lengthcharacterstringhavingmaximumlengthsizecharactersor bytes,dependingonthe choice of national character set. Maximumsize isdeterminedbythenumberofbytesrequiredtostoreeachcharacter,withanupperlimit of4000bytes.

LongStoresvariable-lengthcharacterdataupto2GB(Gigabytes).Itslenthwouldbe restrictedbasedonmemoryspaceavailableinthecomputer.

Number [p,s] Number having precision p and scale s. The precision p indicates totalnumberofdigitvariesfrom1to38.Thescalesindicatesnumber ofdigitinfraction partvariesfrom-84to127.

DateStoresdatesfromJanuary1,4712B.C.toDecember31,4712A.D.Oracle predefineformatofDatedatatypeisDD-MON-YYYY.

Raw (Size) Stores binary data of length size. Maximum size is 2000 bytes. One musthavetospecifysizewithRAWtypedata, becausebydefault it does not specifyany size.**LongRawStorebinarydataof variablelengthup to2GB(Gigabytes).**

LOBS-LARGEOBJECTS

LOBis usetostoreunstructuredinformationsuchassoundandvideoclips,pictures upto4 GBsize.

CLOBACharacterLargeObjectcontainingfixed-widthmulti-byte characters.Varying-

widthcharactersetsarenotsupported.Maximumsizeis4GB.

NCLOBANationalCharacterLargeObjectcontainingfixed-widthmulti- bytecharacters.

Varying-

widthcharactersetsarenotsupported.Maximumsizeis4GB.Storesnationalcharacter setdata.

BLOBTostoreaBinaryLargeObjectsuchagraphics,videoclipsandsoundfiles.

Maximumsizeis4GB.

BFILEContainsalocatortoalarge BinaryFilestoredoutsidethedatabase.

EnablesbytestreamI/OaccesstoexternalLOBsresidingonthedatabaseserver.Maximum sizeis4GB.Apartfromoracleinternaldatatypes,usercancreatetheirowndatatype,whichisusedindata baseandotherdatabaseobject.Wewilldiscussitinthelaterpart.

Thefollowingaretabularrepresentationoftheaboveentitiesandrelationships

BUS:

COLUMNNAME	DATATYPE	CONSTRAINT
BusNo	varchar2(10)	**PrimaryKey**
Source	varchar2(20)	
Destination	varchar2(20)	
CouchType	varchar2(20)	

Reservation:

COLUMNNAME	**DATATYPE**	**CONSTRAINT**
PNRNo	number(9)	**PrimaryKey**
Journeydate	Date	
No-of-seats	integer(8)	
Address	varchar2(50)	
ContactNo	Number(9)	Shouldbeequalto10 numbersandnotallow otherthannumeric
BusNo	varchar2(10)	**Foreignkey**
Seatno	Number	

Ticket:

COLUMNNAME	**DATATYPE**	**CONSTRAINT**
Ticket_No	number(9)	**PrimaryKey**
Journeydate	Date	
Age	int(4)	
Sex	Char(10)	
Source	varchar2(10)	
Destination	varchar2(10)	
Dep-time	varchar2(10)	
BusNo	Number2(10)	

Passenger:

COLUMNNAME	**DATATYPE**	**CONSTRAINT**
PNRNo	Number(9)	**PrimaryKey**
TicketNo	Number(9)	Foreignkey
Name	varchar2(15)	
Age	integer(4)	
Sex	char(10)	(Male/Female)
Contactno	Number(9)	Shouldbeequalto10numbers andnotallowotherthan numeric

Cancellation:

COLUMNNAME	**DATATYPE**	**CONSTRAINT**
PNRNo	Number(9)	Foriegn-key
Journey-date	Date	
Seatno	Integer(9)	
Contact_No	Number(9)	Shouldbeequalto10numbers andnotallowotherthan numeric

Facultysignature

WEEK 2

ConceptdesignwithE-RModelandapplycardinalitiesforeachrelationship.Identifystrong entities and weak entities for relationships like generalization, aggregation, specialization.

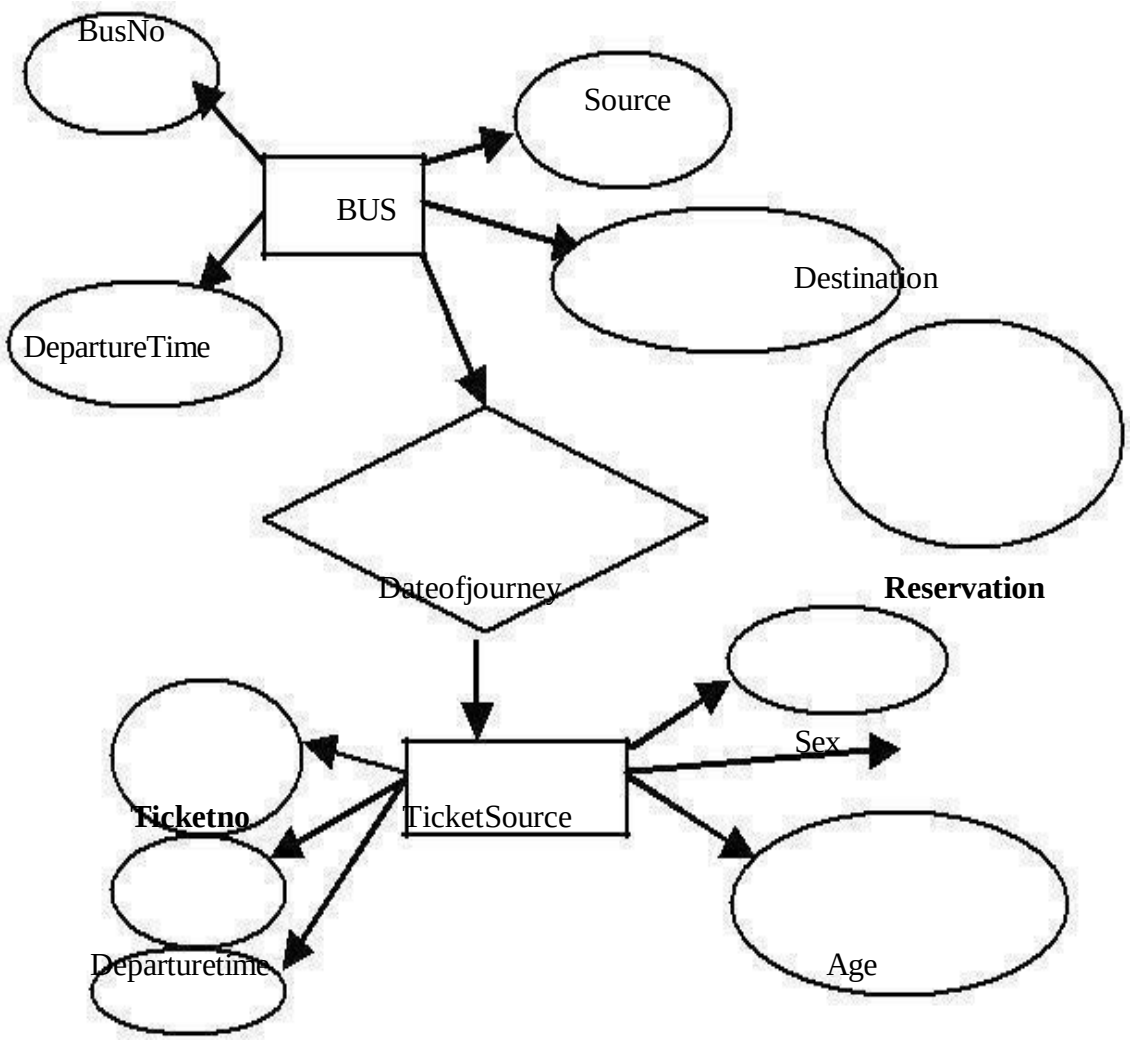
Objectives:

Studentwillabletolearndatastructuresintermsofentitytypes, relationship typesandattributesorclasses, associations and attributes.

Outcomes:

Studentgainstheabilitytodescribethedatarequirementsfor anew informationsysteminadirect and easyto understand graphical notation.

E_Rdiagram:



VIVAQUESTIONS

1. DrawanE-RDiagramForanATMSystem.
2. DrawanE-RDiagramFor schoolmgmtsystem.
3. DrawanE-RDiagramForRoadwaysTravels Systems.
4. DrawanE-RDiagramForBankMgmtSystem.
5. Explainmanytomanyandmanytoonerelationship.

Facultysignature

WEEK3

AIM

RelationModelrepresentsattributesascolumnsintablesanddifferenttypes of attributes like composite, Multi-valued and Derived.

Objectives:

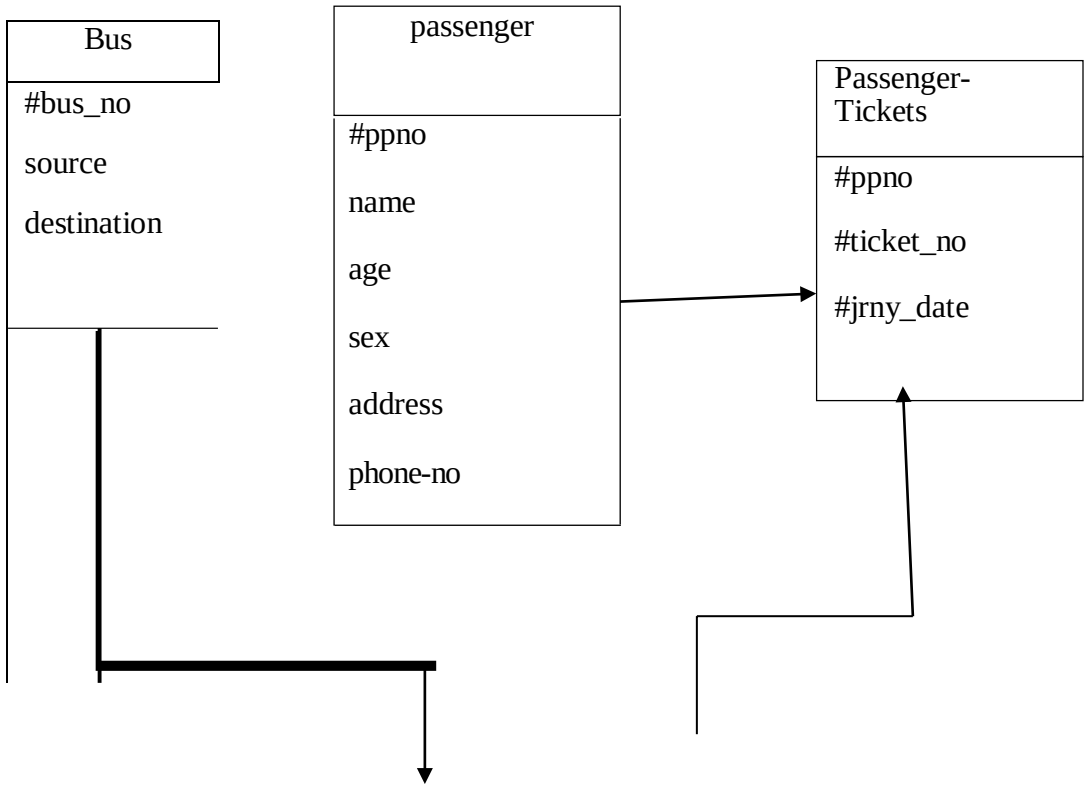
Studentwillabletolearnthestructuralcomponentsoftherelationaldatamodel. Student willable to learnto map ER models into relational models.

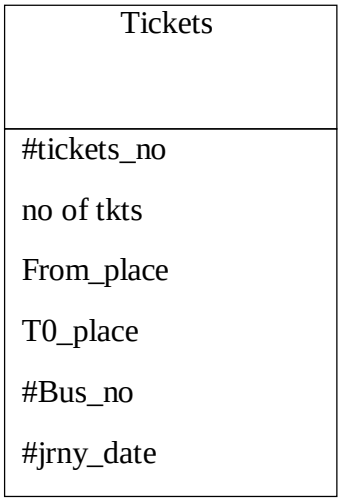
Outcomes:

Studentgainstheability

- TodescribetheModelStructure.
- TodefinePropertiesofRelations.
- TodefineDomains.
- ToimplementNotationtoDescribetheRelationalSchema
- ToRepresentanERModelasaRelationalModel.

Example:Thepassengertableslookasbelow.Thisisanexample.Youcanaddmoreattributes based on your E-Rmodel. This is not a normalized table.Passenger





Name	Age	Sex	Address	Passport ID

Note: Thestudentisrequiredtosubmitadocument byRepresent relationshipsinatabularfashionto the lab teacher.

2. ConceptdesignwithE-Rmodel

Relatetheentitiesappropriateforeachrelationship. Identifystrongentitiesandweekentities(ifany). Indicate the type ofrelationships (total/partial). In this we will design the different E-R diagram for differententities and also the whole “RoadwayTravels”.

E-R diagram: An entity-relationship(ER) diagram is a specified graphic that illustrates the interrelationshipsbetweenentitiesand database.Wecanexpresstheoverallogicalstructureof database graphically with an E-R diagram.

3. RelationalModelandNormalization

Representalltheentities(Strong,Weak)intabularfashion.Representrelationshipsinatabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement. In this we will represent the differententities, attributesofdifferentkeys ina tabularfashionor manner.

RelationalModel:

Therelational modelis a depictionofhow eachpiece ofstored informationrelatesto theother stored information. It showshowtablesare linked, what type ofthe linksare betweentables, what keysare used,what informationisreferenced betweentables.It'sanessentialpartofdevelopinga normalized database structureto prevent repeat and redundant data storage.

Differenttypesofkeys:

- Asuperkeyisasetofoneormoreattributeswhich;takencollectively, allowusto identify uniquely an entity in the entityset.
- Aprimarykeyisa candidate key(there maybemorethanone) chosen bythe DBdesigner to identify entitiesin an entityset.
- Asuperkeymaycontainextraneousattributes,andweareofteninterestedinthesmallest super key.Asuper keyfor whichnosubsetis a super key is called a candidate key.
- Anentitydoesnotpossessufficient attributesto formaprimaryket iscalledaweakentityset. One that does have a primarykey is calleda strong entityset.
- Aforeignkeyisafield inarelationaltablethat matchestheprimarykeycolumnofanother table. The foreign keycan be used to cross-reference tables.

Normalization

Databasenormalizationisatechniquefordesigningrelationaldatabasetablestominimizeduplicationof information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. In this we will write the normalization tables that is entities of "RoadwayTravels."

Normalization: Inrelationaldatabases,normalizationisaprocessthateliminatesredundancy,organizes data efficiently; Normalization is the process of efficientlyorganizing data in a database. There are two goalsofthenormalizationprocess:eliminatingredundantdata(forexample,storingthesamedatainmore thanonetable) andensuringdatadependencies makesense(onlystoringrelateddata inatablet). Bothof these are worthy goals as they reduce the amt of space a database consumes and ensure that data is logicallystores.

The Normal Form: the database community has developed a series of guidelines for ensuring that databasesarenormalized. Theseareferred to asnormalformsand arenumbered fromone(thelowest formtonormalization,referredtoasfirstformorINF)throughfive(fifthnormalformofSNF).Inpractical applications,you'lloftensee INF, 2NF, and 3NF along withoccasional4NF. Fifthnormal formis very rarelyseenand won't be discussed inthis article. It's important to point outthat theyare guidelines and guidelines only. Occasionally, it becomes necessary to stray from them to meet practical business requirements. However, when variations take place, it's extremely important to evaluate any possible requirements they could have on your system and account for possible inconsistencies. That said, let's explore the normal form.

VIVAQUESTIONS

1. What isrelationalmodelanditsimportance.
2. Explainthedifferencebetweencandidatekeyandprimarykey.
3. Whatisasuperkey.
4. Differentiateamongalltypesofkeyswithexample.
5. Explaintheneedofforeignkey.

Facultysignature

WEEK4

AIM

Normalizationoftables

Objectives:

Studentwillabletolearntoavoidproblemsthatareassociatedwithupdatingredundantdata.

Outcomes:

StudentgainstheknowledgetobuildThedatabasethatdoesnothaveredundantdata.

Abasicobjectiveofthefirst normalformdefined byEdgarFrank"Ted"Coddin1970wasto permit data to be queried and manipulated using a "universal data sub-language" grounded in first-order logic.(SQL is an example of such a data sub-language, albeit one that Codd regarded as seriously flawed.)

Theobjectivesofnormalizationbeyond 1NF(FirstNormalForm)werestatedasfollowsbyCodd:

1. Tofreethecollectionofrelationsfromundesirable insertion,updateanddeletion dependencies;
2. Toreducetheneed forrestructuring thecollectionofrelations, asnewtypesofdataare introduced,andthus increase the life spanofapplication programs;
3. Tomaketherelationalmodelmoreinformativetousers;
4. Tomakethecollectionofrelationsneutraltothequerystatistics,wherethesestatistics are liable to change as time goes by.

Querying and manipulating the data within a data structure which is not normalized, such as the followingnon-1NFrepresentationofcustomers'creditcardtransactions, involvesmorecomplexitythan is really necessary:

Customer	**Transactions**		
	Tr.ID	**Date**	**Amount**
Jones	12890	14-Oct-2003	−87
	12904	15-Oct-2003	−50
	r.ID	**Date**	**Amount**
Wilkinso	12898	14-Oct-2003	−21
	Tr.ID	**Date**	**Amount**
Stevens	12907	15-Oct-2003	−18
	14920	20-Nov-2003	−70
	15003	27-Nov-2003	−60

Toeachcustomercorrespondsarepeatinggroupoftransactions.Theautomatedevaluation ofanyqueryrelatingtocustomers'transactionsthereforewouldbroadlyinvolvetwo stages:

1. Unpackingoneormorecustomers'groupsoftransactionsallowingtheindividual transactions ina groupto be examined, and
2. Derivingaqueryresultbasedontheresultsofthefirststage

For example, in order to find out the monetarysumofall transactions that occurred in October2003forallcustomers,thesystemwouldhavetoknowthat it mustfirstunpack the*Transactions* groupofeachcustomer, thensumthe *Amounts*ofalltransactions thus obtained where the *Date* ofthe transaction fallsin October 2003.

OneofCodd's importantinsightswasthatthisstructuralcomplexitycouldalways be removedcompletely,leadingtomuchgreaterpower andflexibilityinthewayqueries could be formulated (byusers and applications) and evaluated (bythe DBMS). The normalized equivalent ofthe structure abovewouldlook like this:

Customer	Tr.ID	Date	Amount
Jones	1289	14-Oct-2003	−87
Jones	1290	415-Oct-2003	−50
Wilkins	1289	814-Oct-2003	−21
Stevens	1290	715-Oct-2003	−18
Stevens	1492	020-Nov-2003	−70
Stevens	1500	327-Nov-2003	−60

Now each row represents an individualcredit card transaction, and the DBMS can obtain the answer of interest, simply by finding allrows with a Date falling in October, and summing their Amounts. The data structure places all of the values on an equal footing, exposingeachtotheDBMSdirectly, so eachcanpotentiallyparticipatedirectlyinqueries; whereas inthe previous situationsome values were embeddedin lower-levelstructures thathadtobehandledspecially.Accordingly,thenormalizeddesignlendsitselftogeneral-purpose queryprocessing, whereas the unnormalized design does not.

VIVAQUESTIONS

1. Explaintheneedofnormalization?
2. What isfunctionaldependency?
3. Explaindifferencebetweenthirdnormalformandboycecoddnormalform?
4. WhatisPJNF?
5. What istransitivitydependency?

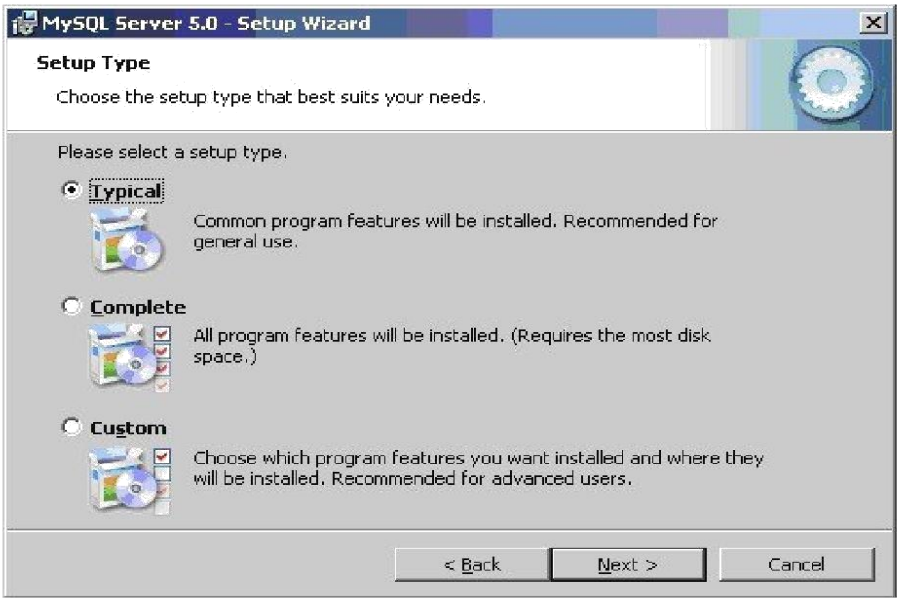
WEEK-5(a)

AIM:InstallationofMySQLandpracticingDDL&DMLcommands.

1.StepsforinstallingMySQL

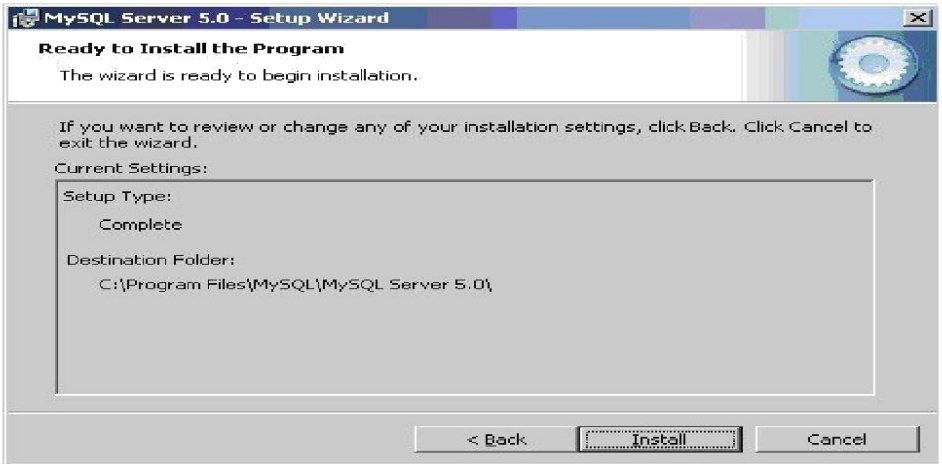
Step1
Makesureyou alreadydownloadedthe**MySQL essential5.0.45win32.msifile**. Doubleclick on the .msifile.

Step2
ThisisMySQLServer 5.0setupwizard.ThesetupwizardwillinstallMySQLServer5.0release5.0.45onyour computer. To continue, click **next.**



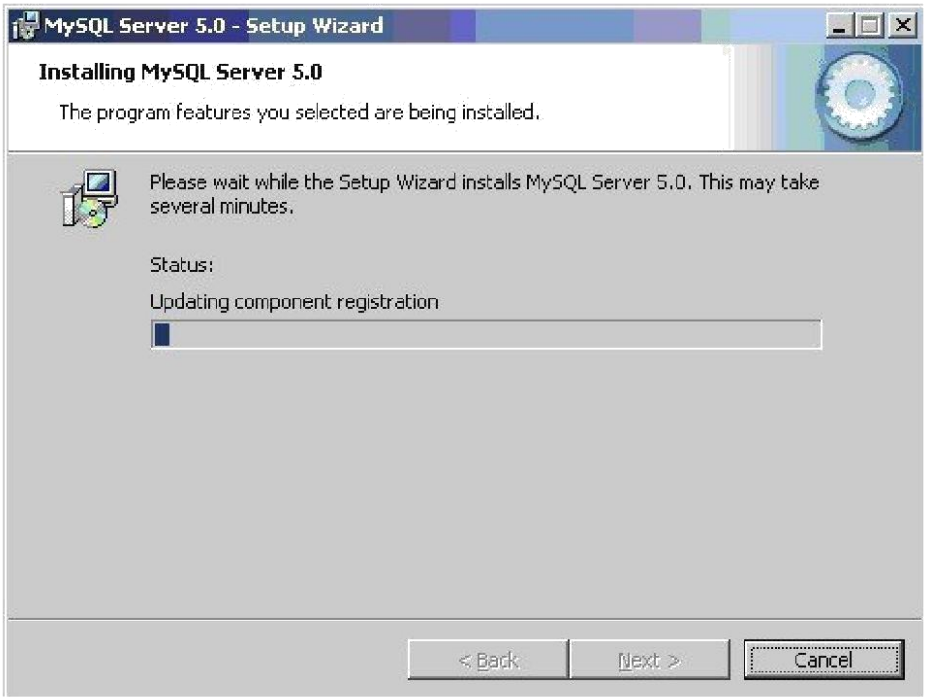
Step3

Choosethesetuptypethatbestsuitsyourneeds.Forcommonprogramfeatures select ***Typical***andit'srecommendedforgeneraluse. Tocontinue, click**next**.



Step4

Thiswizardisreadytobegininstallation.Destinationfolderwillbein **C:\ProgramFiles\MySQL\MySQLServer5.0**.Tocontinue,click**next**.

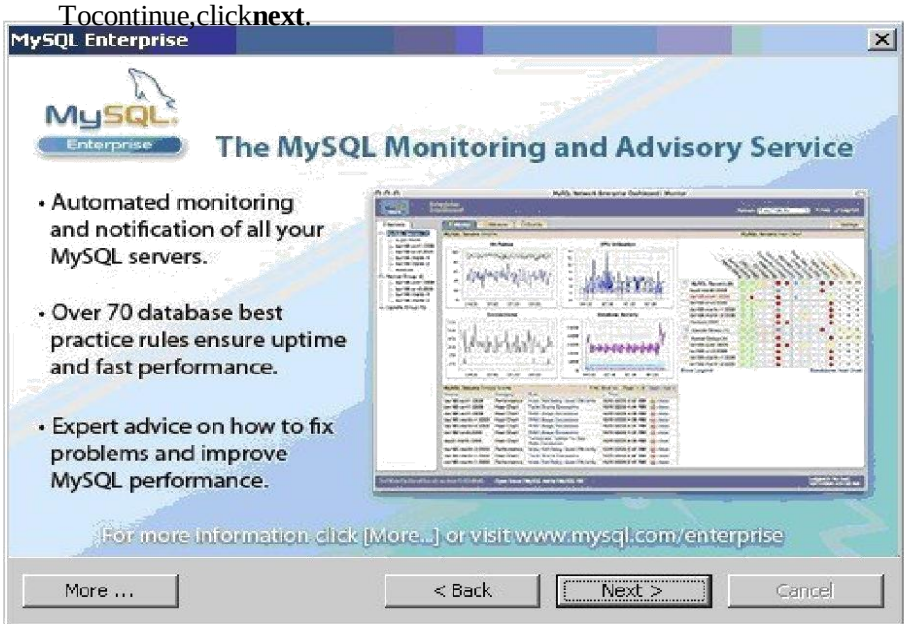


Step5

Theprogramfeaturesyouselectedarebeinginstalled.PleasewaitwhilethesetupwizardinstallsMySQL5.0.Thismaytakeseveralminutes.

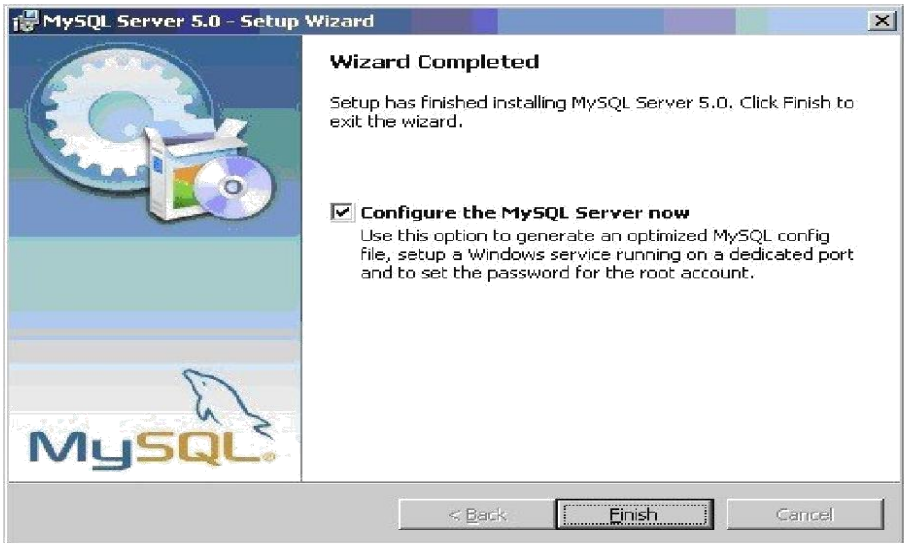
Step6

Tocontinue,click**next**.



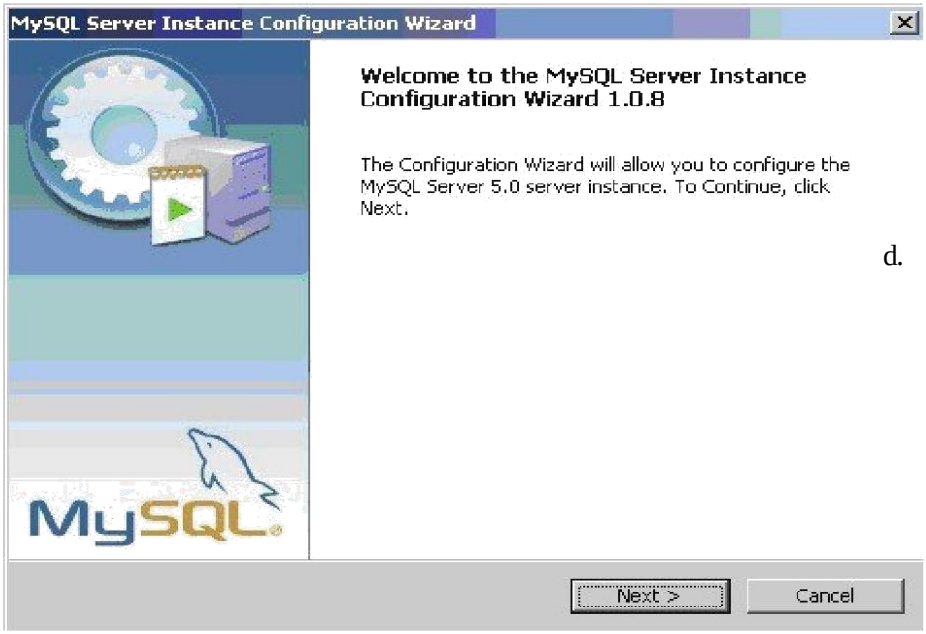
Step7

Tocontinue,click**next**.



Step8

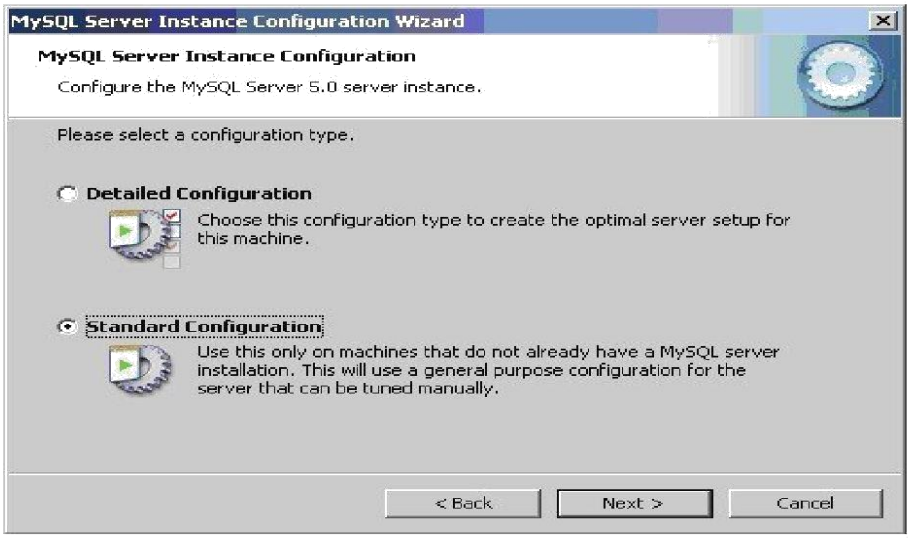
WizardCompleted.SetuphasfinishedinstallingMySQL5.0. **Check**theconfigurethe MySQL servernow to continue. Click **Finish** to exit the wizard



d.

Step9

Theconfiguration wizardwillallow youtoconfigurethe MySQLServer5.0serverinstance.Tocontinue,click**next**.

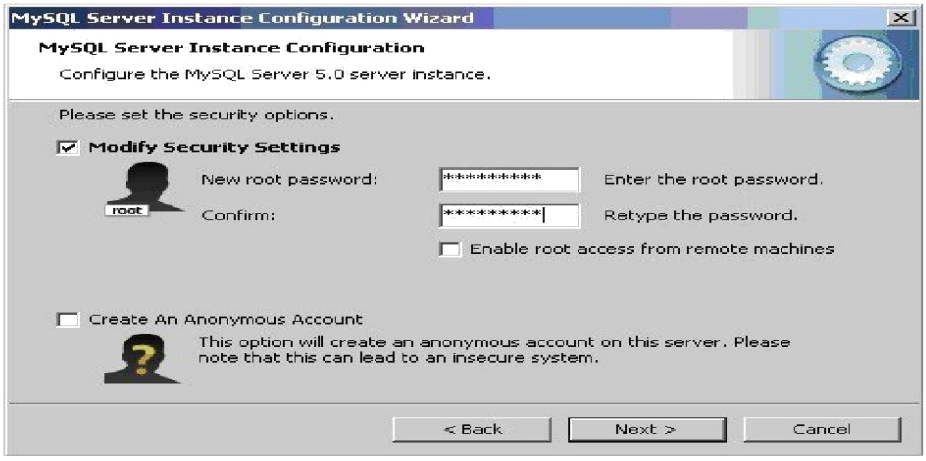


Step10

Selecta **standardconfiguration**andthis willusea generalpurposeconfigurationfor theserver thatcanbetuned manually.To continue,click **next**.

Step11

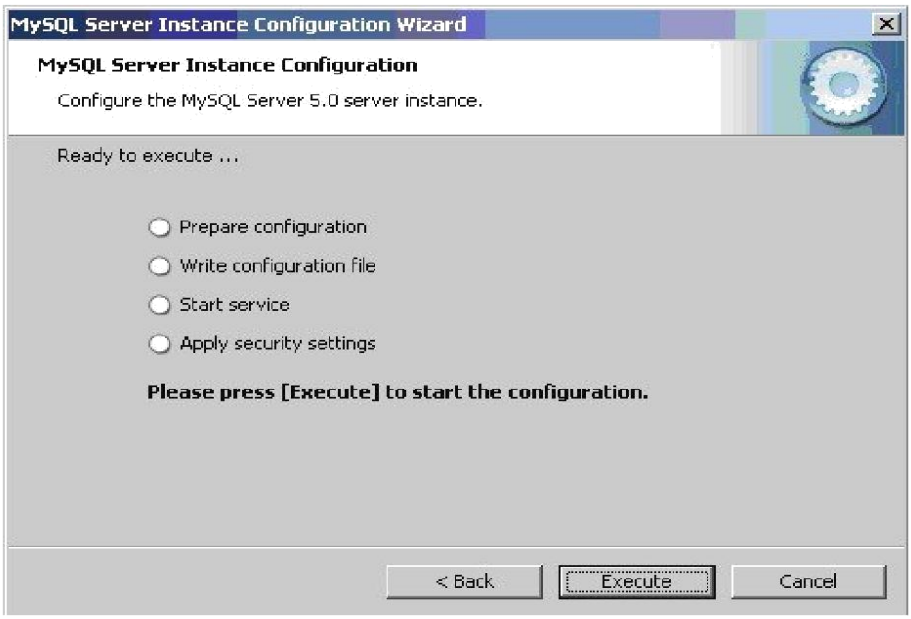
Checkonthe**installaswindowsservice**and**includebindirectory inwindowspath**. Tocontinue,click**next**.



Step12

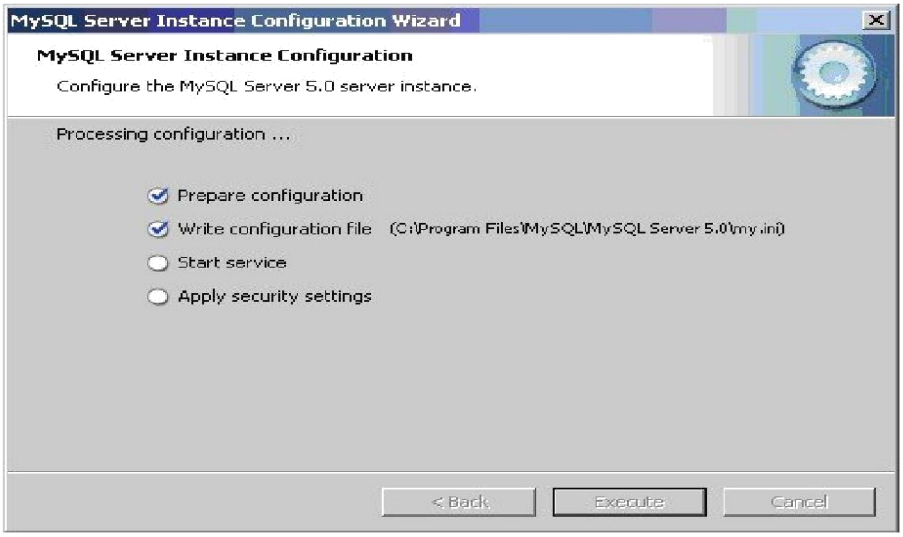
Pleasesetthesecurityoptionsbyenteringtherootpasswordandconfirmretypethepassword.

continue,clicknext.



Step13

Readytoexecute?Clicks**execute**tocontinue.

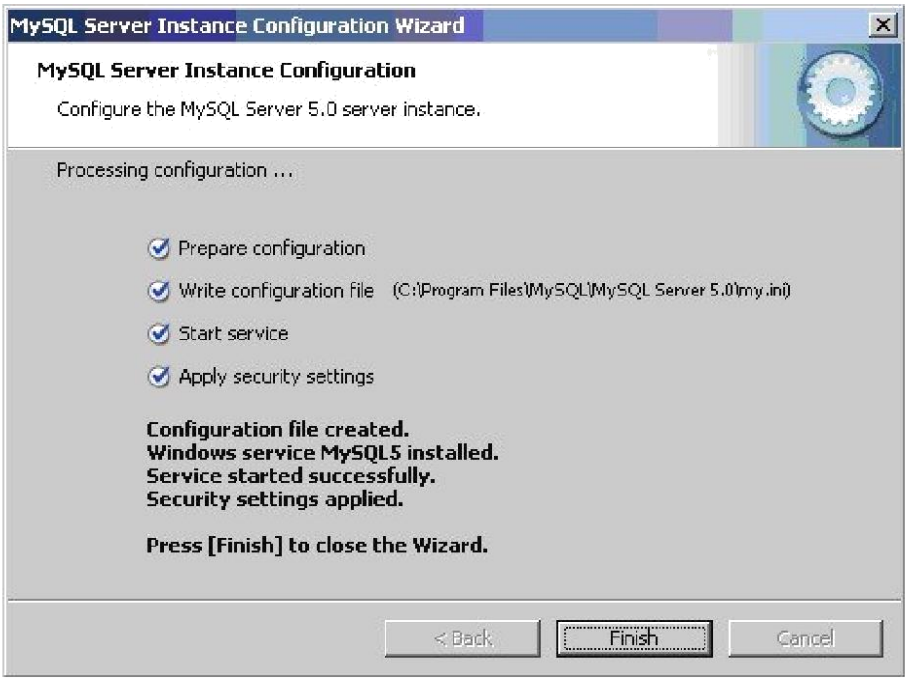


Step14

Processingconfigurationinprogress.

Step15

Configurationfilecreated.WindowsserviceMySQL5installed.Press**finish**toclose thewizard.



WEEK-5

(b)

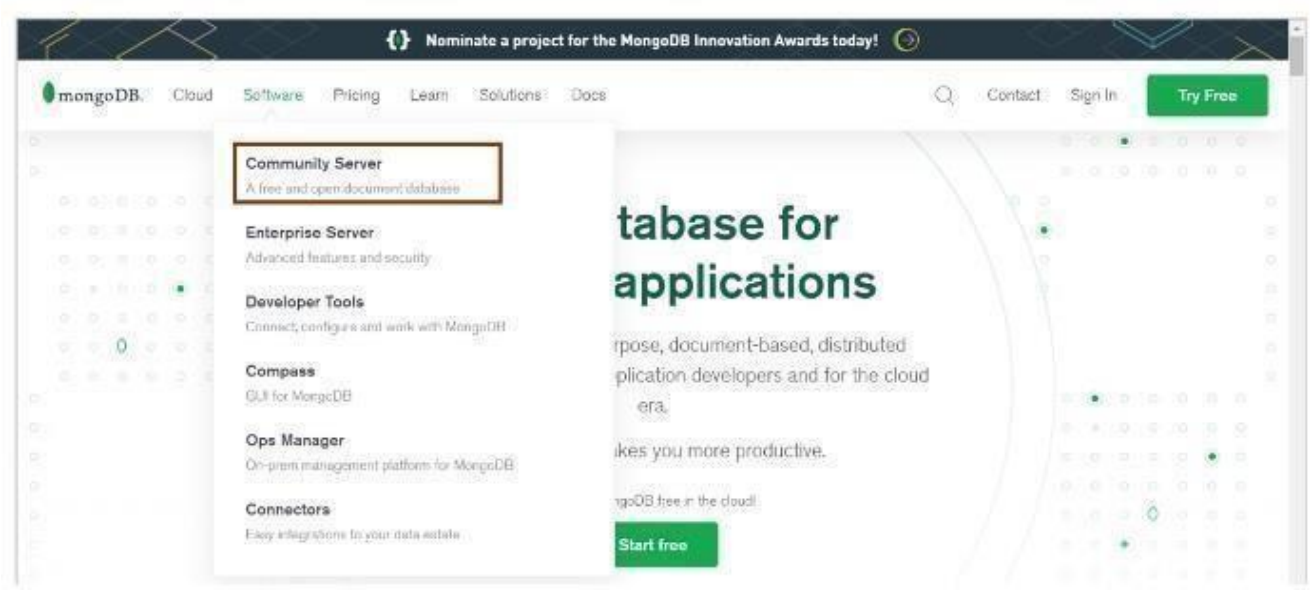
MongoDB

Navigatetothedownloadsite:

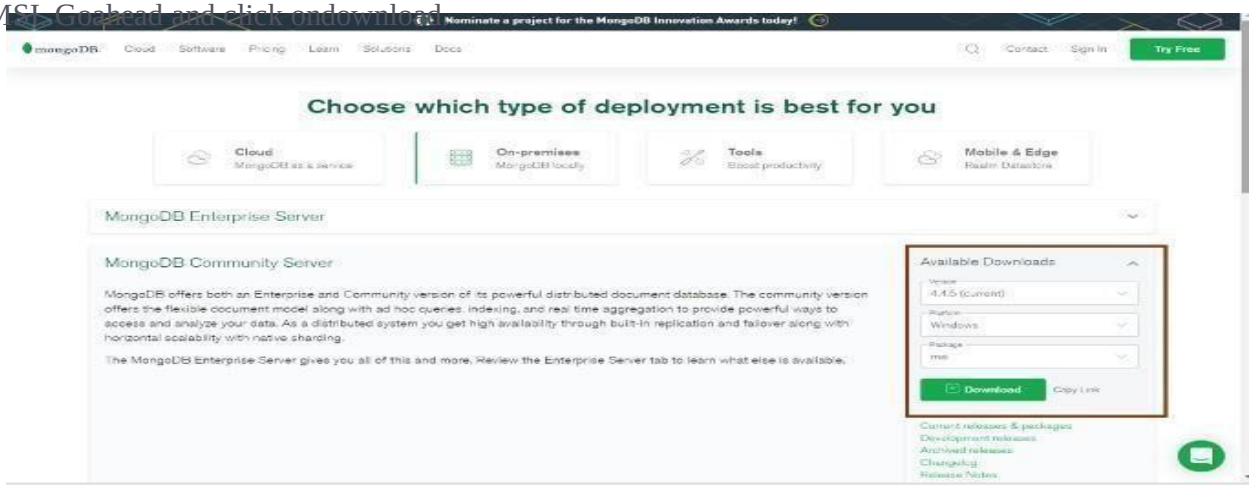
NavigatetotheofficialMongoDB[website](https://www.mongodb.com/)https://www.mongodb.com/

Cross-checktheSpecificationsandDownloadMongoDB

UndertheSoftwaresection,clickontheCommunityserverversion.

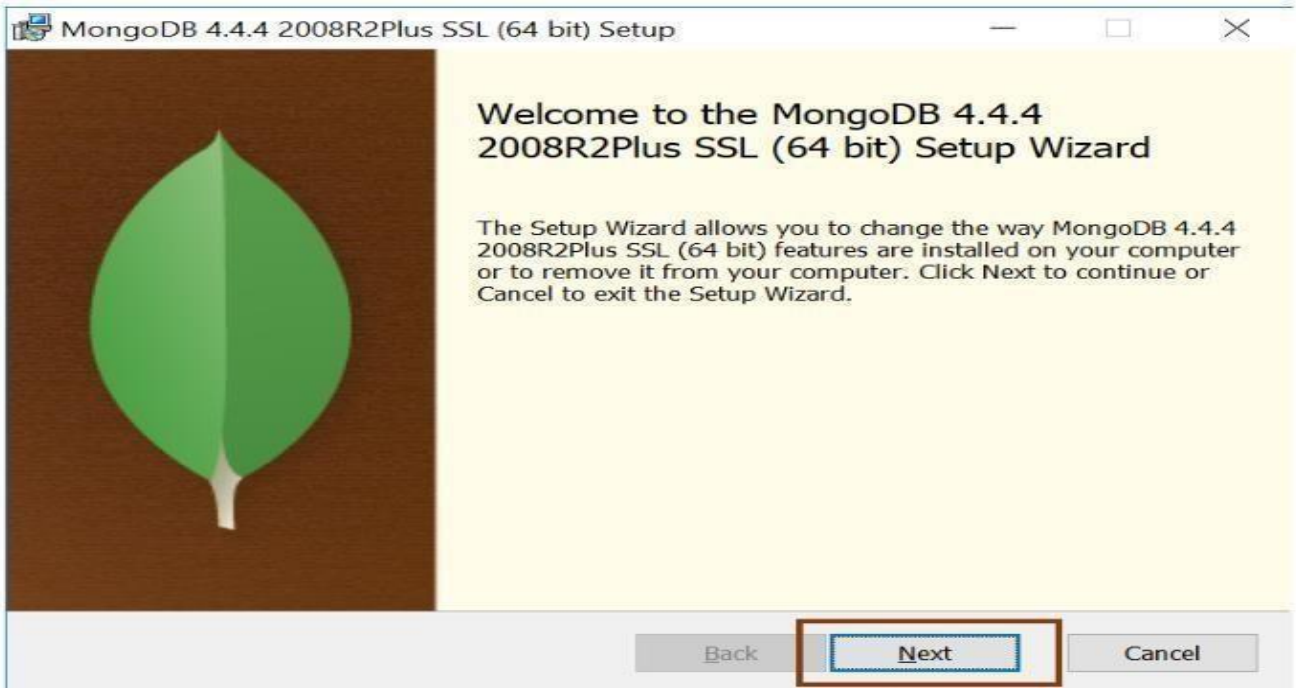


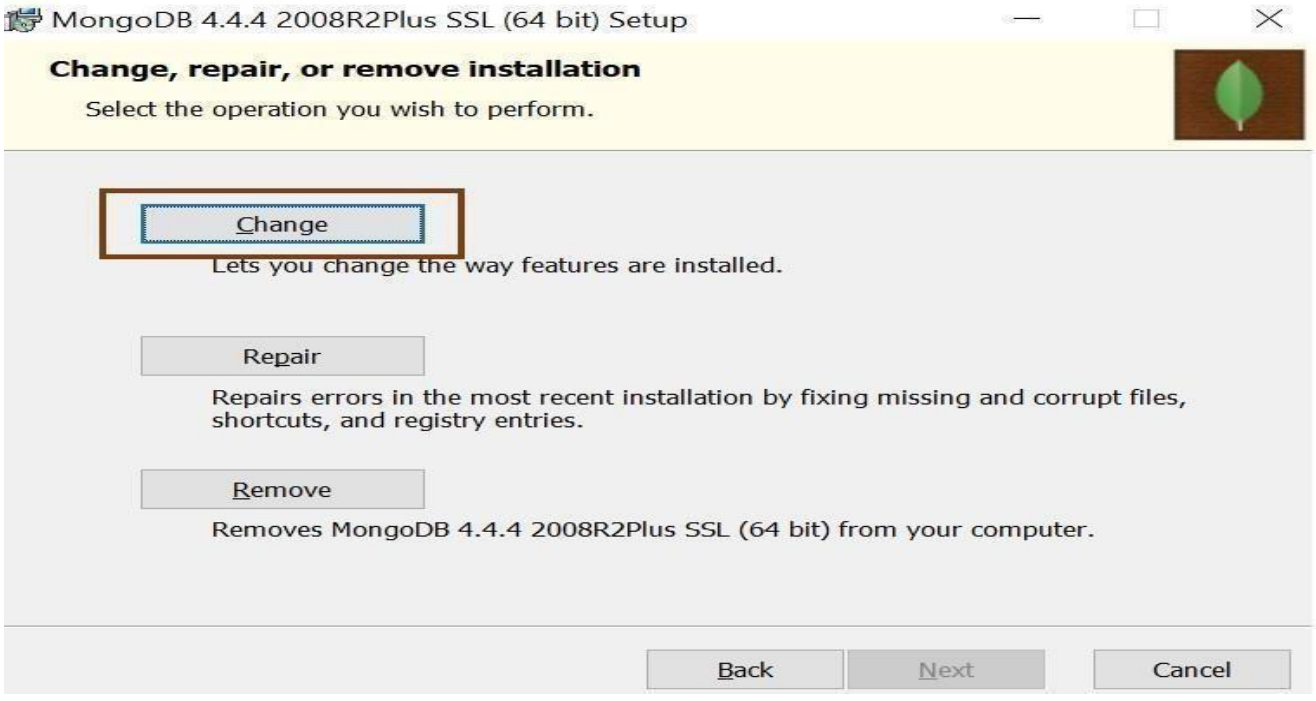
Atthetimeofwriting,the latest versionis4.4.5.Ensurethattheplatform is Windows,andthepackage is MSI.Gocheadandclick ondownload.



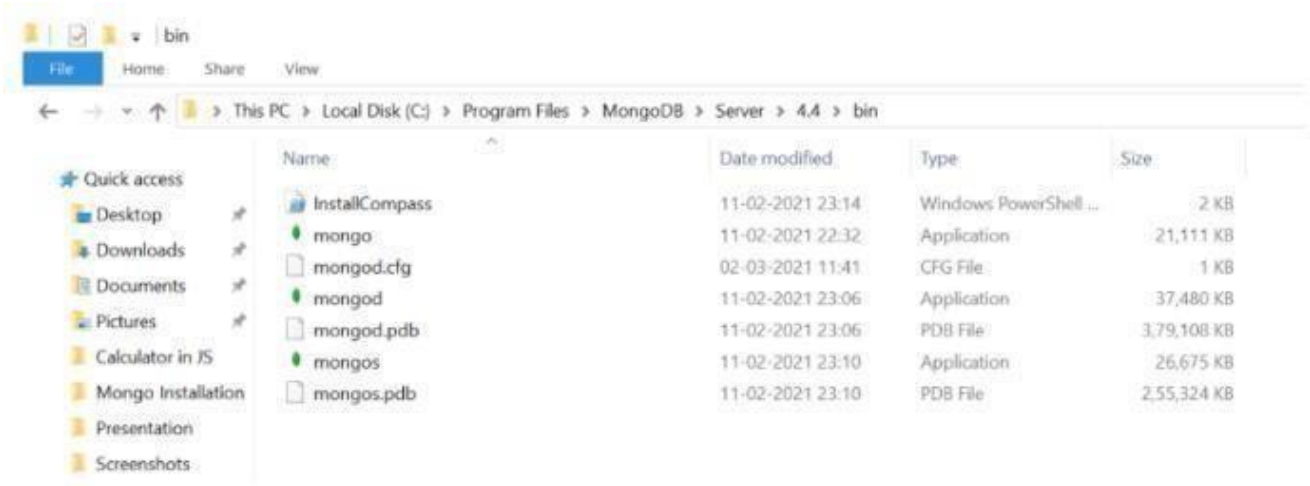
MongoDBInstallation:

Youcanfindthedownloadedfilein thedownloadsdirectory.You canfollowthesteps mentioned there and install the software.





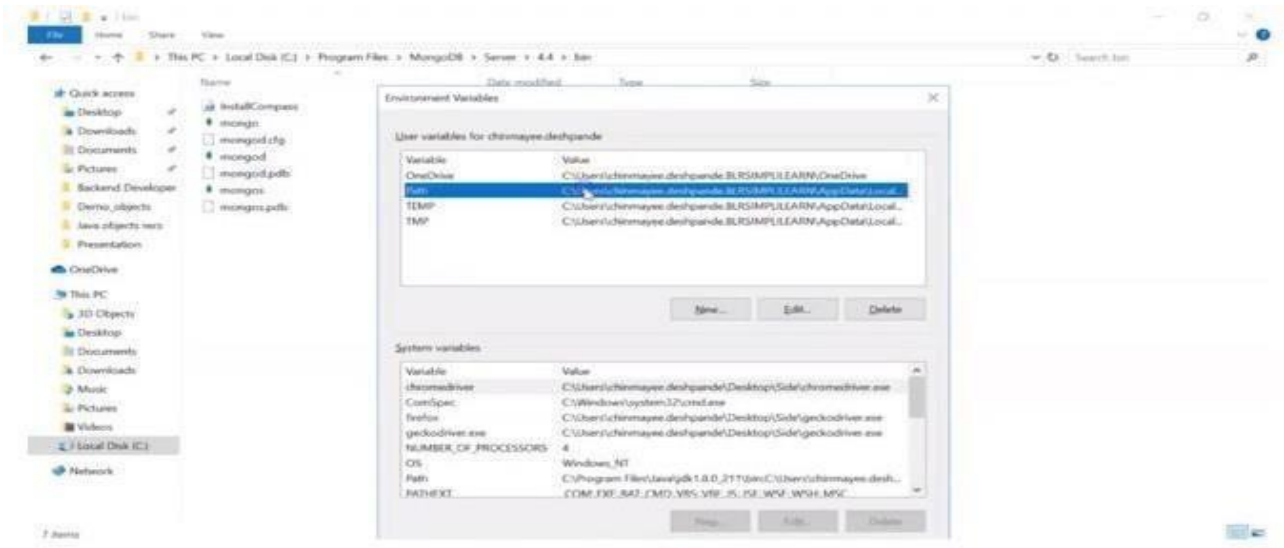
Oncompletingtheinstallation successfully,youwill findthesoftwarepackageinyourC drive. C:\ProgramFiles\MongoDB\Server\4.4\bin.

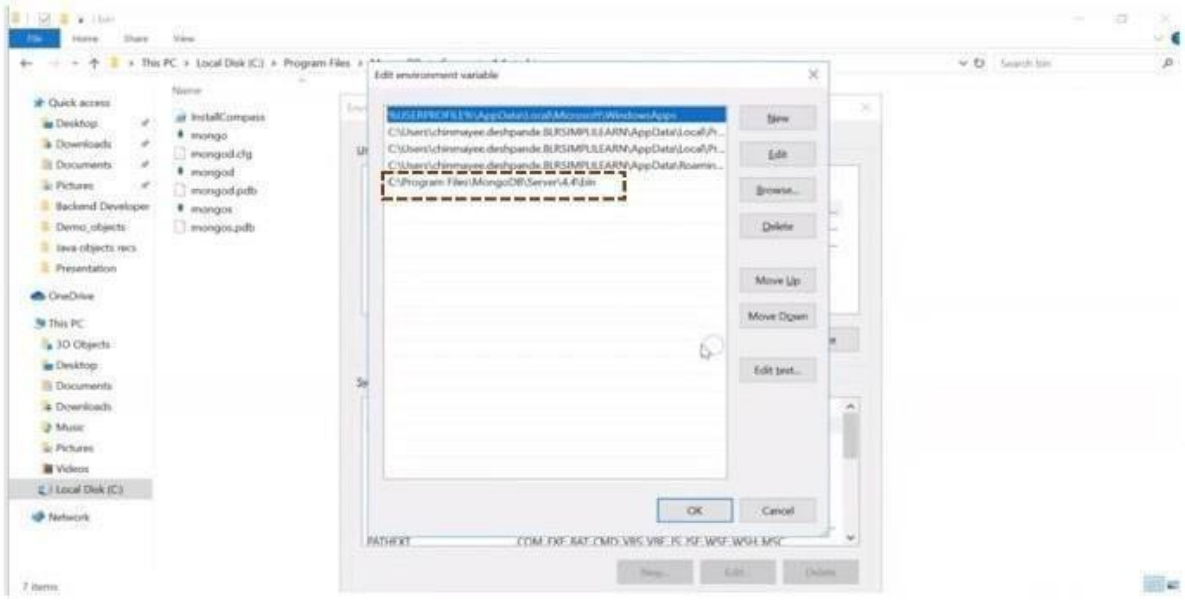


Youcanseethattherearemongoandmongodexecutablefiles. Themongodfileisthedaemon process thatdoesthe background jobs like accessing,retrieving,and updating the database.

CreateanEnvironmentVariable:

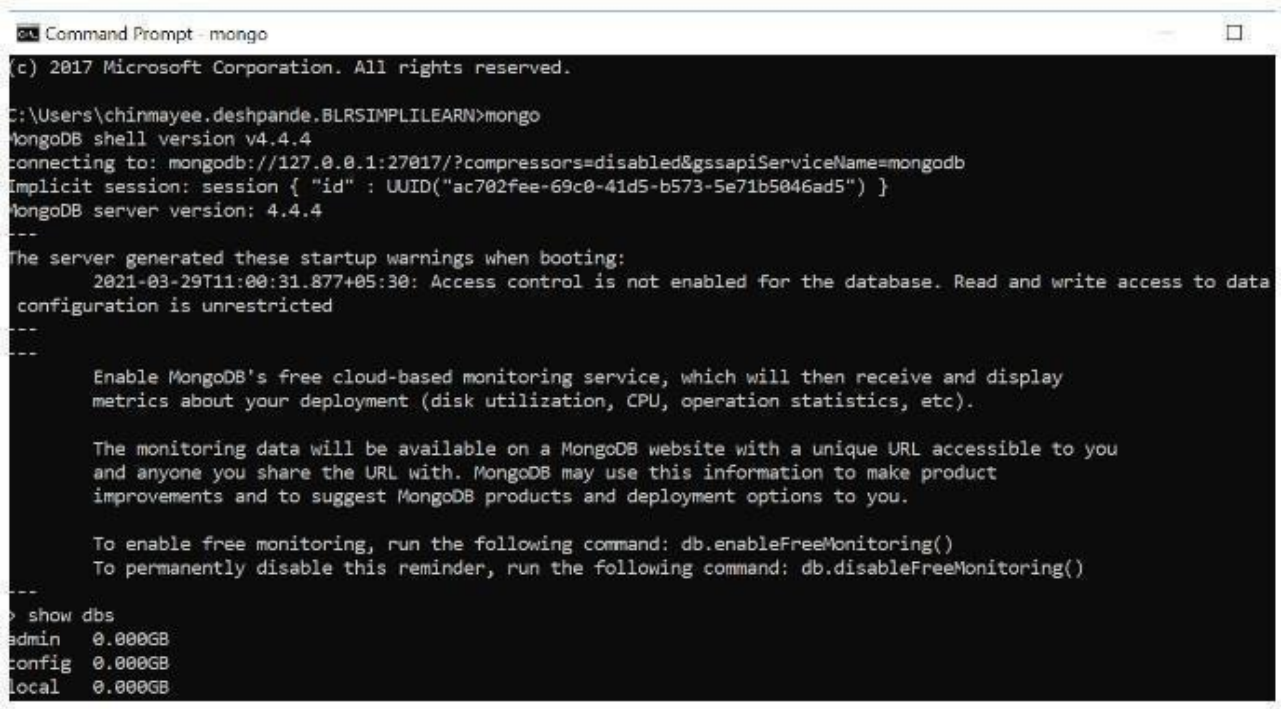
It’sbestpracticetocreateanenvironmentvariablefortheexecutablefilesothatyou don’t have to change the directorystructure everytime youwantto execute thefile.





ExecutetheMongoApp:

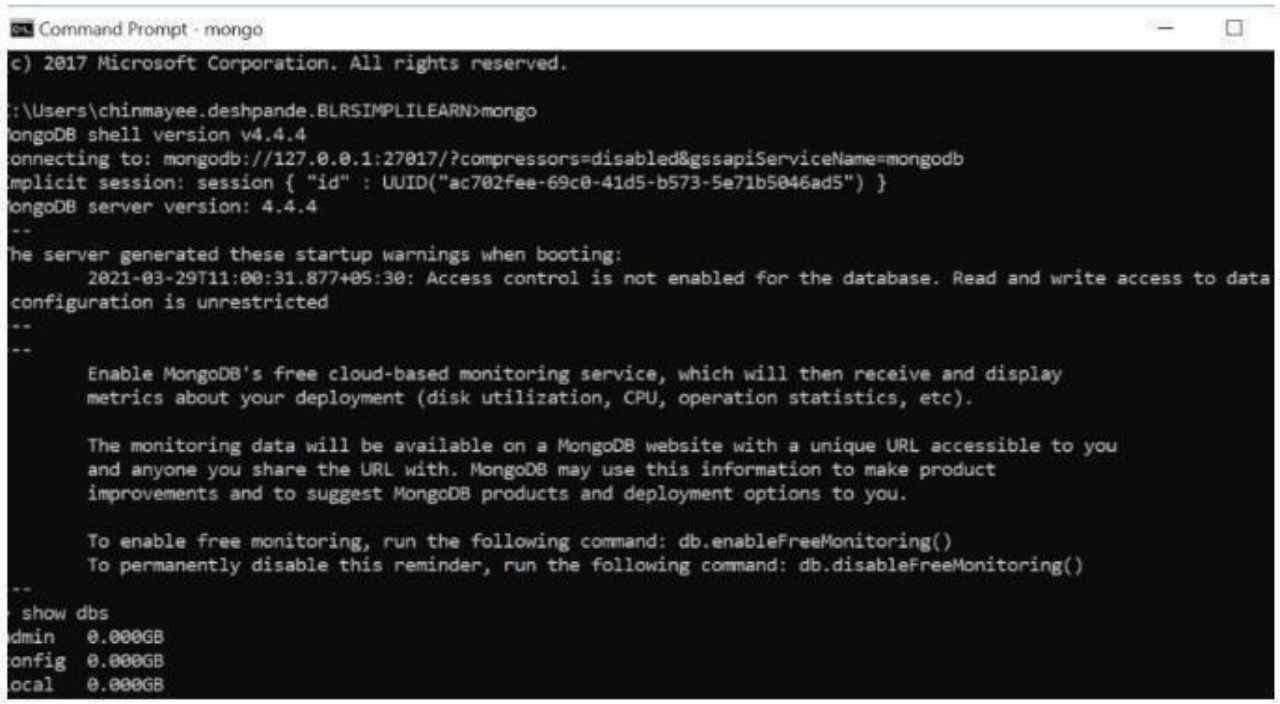
Aftercreatinganenvironment path,youcanopenthecommandprompt andjust type in mongo and press enter.



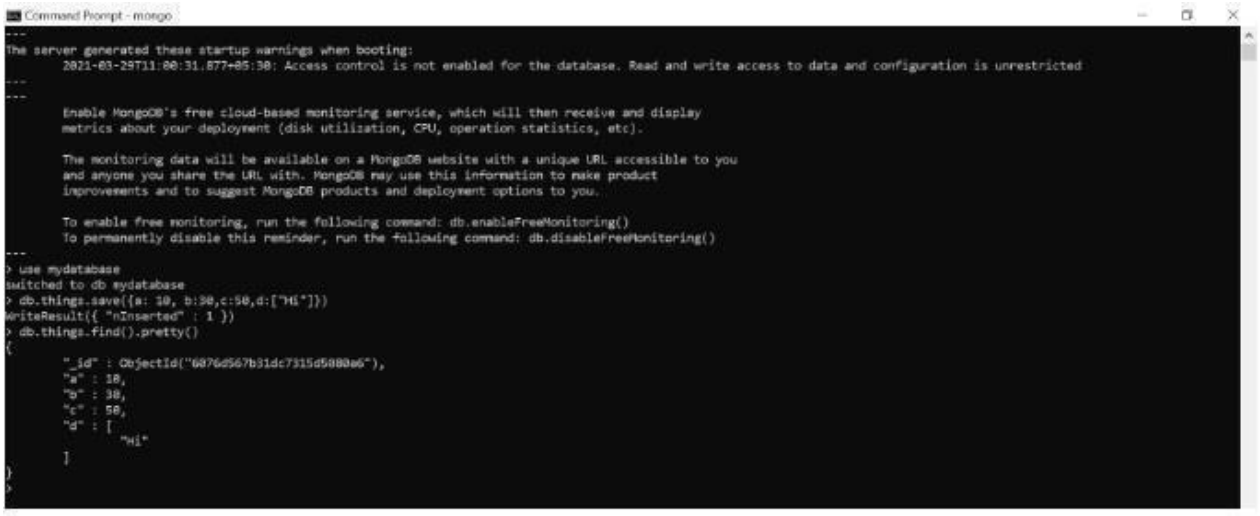
Themongoserveristhengeneratedandisupand running.

VerifytheSetup

Toverifyifitdidthesetupcorrectly,typeinthecommandshowDBS.



Withthat,youhavesuccessfullyinstalledandsetupMongoDBonyourWindowssystem.



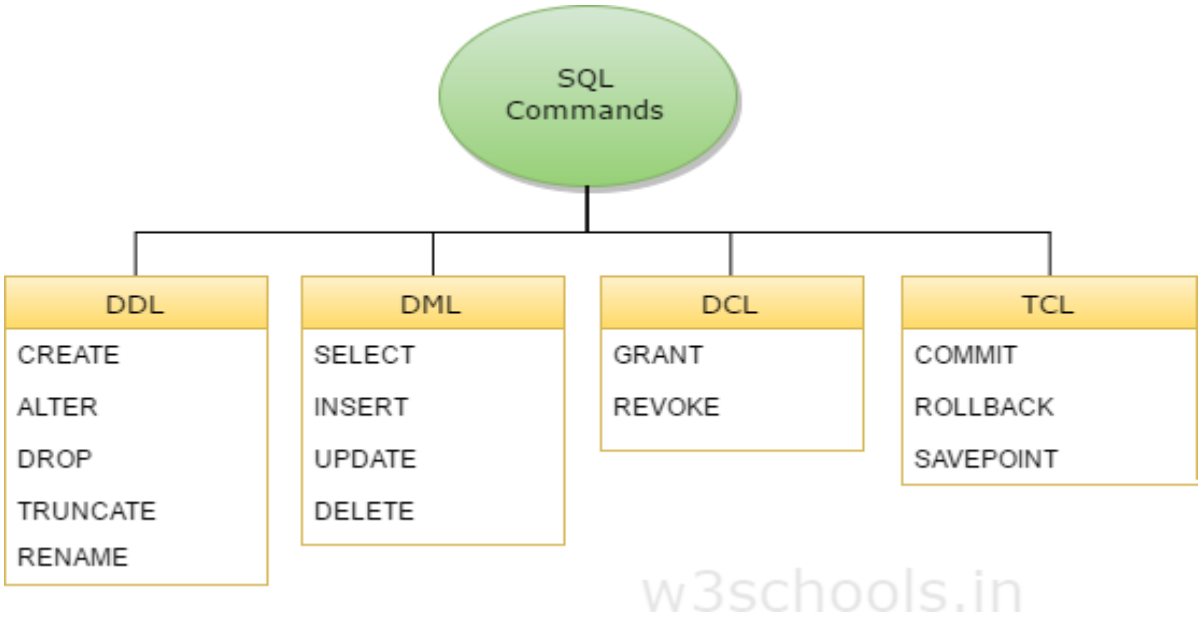
WEEK-6

PRACTISINGDDL&DMLCOMMANDS

DataDefinitionLanguage

The data definition language is used to create an object, alter the structure of an object and also drop already created object. The Data Definition Languages used for table definition can be classified into following:

- Createtablecommand
- Altertablecommand
- Truncatetablecommand
- Droptablecommand



1. **CREATIONOFTABLES:**

SQL-CREATETABLE:
Tableisaprimaryobjectofdatabase,usedtostoredatainformofrowsandcolumns.Itiscreated using

following command:

Syntax:CREATETABLEtablename(column_namedata_typeconstraints,…)

SQL>CREATETABLESAILORS((SIDint(10)PRIMARYKEY,SNAMEVARCHAR(
10),RATINGint(10),AGEint(10));

Table

Created.Descc

Command

TheDESCRIBEcommandisusedtoviewthestructureofatableasfollows

.SQL>DESC SAILORS;

TESTRESULT

Example1:CreateanRESERVEStablewithfields(SID,BID

,DAY)anddisplayusingDESCRIBEcommand.

Example2:CreateaBOATStablewithFields(BID,BNAME,COLOR)anddisplayusingDESCRIBEcommand

2. ALTERTABLE:

ToADDacolumn:

SYNTAX: ALTER TABLE <TABLE NAME>ADD (<NEW
COLUMNNAME><DATATYPE>(<SIZE>),<NEWCOLUMNNAME><
DATATYPE>(<SIZE>).....);

EX:(WriteyourownQuery)

TESTOUTPUT

ToDROPacolumn:SYNTAX:ALTERTABLE<TABLENAME>DROPCOLUMN<COLUMNNA
ME>;.

EX:(WriteyourownQuery)

TESTOUTPUT

ToMODIFYacolumn:SYNTAX:ALTERTABLE<TABLENAME>MODIFY
(<COLUMNNAME>
<NEWDATATYPE>(<NEWSIZE>));

EX:(WriteyourownQuery)

TESTOUTPUT

Example1:

SQL>ALTERTABLESAILORADD(SNONUMBER(10));

TESTOUTPUT

3. **RENAMEATABLE**

Renamecommandisusedtogivenewnamesforexistingtables.

SQL>**RENAME**oldtablename

TOnewtablename;EX:(Writeyourown Query)

TESTOUTPUT

4. **TRUNCATEATABLE**

Truncatecommandisusedtodeleteallrecordsfromatable.

SQL>TRUNCATETABLEtablename;EX:(

WriteyourownQuery)

TESTOUTPUT

5. **DROPATABLE**

Dropcommandisusedtoremoveanexistingtablepermanentlyfromdatabase.

SQL>

DROPTABLEtablename;EX:(Wri

teyourownQuery)

TESTOUTPUT

VIVAQUESTIONS

1. Definedataandinformation.
2. DefineDatabasemanagementsystem.
3. WhatisSQL?
4. Whatisthesyntaxforcreatingatable?
5. ListthecomponentsofSQL.
6. DefineDDL?WhataretheDDLcommands?
7. Listouttheusesofaltercommand.
8. WhatisSyntaxfortruncateatable?
9. Whatistheusedroptablecommand?

DMLCOMMANDS

1. ToRetrieve/DisplayDatafromTables:

a. SelectcommandisusedtoselectvaluesordatafromtableSYNT

AX

SELECT*FROMTABLENAME;

Example:

SQL>SELECT*FROMSAILORS;TE

STOUTPUT:

b. **Theretrievingofspecificcolumnsfromatable**

SQL>**SELECT**columnname1,columnname2,….columnnamenFROMtablename;EX:(Writey ourownQuery)

TESTOUTPUT

c. **Eliminationofduplicatesfromtheselectstatement**

SQL>**SELECTDISTINCT**columnname1,columnname2,….columnnamenFROMtablename;

EX:(WriteyourownQuery)

TESTOUTPUT

d. **Selectingadatasetfromtabledata**

SQL>**SELECT**columnname1,columnname2,….columnnamenFROMtablenameWHEREsearc

hcondition;

EX:(WriteyourownQuery)

TESTOUTPUT

Example1:DisplayDataFromRESERVESTable

Example2:DisplayDataFromBOATSTable

2. INSERTINGDATAINTOTABLE

Insertcommandisusedtoinsertrowsintothetable.SYNT

AX:
INSERTINTOtablename(columnname1,columnname2,….columnnamen)

Example:

SQL>INSERTINTOSAILORSVALUES(22,'DUSTIN',7,45.0);
1rowcreated
SQL>INSERTINTOSAILORSVALUES(29,'BRUTUS',1,33.0);
1rowcreated

INSERTIONofDatacanalsobedonebythefollowingSyntax:

SYNTAX

INSERTINTOtablename(columnname1,columnname2,….columnnamen)VALUES(Value1,Value2,..V
aluen);

Example:
SQL>INSERTINTOSAILORS(SID,SNAME,RATING,AGE)VALUES(31,'LUBBER
',8,55.5);
1rowcreated

Example1:INSERTdataintoRESERVEStable:T

ESTOUTPUT:

Example2:INSERTdataintoBOATStable:TE

STOUTPUT:

UPDATE

ThisSQLcommandisusedtomodifythevaluesinanexistingtable.

SQL>**UPDATE**tablename

SETcolumn1=expression1,column2=expression2,...

WHEREsomecolumn=somevalue;

An expression consists of either a constant (new value), an arithmetic or stringoperation oranSQL query. Note that the new valueto assign to <column>mustmatchingdatatype.

Anupdatestatementusedwithoutawhereclauseresults inchanging respectiveattributesofalltuplesin thespecifiedtable.

Example1:UPDATESAILORSS
SETS.age=S.age+1,S.rating=S.rating-
1WhereS.sid=34546;
TESTOUTPUT

Example2:(WriteyourownQuery)

TESTOUTPUT

DELETE

Inordertodeleterowsfroma

tableweusethiscommandSQL>**DELETE**

FROMtablenameWHEREcondition;

Basedontheconditionspecifiedtherows getsfetchedfromthetableandgetsdeletedintable.HeretheWHEREclauseisoptional.

Example1:DELETES.AGEFROMSAILORSSwhereS.Sname='Smith';

TESTOUTPUT

Example2:DELETEFROMSAILORS;TE

STOUTPUT

VIVAQUESTIONS

1. WhataretheDMLcommands?
2. Howthedataorvaluestobeenteredintoatable?
3. WhatistheuseofDELETEcommand?
4. Howthedataorvaluestobeupdatedonatable?
5. ListouttheusesofSELECTcommand?
6. Howthedataorvaluesareretrievedfromatable?
7. DefineDML?WhataretheDMLcommands?

KEYCONSTRAINTS

Domain Integrity
constraintsEntity Integrity
constraintsReferentialIntegrityc
onstraints

1. PRIMARYKEY&NOTNULL

Example:

```
CREATETABLEsailors(sidinteger,
                   snamevarchar(32)
                   ,ratingintegerNOT
                   NULL,agereal,
                   PRIMARYKEY(sid));
```

Tablecreated.

TestOutput:

Example:PracticewithyourownQuery:

TestOutput

ImposingICusingALTER

Example:AlterTableSailorsMODIFYsname varchar(32)NOTNULL;

TestOutput

Example:PracticewithyourownQuery:

TestOutput

2. **DEFAULT**

```
CREATETABLEsailors(sidinteger,
                   snamevarchar(32)
                   ,rating integer NOT
                   NULL,agerealDEFAULT
                   25,PRIMARYKEY(sid));
```

Example:PracticewithyourownQuery:

TestOutput

3. **UNIQUE**

```
CREATETABLEsailors(sidinteger,
                   snamevarchar(32)UNIQUE
                   ,ratinginteger,age
                    realDEFAULT
                    25,PRIMARY
                    KEY(sid));
```

TestOutput

Example:PracticewithyourownQuery:

TestOutput

4. **FOREIGNKEY**

```
CREATETABLEreserves(sidintegernotnull,bidintegernotnull,daydatetimenotnull,PRIMARY KEY(sid,bid,day),FOREIGN
KEY(sid)REFERENCESsailors(sid));
```

Example:PracticewithyourownQuery:

TestOutput

AddingForeignKeytoanexistingTable

AltertablereservesADDForeignKey(sidREFERENCESSailors(sid));

Example:PracticewithyourownQuery:

TestOutput

VIVAQUESTIONS

1) DifferencebetweenUNIQUEandPRIMARYKEY
2) WhendoyouuseCompositePrimarykey?
3) DifferencebetweenCandidateKey&PrimaryKey
4) WhatisthePrerequisiteforakeytobeusedasaForeignKey
5) WhatisaReferentialIntegrity?
6) GiveTwopracticalexamplesforReferentialIntegrity?

WEEK7

AIMQUERYIN

G

QUERIESUSINGANY,ALL,IN,INTERSECT,UNION

Objectives:

Student willable to learnto operateon multiple resultsetsto return a single resultset.
Student will able to learnto performnested Queries.

Outcomes:

StudentgainstheknowledgetoimplementqueriesusingANY,ALL,IN,INTERSECT,UNION.

6. Create the table using following attributes **TICKET (**TICKET_NO: NUMERIC (9): **PK**, JOURNEY_DATE: DATE, AGE: INT (4), SEX: CHAR (10): MALE/FEMALE, SOURCE: VARCHAR2 (50), DEP_TIME: VARCHAR2 (50)**)**

CreatingtableSy

ntax

Createtable<tablename>(col1datatype,col2datatype,col3datatype)

Query

CREATETABLETICKET**(**TICKET_NONUMBER(9)PRIMARYKEY,JOURNEY_DATEDATE, AGE NUMBER (4), SEX CHAR (10), SOURCEVARCHAR2 (50), DEP_TIME VARCHAR2 (50)**)**

Describingtable

Query

DescTICKET

Output

Table	Column	DataType	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TICKET	TICKET_NO	Number	-	9	0	1	-	-	-
	JOURNEY_DATEAG	Date	7	-	-	-		-	-
	E	Number	-	4	0	-		-	-
	SEXSOU	Char	10	-	-	-		-	-
	RCEDEP_	Varchar2	50	-	-	-		-	-
	TIME	Varchar2	50	-	-	-		-	-

Insertingrecordsinto“TICKET”table

Syntax

Insertinto<table name>values(val1,val2,val3)

Query

insert into TICKET values(1203,'10/FEB/11',19,'MALE','HYDERABAD','10.30 AM');

insert into TICKET values(1213,'10/FEB/11',19,'FEMALE','HYDERABAD','10.30AM');

insertintoTICKETvalues(1201,'13/FEB/11',20,'FEMALE','HYDERABAD','11.30AM');

insertintoTICKETvalues(1202,'14/FEB/11',20,'MALE','TIRUPATHI','11.00AM');insertintoTICKET values(1205,'14/FEB/11',20,'MALE','HYDERABAD','11.00 AM');

DisplaytableSyntax

Select<selectlist>from<tablename>

Query

select*fromTICKET;

Output

TICKET_NO	JOURNEY_DATE	AGE	SEX	SOURCE	DEP_TIME
1203	10-FEB-11	19	MALE	HYDERABAD	10.30 AM

1
1
1
1

7.Create e table using following attributes **PASSENGER_TICKETS (**PPNO: VARCHAR2 (15): **PK**, TICKET_NO: NUMERIC (9)**)**

Creatingtable

Syntax

Createtabe<tablename>(col1datatype,col2datatype,col3datatype)

Query

CREATE TABLE**PASSENGER_TICKETS (**PPNO VARCHAR2 (15)PRIMARYKEY, TICKET_ NUMBER (9)**)**

DescribingtableQuery

DescPASSENGER_TICKETS

Output

213	10-FEB-11	19	FEMALE	HYDERABAD	10.30AM
201	13-FEB-11	20	FEMALE	HYDERABAD	11.30AM
202	14-FEB-11	20	MALE	TIRUPATHI	11.00AM
205	14-FEB-11	20	MALE	HYDERABAD	11.00AM

Table	Column	DataType	Length	Precision	Scale	Primary Key	Nullable	Default	Comme
PASSENGER_TICKETS	PPNO	Varchar2	15	-	-	1	-	-	-
	TICKET_NO	Number	-	9	0	-	✓	-	-

Insertingrecords into

“PASSENGER_TICKETS”tableSyntax

Insertinto<table name>values(val1,val2,val3)

Query

insertintoPASSENGER_TICKETSvalues(1,1203);insertinto

PASSENGER_TICKETS values(2,1213); insert into

PASSENGER_TICKETS values(3,1201); insert into

PASSENGER_TICKETS values(4,1202); insert into

PASSENGER_TICKETSvalues(5,1205);**Displaytable**

Syntax

Select<selectlist>from<tablename>

Query

select *fromTICKET;

Output

PPNO	TICKET_NO
1	1203
2	1213
3	1201
4	1202
5	1205

8. Create the table using following attributes RESERVATION **(**PNR_NO: NUMERIC (9): FK, JOURNEY_DATE: DATE, NO_OF_SEATS: INT (8), ADDRESS: VARCHAR2 (50), CONTACT_NO: NUMERIC (10), STATUS: CHAR (3): YES/NO**)**

➔FOREIGN KEY(PNR_NO)REFERENCES PASSENGER_TICKECTS(PPNO);

Creatingtable

Syntax

Createtable<tablename>(col1datatype,col2datatype,col3datatype)

Query

create table reservation (pnr_no varchar2(15),journey_date date,no_of_seats number(8),address varchar2(50),contact_no number(10),status char(3),foreign key(pnr_no)references passenger_tickets(ppno));

Describingtable

Query

Descreservation

Output

Table	Column	DataType	Length	Precision	Scale	PrimaryKey	Nullable	Default	Comment
RESERVATION	PNR_NO	Varchar2	15	-	-	-	✓	-	-
	JOURNEY_DATE	Date	7	-	-	-	✓	-	-
	NO_OF_SEATS	Number	-	8	0	-	✓	-	-
	ADDRESS	Varchar2	50	-	-	-	✓	-	-
	CONTACT_NO	Number	-	10	0	-	✓	-	-
	STATUS	Char	3	-	-	-	✓		

Insertingrecordsinto“reservation”table

Syntax

Insertinto<tablename>values(val1,val2,val3)

➔FOREIGN KEY(PNR_NO)REFERENCES PASSENGER_TICKECTS(PPNO);

Creatingtable

Syntax

Createtable<tablename>(col1datatype,col2datatype,col3datatype)

Query

create table reservation (pnr_no varchar2(15),journey_date date,no_of_seats number(8),address varchar2(50),contact_no number(10),status char(3),foreign key(pnr_no)references passenger_tickets(ppno));

Describingtable

Query

Descreservation

Output

Table	Column	DataType	Length	Precision	Scale	PrimaryKey	Nullable	Default	Comment
RESERVATION	PNR_NO	Varchar2	15	-	-	-	✓	-	-
	JOURNEY_DATE	Date	7	-	-	-	✓	-	-
	NO_OF_SEATS	Number	-	8	0	-	✓	-	-
	ADDRESS	Varchar2	50	-	-	-	✓	-	-
	CONTACT_NO	Number	-	10	0	-	✓	-	-
	STATUS	Char	3	-	-	-	✓		

Insertingrecordsinto“reservation”table

Syntax

Insertinto<tablename>values(val1,val2,val3)

Query

insert into reservation values(1,'10/feb/11',5,'amberpet','7416944004','yes'); insert into reservation values(1,'11/feb/11',8,'amberpet','7416944004','yes'); insert into reservation values(2,'11/feb/11',8,'b.b nagar','7207204221','yes'); insert into reservation values(2,'14/feb/11',2,'b.b nagar','7207204221','yes'); insert into reservation values(3,'14/feb/11',3,'ecil','00000000','yes');

insert into reservation values(4,'14/feb/11',4,'nagaram','9700135300','yes'); insert into reservation values(5,'16/feb/11',1,'b.b nagar','8143528258','yes'); insert into reservation values(5,'15/feb/11',7,'b.b nagar','8143528258','yes');

Displaytable

Syntax

Select<selectlist>from<table name>

Query

select*fromreservation;

Output

PNR_NO	JOURNEY_DATE	NO_OF_SEATS	ADDRESS	CONTACT_NO	STATUS
1	10-FEB-11	5	amberpet	7416944004	yes
1	11-FEB-11	8	amberpet	7416944004	yes
2	11-FEB-11	8	b.bnagar	7207204221	yes
2	14-FEB-11	2	b.bnagar	7207204221	yes
3	14-FEB-11	3	ecil	0	yes
4	14-FEB-11	4	nagaram	9700135300	yes
5	16-FEB-11	1	b.bnagar	8143528258	yes
5	15-FEB-11	7	b.bnagar	8143528258	yes

Create the table using followingattributes **CANCELLATION (**PNR_NO:NUMERIC (9): **FK**, JOURNEY_DATE: DATE, NO_OF_SEATS: INT (8), ADDRESS: VARCHAR2 (50), CONTACT_NO: NUMERIC (9), STATUS:CHAR (3):YES/NO**)**

➔FOREIGN KEY(PNR_NO)REFERENCES PASSENGER_TICKECTS(PPNO);

Creating

tableSyntax

Createtable<tablename>(col1datatype,col2datatype,col3datatype)

Query

createtablecancellation(pnr_novarchar2(15),journey_datedate,no_of_seatsnumber(8),address varchar2(50),contact_no number(10),status char(3),foreignkey(pnr_no)references passenger_tickets(ppno));

Describingtable

Query

Descreservation

Output

Table	Column	DataType	Length	Precision	Scale	PrimaryKey	Nullable	Default	Commen
CANCELLATION	PNR_NO	Varchar2	15	-	-	-	✓	-	-
							✓		
	JOURNEY_DATE	Date	7	-	-	-	✓	-	-
	NO_OF_SEATS	Number	-	8	0	-	✓	-	-
	ADDRESS	Varchar2	50	-	-	-	✓	-	-
	CONTACT_NO	Number	-	10	0	-	✓	-	-
	STATUS	Char	3	-	-	-		-	-

Inserting records into "CANCELLATION" tableSyntax

Insertinto<tablename>values(val1,val2,val3)

Query

insert into cancellation values(1,'10/feb/11',5,'amberpet','7416944004','yes'); insert into cancellationvalues(1,'11/feb/11',8,'amberpet','7416944004','yes');insertinto cancellation values(2,'11/feb/11',8,'b.b nagar','7207204221','yes'); insert into cancellation values(2,'14/feb/11',2,'b.b nagar','7207204221','yes'); insert into cancellation values(3,'14/feb/11',3,'ecil','00000000','yes');

insert into cancellation values(4,'14/feb/11',4,'nagaram','9700135300','yes'); insert into cancellation values(5,'16/feb/11',1,'b.b nagar','8143528258','yes'); insert into cancellation values(5,'15/feb/11',7,'b.b nagar','8143528258','yes');

Displaytable

Syntax

Select<selectlist>from<table name>

Query

select*fromcancellation

Output

PNR_NO	JOURNEY_DATE	NO_OF_SEATS	ADDRESS	CONTACT_NO	STATUS
1	10-FEB-11	5	amberpet	7416944004	yes
1	11-FEB-11	8	amberpet	7416944004	yes
2	11-FEB-11	8	b.bnagar	7207204221	yes
2	14-FEB-11	2	b.bnagar	7207204221	yes
3	14-FEB-11	3	ecil	0	yes

4	14-FEB-11	4	nagaram	9700135300	yes
5	16-FEB-11	1	b.bnagar	8143528258	yes
5	15-FEB-11	7	b.bnagar	8143528258	yes

26. Writeatriggeronpassengertodisplaymessages‘1Recordisinserted’,‘1recordis deleted’, ‘1 record is updated’ when insertion, deletion and updation are done on passenger respectively.

27. DisplayuniquePNR_NOofallpassengers.

Query

```
selectdistinct(pnr_no)fromreservation;
```

Output

PNR_NO
1
3
5
2
4

28. Displayallthenamesofmalepassengers.

Query

```
selectnamefrompassenger wheresex='MALE'
```

Output

NAME
TIRUMALAY
NAGARAJU
AVS.RAVI

29. Displaytheticketnumbers andnamesofallthepassengers.

Query

```
selectp.name,t.ticket_nofrompassengerp,passenger_ticketstwhere
                                    t.ppno=p.ppno
```

Output

NAME	TICKET_NO
TIRUMALAY	1203
SUPRIYA	1213
AMULYA	1201
NAGARAJU	1202
AVS.RAVI	1205

30. Findtheticketnumbers ofthe passengers whosenamestartwith‘t’andendswith‘y’.

Query

```
selectt.ticket_nofrompassenger p,passenger_ticketstwherep.namelike'T%Y'and
t.ppno=p.ppno;
```

Output

TICKET_NO
1203

31. Findthenamesofpassengerswhoseageisbetween15and20.

Query

```
selectnamefrompassengerwhereagebetween15and20
```

Output

NAME
TIRUMALAY
SUPRIYA
AMULYA
NAGARAJU
AVS.RAVI

32. Displayallthepassengersnamesbeginningwith‘A’.

Query

selectnamefrompassengerwherenamelike'A%';

Output

NAME
AMULY A
AVS.RAV I

33. Displaythesortedlistofpassengersnames.

Query

selectnamefrompassengerorderbyname;

Output

NAME
AMULYA
AVS.RAVI
NAGARAJU
SUPRIYA
TIRUMALAY

34. Write aquerytodisplaytheinformationpresentinthePASSENGERandCANCELLATION tables. (Use UNION Operator).

Query

select*frompassengerp,cancellationcwherep.ppno=c.pnr_nounion select

*frompassenger p1,cancellationc1wherep1.ppno=c1.pnr_no

Output

PP NO	NAME	AGE	SEX	ADDRESS	PNR_NO	JOURNEY_DATE	NO_OF_SEATS	ADDRESS	CONTACT_NO	STATUS
1	TIRUMALAY	19	MALE	AMBERPET	1	10-FEB-11	5	amberpet	7416944004	yes
1	TIRUMALAY	19	MALE	AMBERPET	1	11-FEB-11	8	amberpet	7416944004	yes
2	SUPRIYA	20	FEMALE	B.B NAGAR	2	11-FEB-11	8	b.b nagar	7207204221	yes
2	SUPRIYA	20	FEMALE	B.B NAGAR	2	14-FEB-11	2	b.b nagar	7207204221	yes
3	AMULYA	20	FEMALE	ECIL	3	14-FEB-11	3	ecil	0	yes
	NAGARAJU	20	MALE	NAGARAM	4	14-FEB-11	4	nagaram	9700135300	yes
5	AVS.RAVI	20	MALE	B.B NAGAR	5	15-FEB-11	7	b.b nagar	8143528258	yes
5	AVS.RAVI	20	MALE	B.B NAGAR	5	16-FEB-11	1	b.b nagar	8143528258	yes

35. Displaythenumberofticketsbookedforeach PNR_NO using GROUP BY clause. (Use GROUP BY on PNR_NO).

Query

```
selectpnr_no,sum(no_of_seats)fromreservationgroupbypnr_no;
```

Output

PNR_NO	SUM(NO_OF_SEATS)
1	13
3	3
5	8
2	10
4	4

36. FindthedistinctPNRnumbersthatarepresent.

Query

```
Selectdistict(pnr_no)fromreservation;
```

Output

PNR_NO
1
3
5
2
4

37. Findthenumberoftickets bookedbyapassengerwhere the numberofseatsis greater than5. (Use GROUP BY, WHERE and HAVING clauses).

Query

```
selectpnr_no,sum(no_of_seats)fromreservationgroupbypnr_nohavingsum(no_of_seats)>5
```

Output

PNR_NO	SUM(NO_OF_SEATS)
1	13
5	8
2	10

38. Findthetotalnumberofcancelledseats.

Query

```
selectsum(no_of_seats)fromcancellation;
```

Output

SUM(NO_OF_SEATS)
38

VIVAQUESTIONS

1. Whatisthesyntaxforcreatecommand?
2. Whatisthedifferencebetweenprimarykeyandforeignkey?
3. Whatisthecommandtodisplaydatafroma table?
4. Whatarethetypesofclauseusedinmysql?

WEEK8&9:

QueryingUsingAggregatefunctions(COUNT,SUM,AVERAGEusingGROUPBY andHAVING)Queries usingAggregatefunctions (COUNT, SUM,AVG,MAXand MIN),GROUP BY, HAVING and Creationand dropping of Views.

Objectives:

Studentwillabletolearntoperformmathematicaloperationsthatreturnasinglevalue,calculatedfrom values in a column.

Outcomes:

Studentgainstheknowledgetoperformaggregateoperationsonthedatabaseappropriately.

Aggregate operators: In addition to simply retrieving data, we often want to perform some computation or summarization.SQLallowstheuseofarithmeticexpressions.Wenowconsiderapowerfulclassofconstructs for computing aggregate values such as MIN and SUM.

1. Count:COUNT followingbyacolumn namereturnsthecountoftuple inthatcolumn. If DISTINCT keywordis used then it will return only the count of unique tuple in the column. Otherwise, it will return count of all the tuples (including duplicates) count(*) indicates allthe tuples of the column.

Syntax:COUNT(Columnname)

Example:SELECTCOUNT(Sal)FROM emp;

2. SUM: SUM followed by a column name returns the sum of all the values in that column.

Syntax:SUM(Columnname)

Example:SELECTSUM(Sal)Fromemp;

3. AVG: AVG followed by a column name returns the average value of that column values.

Syntax:AVG(n1,n2..)

Example:SelectAVG(10,15,30)FROMDUAL;

4. **MAX:**MAXfollowedbyacolumnnamereturnsthemaximumvalueofthatcolumn.

Syntax:MAX(Columnname)

Example:SELECTMAX(Sal)FROMemp;SQL>select

deptno,max(sal) from emp group bydeptno;

DEPTNO	MAX(SAL)
10	5000
20	3000
30	2850

SQL>selectdeptno,max(sal)fromempgroupbydeptnohavingmax(sal)<3000;DEPTNO MAX(SAL)

30	2850

5. **MIN:**MIN followedbycolumnnamereturnstheminimumvalueofthatcolumn.

Syntax:MIN(Columnname)

Example:SELECTMIN(Sal)FROMemp;

SQL>selectdeptno,min(sal)fromempgroupbydeptnohavingmin(sal)>1000;DEPTNO MIN(SAL)

10	1300

VIEW: InSQL,aviewisavirtualtablebased ontheresult-setofanSQLstatement.

Aview contains rows and columns, just likearealtable.The fieldsina view are fieldsfrom one or more real tablesin the database.

You canadd SQL functions, WHERE, and JOIN statements toa view and present thedata asifthe datawere comingfrom one single table.

Aviewis a virtualtable,whichconsistsofasetofcolumnsfrom oneormore tables. It is similarto atablebut itdoest not store inthe database. Viewis aquerystored as an object.

Syntax: CREATEVIEWview_nameASSELECTsetoffieldsFROM relation_name WHERE (Condition)

1. **Example:**

SQL>CREATEVIEWemployeeASSELECTempno,ename,jobFROMEMPWHEREjob= ‘clerk’;

Viewcreated.

SQL>SELECT*FROMEMPLOYEE;EMPNO

	ENAME	JOB
7369	SMITH	CLERK
7876	ADAMS	CLERK
7900	JAMES	CLERK
7934	MILLER	CLERK

2. Example:

```
CREATEVIEW[CurrentProductList]ASSELECT
ProductID,ProductName
FROMProducts
WHEREDiscontinued=No
```

DROPVIEW:Thisqueryisusedtodeleteaview,whichhasbeenalreadycreated.

Syntax: DROPVIEWView_name;

Example: SQL>DROPVIEWEMPLOYEE;

Viewdropped

Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr andinstr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

1. **Conversionfunctions:To_char:**TO_CHAR (number)converts n toavalue of VARCHAR2 data type, using the optional number format fmt. The value n can be oftype NUMBER, BINARY_FLOAT, or BINARY_DOUBLE.

```
SQL>selectto_char(65,'RN')fromdual;
```

To_number:TO_NUMBERconvertsexprtoavalueofNUMBERdatatype.

SQL> Select to_number('1234.64') from Dual;1234.64

To_date:TO_DATEconvertscharofCHAR,VARCHAR2,NCHAR,or NVARCHAR2 data type toa value ofDATE data type.

SQL>SELECTTO_DATE('January15,1989,11:00A.M.')FROMDUAL;TO_DATE('15

-JAN-89

---

2. Stringfunctions:

Concat:CONCATreturnschar1concatenatedwithchar2.Bothchar1andchar2canbeanyofthedatatypes SQL>SELECT

CONCAT(‘ORACLE’,’CORPORATION’)FROM DUAL;

ORACLECORPORATION

Lpad:LPADreturnsexpr1,left-paddedtolengthncharacterswiththesequenceofcharactersinexpr2.

SQL>SELECT LPAD(‘ORACLE’,15,’*’)FROM DUAL;**********ORACLE

Rpad:RPADreturnsexpr1,right-paddedtolengthncharacterswithexpr2,replicatedasmanytimesas necessary.

SQL>SELECTRPAD(‘ORACLE’,15,’*’)FROMDUAL;

ORACLE*********

Ltrim: Returnsacharacterexpressionafterremovingleadingblanks.

SQL>SELECTLTRIM(‘SSMITHSS’,’S’)FROMDUAL;MITHSS

Rtrim: Returnsacharacterstringaftertruncatingalltrailingblanks

SQL>SELECTRTRIM(‘SSMITHSS’,’S’)FROMDUAL;SSMITH

Lower: Returnsacharacterexpressionafterconvertinguppercasecharacterdatatolowercase.

SQL>SELECT LOWER(‘DBMS’)FROM DUAL;

dbms

Upper: Returnsacharacterexpressionwithlowercasecharacterdataconvertedtouppercase

SQL>SELECT UPPER(‘dbms’)FROM DUAL;DBMS

Length: Returnsthenumberofcharacters,ratherthanthe numberofbytes,ofthegivenstring expression, excluding trailing blanks.

SQL>SELECTLENGTH(‘DATABASE’)FROMDUAL;8

Substr: Returnspartofacharacter,binary,text,orimageexpression.

SQL>SELECTSUBSTR(‘ABCDEFGHIJ’3,4)FROMDUAL;CDEF

Instr: TheINSTRfunctionssearchstringforsubstring.Thefunctionreturnsanintegerindicatingthe position ofthe character instring that is the first character ofthis occurrence.

SQL>SELECTINSTR('CORPORATEFLOOR','OR',3,2)FROMDUAL;14

3. Datefunctions:

Sysdate:SQL>SELECTSYSDATEFROMDUAL;29-DEC-08

next_day:

SQL>SELECT NEXT_DAY(SYSDATE,'WED')FROMDUAL;05-JAN-09

add_months:

SQL>SELECTADD_MONTHS(SYSDATE,2)FROMDUAL;28-FEB-09

last_day:

SQL>SELECTLAST_DAY(SYSDATE)FROMDUAL;31-DEC-08

months_between:

SQL>SELECTMONTHS_BETWEEN(SYSDATE,HIREDATE)FROMEMP;4

Least:

SQL>SELECTLEAST('10-JAN-07','12-OCT-07')FROMDUAL;10-JAN-07

Greatest:

SQL>SELECTGREATEST('10-JAN-07','12-OCT-07')FROMDUAL;10-JAN-07

Trunc:

SQL>SELECTTRUNC(SYSDATE,'DAY')FROMDUAL;28-DEC-08

Round:

SQL>SELECT ROUND(SYSDATE,'DAY')FROM

DUAL;28-DEC-08

to_char:

SQL>selectto_char(sysdate,"dd\mm\yy")from

dual;24-mar-05.

to_date:

SQL>selectto_date(sysdate,"dd\mm\yy")from

dual;24-mar-o5.

VIVAQUESTIONS

1. Whatareaggregatefunctions?
2. WhatisthedifferencebetweenLPADandRPAD?
3. DefineView?
4. Whatisthedifferencebetweengroupbyandorderbyclause?

Week-10TRIGGERS

In MySQL, a trigger is a set of SQL statements that is invoked automatically when a change ismadetothedataontheassociatedtable.Atriggercanbedefinedtobeinvokedeitherbeforeorafterthedatai schangedbyINSERT, UPDATEorDELETEstatement.

- BEFOREINSERT –activatedbeforedataisinsertedintothetable.
- AFTERINSERT – activatedafterdataisinsertedintothetable.
- BEFOREUPDATE –activatedbeforedatainthetableisupdated.
- AFTERUPDAT –activatedafterdatainthetableisupdated.
- BEFOREDELETE –activatedbeforedataisremovedfromthetable.
- AFTERDELETE –activatedafterdataisremovedfromthetable.

Adatabasetriggerisproceduralcodethatisautomaticallyexecutedinresponsetocertainevents on a particular table orview in a database. The trigger is mostly used for maintainingtheintegrity oftheinformationon thedatabase.

Theeventsthatfireatriggerincludethefollowing:
1)DMLstatementsthatmodifydatainatable(INSERT,UPDATE,orDELETE)2)DDLstateme nts.
3) Systemeventssuchasstartup,shutdown,anderrormessages.
4) Usereventssuchaslogonandlogoff.Note:OracleFormscandefine,store,andruntriggersofadiffe rentsort.

ToViewlistoftriggers;

Showtriggers;

ToremoveatriggerforDatabase

droptriggertrigger_name;ex:

droptriggerins_sal;

TypesofTriggers:-
1. **RowTriggers**:-Arowtriggerisfiredeachtimethetableisaffectedbythetriggeringstatement. For example, if an UPDATE statement updates multiple rows of a table, a rowtrigger is fired once for each row affected by the UPDATE statement. If a triggering statementaffectsnorows,arowtriggerisnotexecutedatall.

Rowtriggersareusefulifthecodeinthetriggeractiondependsondata providedbythetriggering statement or rows that are affected. For example, Figure 15- 3 illustrates a rowtriggerthatusesthevaluesofeachrowaffectedbythetriggeringstatement.

2. **Statement Triggers**:A statement trigger is fired once on behalf of the triggering statement,regardless ofthe numberofrowsinthetablethatthetriggering statementaffects (even if norows are affected). For example, if a DELETE statement deletes several rows from a table, astatement-level DELETE triggeris fired only once, regardless of how many rows are deletedfromthetable.

Statement triggers are useful if the code in the trigger action does not depend on thedata provided by the triggering statement or the rows affected. For example, if a trigger makesacomplexsecuritycheckonthecurrenttimeoruser,orifatriggergeneratesasingle auditrecordbasedonthetypeoftriggeringstatement,astatementtriggerisused.

When defining a trigger, specify the trigger timing. That is, specify whether the triggeractionistobeexecutedbeforeorafterthetriggeringstatement.BEFOREandAFTER apply tobothstatementandrowtriggers

Sample:

```
CREATETRIGGERtrigger_name          trigger_time
                                    trigger_eventONtable_name
 FOREACHROW
 BEGIN
 ...END
 ;

trigger_time=before/aftertrigger_event=insert/delete/
 update
```

Example:

```
CREATE TRIGGER sal_sum after insert ON
    empFOREACHROWSET@sal=@sal+NEW.sal;
```

Firingatrigger:

Question:Findthesumofsalariesofallemployees1)Firs

tcreateatable**emp**withfollowingcolumns

Field	**Type**

empid	int(11)
ename	varchar(50)
sal	int(11)

WriteaQuery:

TESTOUTPUT

2) createvariable/parameter**sal**asbelowatmysqlpromptmysql

>**set@sal=0;**

3) nowcreatetriggeronemp

CREATE TRIGGER sal_sum after insert ON
empFOREACHROWSET@sal=@sal+NEW.sal;
TESTOUTPUT

4) insertthevaluesintotableemp;
mysql>insert into emp
values(1001,'suhaas',10000);mysql> insert into emp
values(1002,'Dilraj',15000);mysql>insertintoempvalues(1
003,'Riyanshi',25000);

Note:triggerisfiredonafterinsert5)
checkvaluesinthetableemp;

mysql>select*fromemp;
TESTOUTPUT

6) checkingvalueintheparametersalm

ysql>select@salasTotalSalary;
TESTOUTPUT

Note:whenevertherеisinsertoperationthatvalueinthe**sal**variableincreases

VIVAQUESTIONS

1. Definedatabasetriggers.
2. Listouttheusesofdatabasetriggers.
3. Whataretheparsoftriggersandituses?
4. Listoutthetypesoftrigger.
5. Whatistheuseofrowtrigger?
6. Whatistheuseofstatementtrigger?
7. Whatdoyoumeantbytriggertime?
8. Comparebeforetriggerandaftertrigger.
9. WhatisthesyntaxforDROPatrigger?
10. Listoutthesomesituationstoapplybeforeandaftertriggers.

WEEK-11,12

PROCEDURES

Procedures:Creationofstoredprocedures,Executionofprocedureandmodificationof procedures.

Objectives:

Studentwillabletolearnthefeatureslikereusability,maintainabilityandmodularity. Student willable to learnto develop proceduresand function for variousoperation.

Outcomes:

Studentgainstheknowledgetoimplementproceduresandfunctionforvariousoperations.

```
CREATEPROCEDUREmyProc()BEGIN
SELECTCOUNT(Tickets)FROMTicketWHEREage>=40;End; Procedures
created
SQL>CREATEPROCEDUREmyproc(in_customer_id INT)BEGIN
 DECLARE v_id INT;
 DECLARE v_nameVARCHAR(30);
 DECLAREc1CURSOR FORSELECTstdId,stdFirstnameFROMstudents
 WHERE stdId=in_customer_id;
 OPENc1;
 FETCHc1intov_id,v_name;Close c1;
 End;
 /
 PL/SQLproceduresuccessfullycompleted.
```

PL/SQL

PL/SQLprogramsarewrittenaslinesoftextusingaspecificsetofcharacters:

- Upper-andlower-caselettersA..Zanda..z
- Numerals0..9
- Symbols()+-*/<>=!~^;:.'@%,"#$&_|{}?[]
- Tabs,spaces,andcarriagereturns

PL/SQLkeywordsarenotcase-sensitive,solower-caselettersareequivalenttocorrespondingupper-caselettersexceptwithinstringandcharacterliterals.

AlineofPL/SQLtextcontainsgroupsofcharactersknownaslexicalunits:

- Delimiters(simpleandcompoundsymbols)
- Identifiers,whichincludereservedwords
- Literals
- Comments

To improve readability, you can separate lexical units byspaces. In fact, you must separateadjacentidentifiersbyaspaceorpunctuation.ThefollowinglineisnotallowedbecausethereservedwordsENDandIFarejoined:

IFx>yTHENhigh:=x;ENDIF;--notallowed,mustbeENDIF

You cannot embed spaces inside lexicalunits except for string literals and comments. Forexample,thefollowinglineisnotallowedbecausethecompoundsymbolforassignment(:=)issplit:

count:=count+1;--notallowed,mustbe:=

Toshowstructure,youcansplitlinesusingcarriagereturns,andindentlinesusingspacesortabs.Thisformattingmakesthefirst IF statementmorereadable.

IFx>yTHENmax:=x;ELSEmax:=y;ENDIF;

1) WRITEA PROGRAM TOPRINTHELLO WORLDBEGIN DBMS_OUTPUT.PUT_LINE('HELLOWORLD');END;

TESTOUTPUT

2) WRITEAPROGRAMTOPRINTEVENNUMBERSFROM1TO100DECLA
RE
NNUMBER(3):=0;
BEGIN
WHILE
N<=100LOOP
N
:=N+2;DBMS_OUTPUT.PUT_LINE
(N);ENDLOOP;
END;

TESTOUTPUT

3) WRITEAPROGRAMTOACCEPTANUMBERANDFINDSUMOFTHEDIGITSDECLAR
E
NNUMBER(5):=&N;S
NUMBER:=0;
RNUMBER(2):=0;BEGIN
WHILE N
!=0LOOPR:=MOD
(N,10);S:=S+R;
Page 1 of
7N:=TRUNC(N/10
);ENDLOOP;
DBMS_OUTPUT.PUT_LINE('SUMOFDIGITSOFGIVENNUMBERIS'||S);END;

TESTOUTPUT

4) Writeaprogramtoacceptanumberandprintitinreverseorder
```
DECLA
RE
N
NUMBER(5):=&N;RE
VNUMBER(5):=0;R
NUMBER(5):=0;BEGI
N
WHILE N
!=0LOOPR:=MO
D(N,10);REV:=RE
V*10+R;N:=TRUN
C(N/10);ENDLOO
P;
DBMS_OUTPUT.PUT_LINE('THEREVERSEOFAGIVENNUMBERIS'||REV);END;
```

TESTOUTPUT

5) WriteaprogramacceptthevalueofA,B&Cdisplay
```
whichisgreaterDECLARE
ANUMBER(4,2):=&A;
B
NUMBER(4,2):=&B;C
NUMBER(4,2):=&C;B
EGIN
IF(A>BANDA>C)THENDBMS_OUTPUT.PUT_LINE('
AISGREATER'||''||A);ELSIFB>CTHEN
DBMS_OUTPUT.PUT_LINE('BISGREATE'||''||B);ELSE
DBMS_OUTPUT.PUT_LINE('CISGREATER'||''||C);END
IF;
END;
```

VIVAQUESTIONS

1. WhatisPL/SQL?
2. WhatisthebasicstructureofPL/SQL?
3. HowisaprocessofPL/SQLcompiled?
4. MentionwhatPL/SQLpackageconsistsof?
5. WhatarethebenefitsofPL/SQLpackages?
6. WhatisthedifferencebetweenFUNCTION,PROCEDUREANDPACKAGEinPL/SQL
7. ShowhowfunctionsandproceduresarecalledinaPL/SQLblock?
8. WhatisStoredProcedure?
9. WhatisdifferencebetweenFunctionandStoredProcedure?

WEEK13

DCL(DATACONTROLLANGUAGE): DataControlLanguagestatementsareusedtocreate roles, permissions, and referential integrity as wellit is used tocontrol accesstodatabasebysecuringit.DCLCommandsareGrantandRevoke

GRANT -givesuser'saccessprivilegestodatabase

REVOKE -

withdrawaccessprivilegesgivenwiththeGR

ANTcommand

CheckingofUserPrivileges,Grantsetcm

ysql>createusermrcet_cse;
QueryOK,0rowsaffected(0.30sec)
*ToCheckwhereisthecreateduseri.elocationinourdatabase
mysql>selectuser();
TESTOUTPUT

***Tocheckwhatarethegrantsthatthelocationishaving/mysq
l>**showgrants**;**

TESTOUTPUT

***ToCheckwhataretheGRANTShavingforcreatedusermys
ql>**showgrantsformrcet_cse;

TESTOUTPUT

mysql>showtables;

TESTOUTPUT

***ToFlush(RE-FRESH)theprivilegesmysql>**flushprivile
ges;
QueryOK,0rowsaffected(0.08sec)

*__Explanation:__*Tocheckwhereistheuseri.eincaseifwecreateduser(Ex:mrcet_cse)itwillbedisplayed as"%". Rootuserisbydefaultsoitwill beavailablein"Localhost"*

mysql>selecthost,userfrommysql.user;
TESTOUTPUT

VIVAQUESTIONS

1. WhatareDCLcommands?
2. ListouttheusesofvariousDCLcommands?
3. WhatarethedifferenttypesofCommandsinSQL.
4. WhatisthedifferencebetweenTCL&DCLcommands.
5. WhohastheprivilegetoaccesstheDCLcommands.

CASESTUDY1

Emp(eid:integer,ename:string,age:integer,salary:real)Works(

eid:integer,did:integer,pcttime:integer)

Dept(did:integer,dname:string,budget:real,managerid:integer)

1. Giveanexampleofaforeignkeyconstraintthatinvolvestheprelation.Whataretheoptions forenforcingthisconstraintwhenauserattemptstodeleteaDepttuple?
2. Write theSQLstatementsrequired tocreate the above relations,including appropriateversionsofallprimaryandforeignkeyintegrityconstraints.
3. DefinetheDeptrelationinSQLsothateverydepartmentisguaranteedtohaveamanager.
4. WriteanSQLstatementtoadd`JohnDoe'asanemployeewitheid=101,age=32and salary=15;000.
5. WriteanSQLstatementtogiveeveryemployeea10%raise.
6. WriteanSQLstatementtodeletethe`Toy'department.Giventhereferentialintegrityconstraints you choseforthis schema, explain whathappenswhen thisstatementisexecuted.

CASESTUDY2

Suppliers(sid:integer,sname:string,address:string)Pa

rts(pid: integer, pname: string, color:

string)Catalog(sid:integer,pid:integer,cost:

real)RelationalAlgebraandCalculus117

The key fields are underlined, and the domain of each field is listed after the field

name.ThussidisthekeyforSuppliers,pidisthekeyforParts,andsidandpidtogetherformthekey

for Catalog.The Catalog relation lists the pricescharged for partsbySuppliers.

Writethefollowingqueriesinrelationalalgebra,tuplerelationalcalculus,anddomainrelationalc

alculus:

1. Findthenamesofsupplierswhosupplysomeredpart.
2. Findthesidsofsupplierswhosupplysomeredorgreenpart.
3. Findthenumberofpartswhosenamehas5letters.
4. Findthesidsofsupplierswhosupplyatleast3parts.
5. Findthesidsofsupplierswhosupplyeverypart.

6. Findthesidsofsuppliers whosupplyeveryredpart.
7. Findthesidsofsuppliers whosupplyeveryredorgreenpart.
8. Findthesidsofsuppliers whosupplyeveryredpartorsupplyeverygreenpart.
9. Findpairsofsidssuchthatthesupplierwiththefirstsid chargesmoreforsomepartthanthesupplierwiththesecondsid.
10. Findthepidsofpartsthataresuppliedbyatleasttwodifferentsuppliers.
11. Findthesidofsupplierwhosupplycostliestpart.
12. Findthepidsofpartssuppliedbyeverysupplieratlessthan$200.(Ifanysupplie reitherdoesnotsuppl ythepartorchargesmorethan$200forit,thepartisnotselected.)

CASESTUDY3

Considerthefollowingrelationscontainingairlineflightinformation:Flig

hts(flno:integer,from:string,to:string,

distance:integer,departs:time,arrives:time)

Aircraft(aid:integer,aname:string,cruisingrange:integer)Certified(eid:int

eger,aid:integer)

Employees(eid:integer,ename:string,salary:integer)

NotethattheEmployeesrelationdescribespilotsandotherkindsofemployeesaswell;everypilot is certified for some aircraft (otherwise, he or she would not qualify as a pilot), and onlypilotsarecertified tofly.

Write thefollowingqueriesinrelationalalgebra,tuple relationalcalculus, anddomainrelational

calculus.

1. Findtheeidsofpilotscertifiedforsome Boeingaircraft.
2. Findthenamesofpilotscertifiedforsome Boeingaircraft.
3. Findtheaidsofallaircraftthatcanbeusedonnon-stopflightsfromBonntoMadras.
4. Identifytheflightsthatcanbepilotedbyeverypilotwhosesalaryismorethan$100,000.

(Hint:Thepilotmustbecertifiedforatleastoneplanewithasufficientlylargecruisingrange.)

5. Findthenamesofpilotswhocanoperatesomeplanewitharangegreaterthan3,000milesbutare notcertifiedonanyBoeingaircraft.

6. Findtheeidsofemployeeswhomakethehighestsalary.

7. Findtheeidsofemployeeswhomakethesecondhighestsalary.

8. Findtheeidsofpilotswhoarecertifiedforthelargestnumberofaircraft.

9. Findtheeidsofemployeeswhoarecertifiedforexactlythreeaircraft.

10. Findthetotalamountpaidtoemployeesassalaries.

11. Istherea sequenceofflightsfromMadisontoTimbuktu?Eachflight inthesequenceisrequiredtodepartfromthecitythatisthedestinationofthepreviousflight;thefirst flightmustleaveMadison,thelastflightmustreachTimbuktu,andthereis norestrictiononthenumberofintermediateflights.Yourquerymustdeterminewhetherasequence