Day 04 - 29 Sep 2020

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Complete Default/Unique/Foreign Key/primary key constraint deletion

Primary Keys/Foreign Keys/Candidate Keys/Alternate Keys/Composite Key[Completed]

Basic Queries

Operators -

Arithmetic/Logical/Comparison

in any between and or operators

like operator

Functions qd0102

Literals

Keys:Primary Keys/Foreign Keys/Candidate Keys/Alternate Keys/Composite Key[Partially Completed]

Order by asc desc

Group by / Having

JOINS [q28Feb20/27Feb]

INDEXING [(sampledemo)]

VIEWS [(Q5Mar20)]

SUBQUERIES [(Q28Feb))]

GRANT SELECT,INSERT,UPDATE,DELETE ON emp TO 'root'@'localhost';

grant select on customer to 'sa'@'localhost'

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Database Integrity: feature by which the system allows me to store only valid data

for eg. the DBMS facilitating through Primary Key , which allows me to identify every record uniquely

Entity Integrity/Domain Integrity/Referential Integrity / User Defined Integrity

Entity Integrity: Table Level Integrity, to idenitfy every record uniquely

Primary Key

Employee Salary

EmpId(Pk) EmpId(Fk)

E1 E1200

E1000

---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Primary Key

EmployeeId [] EmpName EmpAddress EmpPhone [] EmpEMail[]

Candidate Keys: Those fields which are eligible to become Primary Keys are called Candidate Keys

Alternate Keys : Those fields which were capable of becoming primary, but not chosen as primary key

Composite Key : combination of fields behaving as a Primary key

ICode ProjectCode EmployeeCode ProjectDescription ProjectDuration

1 P1 E1 ShoppingSystem 2 Years

2 P2 E2 ReservationSystem 1 Yr

3 P3 E1 HR Mgmt System 3 Yr

4 P1 E3 XYZ 4 Yr

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Authors

AuthorId AuthorName AuthorCity

A1 Shakespeare Lancashire

A2 Shelly Wankashire

A3

A100

Publishers

PublisherId

P1

P2

P3

-

P50

Titles

TitleId(Pk) PublisherID(Fk) Title Category AuthorID[Fkey]

T1 P1 A Comedy A1

T2 P2 B Tragedy A1

T3 P3 C Comedy A120

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Constraints

Primary Key

Foreign Key

Check

Unique

Default Object

-----------------------------------------------------------

Create table Department

(

DeptId varchar(20) primary key,

DeptName varchar(40),

DeptHead varchar(30),

DeptStrength int

)

Create table Employee1

(

EmployeeId varchar(20) primary key,

EmployeeName varchar(30),

EmployeeAddress varchar(40),

EmployeeSalary int,

DepartmentId varchar(30)

)

alter table Employee1

add constraint cfkey1 foreign key (DepartmentId ) references Department(DeptId)

DeptId DeptName DeptHead DeptStrength

D1 Engg RS 100

D2 Maint KR 150

Employee1

EmployeeId Name Address Salary DepartmentId

E1 JK RTngr 1000 D1

E2 RT JNgr 2000 D2

For a Table to become a Primary Table with respect to a field , that table shoudl have that field as a Primary Key in it

For a Table to become a Sub Table with respect to a field , that table should have that field as a Foreign key in it.

----------------------------------------------------------

Create table Salary

(

SalaryId varchar(20) primary key,

EmployeeId varchar(20),

Basic int,

HRA int,

CCA int,

GrossSalary int,

NettSalary int,

constraint cfkey2 foreign key (EmployeeId) references Employee1(EmployeeId)

)

----------------------------------------------------------------------------------------T Break 11.00 to 11.20-----------------------------------------------

Mathematical/Arithmetic + - / %

Logical and or not

< > <= >=

<>

between and and

in / not in

Functions

Aggregate Functions

String Functions

Mathematical

Date

=================================================

MovieId MovieCategory MovieName Collections

M1 Comedy HAHK 450000

M2 Tragedy HAHK 500000

M3 Drama HAHK 650000

M4 Comedy HAHK 450000

M5 Action HAHK 350000

M6 Drama HAHK 850000

M7 Comedy HAHK 950000

M8 Action HAHK 450000

Action

Comedy

Drama

=================================================Break 4.00 pm to 4.15 PM======================================

values('P001','Television','Electronics',15000),

('P002','Refrigerator','Electronics',25000),

('P003','WashingMachine','Electronics',20000),

('P004','MixerGrinder','HomeAppliance',5000),

('P005','Microwave','HomeAppliance',7000),

('P006','Cooker','Utensil',1000),

('P007','BookShelf','Furniture',12000)

values('O001','2020-1-15','C001',100000,'P001',10),

('O002','2018-2-15','C001',100000,'P001',20),

('O003','2017-1-15','C001',120000,'P002',15),

('O004','2016-1-15','C002',10000,'P002',5),

('O005','2019-3-15','C002',150000,'P003',25),

('O006','2020-4-15','C003',200000,'P003',30),

('O007','2020-5-15','C003',100000,'P003',15),

('O008','2020-6-15','C001',100000,'P004',10),

('O009','2020-2-15','C004',120000,'P005',12),

('O010','2020-3-15','C001',130000,'P005',13)

===================================================================JOINS==========================================================

Joins enable us to fetch data from multiple tables, in a consistent way

Inner Join[ Equi / Natural]

Outer Join[ Left Outer / Right Outer Join]

Cross Join

Self Join

Customer ----- Orders------Products

CId(Pk) CId(FK)

OrdId(PK

PId(FK) PId(PK)

Customers Orders Products

select CustomerName,CustomerAddress,OrderDate,OrderValue,Productname,ProductCategory

from Customers join Orders

on Customer.CustomerId = Orders.CustomerID

A B

------------------------------------------------------------------------------------------------

AuthorId AuthorName AuthorCity AuthorId AuthorName AuthorCity

A001 Kuvempu Mysore A001 Kuvempu Mysore

A002 Akilan Chennai A002 Akilan Chennai

A003 Shakespeare London A003 Shakespeare London

A004 Shelly London A004 Shelly London

A005 Hari Allahabad vs A005 Hari Allahabad

A006 AnaKru Mysore A006 AnaKru Mysore

A007 Nisar Bangalore A007 Nisar Bangalore

A008 Keats London A008 Keats London

A008 Keats London A008 Keats London

select a.AuthorName,a.AuthorCity, b.AuthorName,b.AuthorCity

from Authors a join Authors b

on

a.AuthorCity = b.AuthorCity

and

a.AuthorId < b.AuthorId

A003 Shakespeare London A004 Shelly London

A001 Kuvempu Mysore A006 AnaKru Mysore

========================================================================================================================

Day 03 - 28 Sep 2020

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

MySql

Understanding Features[Completed]

RDBMS vs DBMS[Completed]

RDBMS vs DBMS[Completed]

MySql[Completed]

Understanding DDL,DML & DCL differences[Completed]

Datatypes[Completed]

Creating Tables[Completed]

Basic Queries[Completed]

Primary Keys/Foreign Keys/Candidate Keys/Alternate Keys/Composite Key[Completed]

Integrity/Entity Integrity/Domain Integrity/Referential Integrity[Completed]

Constraints[Completed]

--- Entity Constraints

--- Check Constraint

--- Unique Key Constraint

--- Primary Key Constraint

--- Foreign Key Constraint

CRUD[Completed]

C-Create[Completed]

R-read[partially Completed]

U-Update[Completed]

D-Delete[Completed]

DDL - Data Definition Language[create/alter/drop][Completed]

DQL/DML- Data Manipulation Language[select/insert/delete/update][ Completed]

DCL - Data Control Language[Grant/revoke][Not Completed]

DTL-Data Transaction Language[commit/save transaction/rollback][Not Completed]

Datatypes-MySql[Completed]

Creating Tables[Completed]

Basic Queries[Completed]

Operators - [NOT Completed]

Arithmetic/Logical/Comparison[NOT Completed]

in any between and or operators[NOT Completed]

like operator[NOT Completed]

Functions[NOT Completed]

Literals [NOT Completed]

Keys:Primary Keys/Foreign Keys/Candidate Keys/Alternate Keys/Composite Key[Partially Completed]

Order by asc desc[NOT Completed]

Group by / Having[[Not Completed]]

Database-Integrity/Entity Integrity/Domain Integrity/Referential Integrity/User Defined[Partially Completed]

=========================================================================================================================

RDBMS vs DBMS

DBMS:

Foxbase

DBase1/2/3

Foxpro

MSAccess

---------------------------------------------

RDBMS

Oracle/MySql/MSSql/Ingres...../MSAccess

RDBMS - Relational Databases

DBMS -

------------------------------------------------------------------------------------------------------

RDBMS --- tabular form

DBMS -----

------------------------------------------------------------------------------------------------------

DBMS-

Integrity Constraints

Security/DB - Integrity/Database Consistency

DB Integrity: Valid Database

Relation vs Table

Tuple vs Record

Authors

AuthorId AuthorName AuthorAddress eMail City TitleID Title Publisher

A1 Shelly Lancashire shel@gmail.com Lancashire T1 XYZ Publisher

A1 Shelly Lancashire shel@gmail.com Lancashire T2 XYZ Publisher

A1 Shelly Lancashire shel@gmail.com Lancashire T3 XYZ Publisher

A2 Keats Manchester Keat@gmail.com Manchester T4 XYZ Publisher

A2 Keats Manchester Keat@gmail.com Manchester T5 XYZ Publ isher

A2 Keats Mancheste Keat@gmail.com Manchester T6 XYZ Publisher

Authors

Primary Key

AuthorId AuthorName AuthorAddress eMail

A1 Shelly Lancashire shel@gmail.com Lancashire

A2 Keats Manchester Keat@gmail.com Manchester

Books

Primary Key ForeignKey

TitleID Title Publisher AuthorId

0x12 T1 XYZ Publisher A1

T3 XYZ Publisher A1

T2 XYZ Publisher A1

T4 XYZ Publisher A2

T5 XYZ Publisher A2

T6 XYZ Publisher A2

------------------

Employees Attendance Performance Department

E1 Addres Aid E1 Pid E1 Did E1

-----------------

Primary Key - Data Sharding

Clusters - Bucket Hashing

- X - Indexing

----------------------

Security

Integrity

Indexing

Consistency

Scalability

Transaction - ACID

RDBMS

E.F.Codd

12 Codd Rules

---------------------------------------------------------------------------

Transaction -

Operation -

Transaction - Its a Logical unit which consists of several SQL statements .

Transaction is nothing but several sql statements sent to the server as a single logical unit

---------------------------------------------------------------------------

Client - Server

Employee

EmpId PosCode Dept

E1 AM-M Engg

E2 M Engg

E3 C Accts

------------------------

Position

PosCode Description BudgetedStrenth ActualStrength

AM Asst Manager 1200 1100 - 1

M Manager 800 750 + 1

C Clerk 3000 2750

------------------------------------

Structured Query Language

Atomicity: Feature by which , all the statements of a transaction get executed or none of them excecute

Begin Transaction

Update Employee

set PosCode = 'M' where EmpID='E1'

Update Position

set ActualStrength = ActualStrength - 1

where PosCode = 'AM'

Update Position

set ActualStrength = ActualStrength +1

where PosCode = 'M'

Commit Transaction

--------------------------------

Update Employees

Update Customers

Update Suppliers

-------------------------------

Isolation

Every Transaction should be isolated from each other -

Transaction 9.30.45 9.30.45

Blore - Chennai 5 B.lore-Chennai 5

DB - 100 - 5

-------------------------------------------------------------

Durability

----------------------------------------------------------------

MySql -- Open Source LAMP

[Linux - Apache - MySql PHP]

----------------------------------------------------------------

Schema - Structure

DDL - Data Definition Language - Create / Alter /Drop

DML/DQL - Data Manipulation - Update/Insert/Delete /Truncate/

DCL - Grant / Revoke

DTL - Data Transaction /Begin/Commit/Save / Rollback

====================================================T Break 4.10 to 4.25 Break===================================

Constraint :

Entity Integrity - Primary Key [Primary Key Constraint, Unique Constraint]

Domain Integrity - Check Constraint / Default Constraint

Referential Integrity - Foreign Key Constraint

User Defined -

=================================Objects of Database

Tables

Cursors

Indexes

Triggers

Views

Constraints

Procedures

Functions

Rules

===============================

Delete can delete row by row and we can do conditionally; Deletes only records ; Structure is Retained after deletion

Drop ; Table is dropped along with the data; Structure is also deleted

Truncate; Structure is retained ; Rows are deleted in a single shot;

======================================================================================================

Soft Delete-LogicalDelete / Hard Delete

EmpId EmpName Addres Deleted

E1 Harsha RTnagar NO

E1 Harsha RTnagar NO

E1 Harsha RTnagar NO

E1 Harsha RTnagar YES

=========================================================================================================================

Day 02 - 25 Sept 2020

https://www.uml-diagrams.org/state-machine-diagrams.html

Interaction Diagrams

- Sequence Dia

- Communication Dia

State Chart Diagram[start- transition-join fork - choicestate]

Activty Diagram

ACTOR: stakeholders

AnyBody who interacts with the system:

PrimaryActor: Directly involved Clerk,Teller,Customer , Manager

SecondaryActor: Taxation Agency

OffStage Actors : paypal-service in an online shopping system

-----------------------------------------------------------------------

Order:

Try Finishing The Diagrams for your Project:

I will help you, if you need any help;

1) SRS

2) Use Case

3) Class Diagram

4) System Sequence Diagram

------------------------------------------------------------------------T BREAK 11.25 to 11.40 am--------------------------------------------------------------------------------

REVIEWS OF ASSOCIATES UML

------------------------------------------------------------------------LUNCH BREAK 1.15 pm to 2.00 pm----------------------------------------------------------------------

PLAN FOR THE DAY :

POST LUNCH :

REVIEW UML DIAs

DEMO UML - ER , DFD, Sequence Diagram , StateChart & Activity Diagram

SAT PLAN: Associates to complete maximum artifacts ,

by Monday First Half - Freeze DEsign Work

SECOND HALF : RDBMS - MySql:

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Login

loginId

password

Name

Role : patient/doctor/admin

Patient [RegisterPatient /RequestAppointment/ ViewDoctorsdetails/ViewAvailableDoctors]

Doctor - Id/Name/License/Specialization [RegisterData/GrantAppoint]

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

DFD - Flow of activities that takes place in the System/process with respect to the Data

Data Store

Entity

Functionality

==========================================================================

Database Design plays vary important role for any system, Database and its tables and their relationships have to be properly established,

In the Architectural style like MVC

Model represents data, corresponding to every Model class in the front end, we need to have a table in the back end

therefore planning of these entities are very critical

DB design can be easily/clearly represented using ER diagram.

-------------------------------------------------------------------------------------------------------------------

Interaction Diagrams are basically representing the interactions among the entities of a system;

1) System Sequence Diagram(SSD)

2) Communication Diagram

-----

SSD depicts the sequence of events giving rise to functions, here Object Lifeline is significant , emphasizing the process events and their sequence wrt the Objects/Entities

SSD represents/depicts the interactions that takes place in a system through various events, that are invoked by the actors[],

SSD always shows the sequence in which the events take place as well as the invocation of the functionalities wrt the various Objects.

multiple SSDs can be depicted for a system which is very complex with huge number of scenarios, not necesssarily every system ends up with a single SSD, until and unless it is a small and

simple system.

-----------------------------------------------------------------------------------------------------------

Activity Diagram : focuses on the flow of activity for a scenario, signifies the Actions / Events and their resultant functionalities in multiple channels for a given scenario/system

One can have multiple Activity Diagrams for a complex system.

multiple Activity Diagrams can be depicted for a system which is very complex with huge number of scenarios, not necesssarily every system ends up with a single SSD, until and unless it is a small and

simple system.

One can have multiple swim lanes to depict the various channels in the process

--------------------------------------------------------------

State machine Diagram / State Chart Diagram emphasizes the states of the entity, during the sequence of activities within a scenario

States categorized into simple state/complex state

==========================================================

Day 01 - 24 Sep 2020

Introduction Students + Trainer[Completed]

Brief Idea about How Real Time Projects work[Completed]

Need for DevOps[Completed]

Why Dev Ops Based Eco System[Completed]

Project Allocation[Completed]

Discussion On Project Topics - Interactive[Completed]

What is Expected From the Students in Completing Project[Completed]

Explaining briefly about Various Technologies That are to be learnt from Project Perspective.[Completed]

The Brief Idea about Various Tools / Usage in Projects - Linking All tools Brief Overview[Completed]

Making them Understand why learning these Technologies & tools[Completed]

Thereby Enabling them to Understand the Need for EcoSystem [Completed]

Making Them Come out With Architectural Diagram/Approach[IN PROGRESS]

Groups to Come out With DB Design - Briefly atleast - to be tuned in due course[IN PROGRESS]

Groups to Come Out With Flow Diagram- to be tuned in due course[IN PROGRESS]

Groups To come out with UML Diagram(UseCase/Sequence/Class/Activity)- to be tuned in due course[IN PROGRESS]

========================================================================================================================

Dev-Ops - Agile

Aims shortening SDLC

Release software more faster /

communicates & collaborates among business

practices

--------------------------------------------------------------------------------------------------

Quality Aspects - CORRECTNESS-Reliability/Performance/Security/Durability/Scalability/Portability/Maintenability/UserFriendliness/Deployablility

Engineer -

Code Review

-----------------------------

IDE - PROJECT MGMT SYSTEM / ISSUE TRACKING SYSTEM - SCM - BUILD - UNIT TESTING -CODE REVIEW - INTEGRATION - INT-TESTING-

Eclipse - JIRA - GIT - MAVEN -------JUNIT - CRUCIBLE - JUNIT/SELENIUM - JENKINS

Code Review - Crucible

DEPLOY/INTEGRATION [ Architecture-ecosystem - Devops]

--------------------------------------------------------

STRUCTURAL ELEMENTS

Approach - style

Platform

H/W/S/w - Technologies/environment

Business Strategy

Quality Attributes []

Language - Technology

Framework

Complexity

Documentation

Database- DB connectivity principle/Data Access Technique

Artifacts - Design Tools -

DataStructures

Algorithms-

Design Patterns

----------------------------------------------------------------------------------------------------------------------------------

BREAK - 11.05 - 11.20

-----------------------------------------------------------------------------------------------------------------------------------

Doubts:

Q&A

Maven Build

Build -

-----------------------------------------------------------------------------------------------------------------------------------

Upto 12.30 extended upto 12.35 Project Analysis - Selection By Team

---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Grady Booch - Booch Priciples

Ivor Jacobson - OMT

James Rumbaugh - OOSE

--------------------------------------------------------------

UML - Unified Modelling Language -

Static Modelling / Dynamic Modelling- Activity Diagram

---------------------

13 Diagrams

===============================================================LUNCH BREAK 1.05 to 1.50=========================================================

OOAD - UML

[Inception - Analysis - Design - Construction - Testing - Deployment - Maintenance/Support

Requirement Anaylsis

Requirement - Gathering Techniques

Brain Storming

Questionnaire

Meetings

SRS - [ERS]- Freezing Requirements

---------------------------------------------------------------------

Use case Diagram vs Use Case - Descriptive [ Casual / Fully Dressed - Scope/Intension/Extension / Terse]

OOAD - Class

Instantiation

Realization - a Concrete class from [Interface] -

Specialization --

Generalization

Composition --

Aggregation --

Association ---

----------------------------------------------------------------------------------------------------------------------------------------

Composition : a scenario where a Contained entity loses its scope/ceases to exist, when a containing entity

goes out of scope/ceases to exist

Aggregation : a scenario where a Contained entity independently exists inspite of the ceasing of the Containing entity

----------------------------------------------------------------------------------------------------------------------------------

Identifying Entity -

Domain Modelling - Real life entities [ Identifying Real Life Objects and identifying chracteristics]

---------------------------------------------------------------------------------------------------------------------------------

List myList;

myList = new ArrayList();

myList = new LinkedList()

Account

{

}

SBAccount RDAccount FDAccount InsuranceAccount SB

|

|leads me to coming out with Class Diagram

class Address

{

String dNo;

String street;

}

class Taxation

{

}

class Invoice

{

Taxation t;

}

class Employee

{

String employeeId;

Address employeeAddress;

public void getEmployeeDetails()

{

}

--------------------------------------------------------------------------------------------------------------------------------

Today

- SRS

- UseCase

- Class Diagram

---------------------------------------------------------------------------------------------------------------------------------

Tomorrow

- - DFD

- DB design

- ER Diagram

- System Sequence Diagram

- Activity Diagram

- State Chart Diagram

- Architecture Diagram

----------------------------------------------------START UML DIAGRAMS ON PROJECT -----------------------------------UPTO 5.30------------------------

----------------------------------Students Interaction--------------------------------------------------------------------------

1)Class combination of methods/functions & objects -

2) Data members

-------------------------------------------

}