**DESIGN AND ANALYSIS OF ALGORITHMS**

**UNIT I**

Introduction to algorithms and its importance, Fundamentals of the analysis of algorithm efficiency - analysis frame work - Asymptotic notations – Big Oh Notation, Omega Notation, Theta Notation and Little Oh Notation.

**UNIT II**

Sorting algorithms design and analysis: Insertion sort, selection sort Divide and conquer - General method, applications – binary search, quick sort, merge sort, Strassen’s matrix multiplication.

**UNIT III**

**Greedy method** – General method, applications – job sequencing with deadlines, 0/1 Knapsack problem, minimum cost spanning trees, single source shortest path problem.

**UNIT IV**

Dynamic Programming - General method, applications, matrix chain multiplication, optimal binary search trees, 0/1 knapsack problem.

**UNIT V**

All pairs shortest path problem, traveling sales person problem, reliability design. Backtracking – General method, applications – n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

**UNIT VI**

Branch and Bound - General method, applications–travelling sales person problem, 0/1 knapsack problem – LC Branch and bound solutions, FIFO branch and bound solutions. Np-hard and Np-complete Problems- Basic concepts, non-deterministic algorithms, NP-Hard and NP-Complete classes, Cook’s Theorem.

**TEXT BOOKS**

1. Ellis Horowitz, SartajSahni and Sanguthevar Rajasekaran, “Fundamental of Computer Algorithms”, 2nd Edition, Universities Press.
2. Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C., “Introduction to Algorithms”, 3rd Edition, Prentice Hall of India Pvt. Ltd.

**REFERENCE BOOKS**

1. Anany V. Levitin “Introduction to the Design and Analysis of Algorithms” Villanova University ©2003 ISBN: 0-201-74395-7
2. Sara Baase and Allen Van Gelder.. Computer Algorithms: Introduction to Design and Analysis , Pearson education (Singapore) Pvt. Ltd, New Delhi.
3. Aho, A.V., Hopcroft J.E. and Ullman, J.D., “The Design and Analysis of Computer Algorithms”, Pearson Education.
4. Robert Sedgewick, Kevin Wayne, “Algorithms”, 4th Edition.
5. SanjoyDasguptha, Christos Papadimitriou, UmeshVazirani, “Algorithms”, 1st Edition.

**DATABASE MANAGEMENT SYSTEMS**

**UNIT I**

Introduction: Data base System VS file System, View of Data, Data Abstraction, Instances and Schemas, data Models, Database Languages and Data base System Structure.

Data base design and ER diagrams – Beyond ER Design Entities, Attributes and Entity sets Relationships and Relationship sets, Additional features of ER Model, Concept Design with the ER Model

**UNIT II**

Introduction to the Relational Model- Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views, Destroying /altering Tables and Views.

Relational Algebra – Selection and projection set operations, renaming, Joins, Division, Examples of Algebra overviews, Relational calculus, Tuple relational Calculus, Domain relational calculus.

**UNIT III**

Introduction to Schema Refinement - Problems Caused by redundancy, Decompositions- Problem related to decomposition, Functional Dependencies- Reasoning about FDS, Normal Forms - FIRST, SECOND, THIRD Normal forms - BCNF - Properties of Decompositions - Loss less join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design - Multi valued Dependencies - FOURTH Normal Form.

**UNIT IV**

Transaction Concept- Transaction State- Implementation of Atomicity and Durability – Concurrent – Executions – Serializability- Recoverability – Testing for serializability.

**UNIT V**

Lock –Based Protocols – Timestamp Based Protocols - Validation- Based Protocols.

Recovery and Atomicity – Log - Based Recovery – Recovery with Concurrent Transactions – Buffer Management

**UNIT VI**

Overview of Storage and Indexing: Data on External Storage – File Organization and Indexing – Hash Based Indexing – Tree base Indexing

**TEXT BOOKS**

* 1. Raghurama Krishnan, Johannes Gehrke“ Data base Management Systems” TATA McGrawHill 3rd Edition.
  2. Silberschatz, Korth “Data base System Concepts” McGraw hill, V Edition.

**REFERENCES**

* + - 1. Peter Rob and Carlos Coronel “Data base Systems design, Implementation, and Management” 7th Edition.
      2. ElmasriNavrate “Fundamentals of Database Systems” Pearson Education

**WEB LINKS:**

1. http://nptel.iitm.ac.in
2. http://highered.mheducation.com/sites/0072465638/student\_view0/index.html

**COMPUTER NETWORKS**

**UNIT I**

Introduction – network architecture –network topologies- network design. Reference models- The OSI Reference Model- The TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models

Physical layer: Transmission media, Introduction to Encoding.

**UNIT II**

Datalink Layer: Channelization, Multiplexing, Framing, error detection and correction, flow control-sliding window protocol, Medium access control sub layer, Basic structure of a switch, circuit switching and packet switching, Ethernet, 802.11/WiFi, WiFi-Direct, Bluetooth, NFC, RFID.

**UNIT III**

Network layer –Routing algorithms- Unicast, Multicast, Broadcast, Intra domain and Inter domain routing, Source based tree and group shared tree. Internetworking- Internet Protocols (IPv4 and IPv6).

**UNIT IV**

Transport layer - Elements of transport protocol-Congestion control – Performance issues-The Internet’s Trans-mission Control Protocol (TCP)- Remote Procedure Call (RPC)- – Implementation semantics of RPC -client-server applications- The Real-time Transport Protocol(RTP) - Multimedia applications- Congestion control and resource allocation.- congestion control in TCP –UDP –Quality of service in IP.

**UNIT V**

Application layer - Domain name server-World wide web-Hyper text transfer protocol-Presentation formatting and data compression- Network security- crypto graphic tools- the problems of key distribution – General authentication techniques.

**UNIT VI**

Network applications and the protocols - File transfer protocol - email and the Web, multimedia applications such as IP telephony and video streaming- Overlay networks like peer-to-peer file sharing and content distribution networks- Web Services architectures for developing new application protocols.

**TEXT BOOKS**

1. Andrew S. Tanenbaum, David J Wetherall, *Computer Networks*, 5th Edition, Pearson Edu, 2010.
2. James F. Kurose, Keith W. Ross, “Computer Networking, A Top-Down Approach Featuring the Internet”, Sixth Edition, Pearson Education, 2013.

**REFERENCE BOOKS**

1. William Stallings, “Data and Computer Communications”, Eighth Edition, Pearson Education, 2011
2. Nader F. Mir, “Computer and Communication Networks”, First Edition, Pearson Education, 2007.
3. Ying-Dar Lin, Ren-Hung Hwang and Fred Baker, Computer Networks: An Open Source Approach “, McGraw Hill Publisher, 2011
4. Behrouz A. Forouzan, “Data communication and Networking”, Tata McGraw-Hill, 2004.

**WEB LINKS:**

1. http://nptel.ac.in/video.php?subjectId=106105081.
2. http://wps.pearsoned.com/ecs\_kurose\_compnetw\_6/216/55463/14198700.cw/

**PROBABILITY AND STATISTICS**

**UNIT I**

**Probability**

Sample space – events – Probability – The axioms of probability – Some Elementary theorems - Conditional probability – Baye’s theorem.

**UNIT II**

**Random Variables and Distributions**

Discrete and continuous Random Variables – Distributions – Distribution function – Binomial, Poison and Normal distributions - Sampling distribution: sample – population – statistic - parameter – standard error.

**UNIT III**

**Testing of Hypothesis-I**

Estimation: Point estimation – Interval estimation, Test of Hypothesis: Null Hypothesis – Alternative Hypothesis – Type1 and Type2 errors – One tailed and two tailed tests – Critical Region – level of significance. Large Sample Tests: Test for single mean and difference of means, test for single proportion, difference of proportions.

**UNIT IV**

**Testing of Hypothesis-II**

Student t-distribution – F distribution - χ2-distribution, Small Sample Tests: Test of significance for single mean and difference of means, Test for equality of variances (F-test), χ2-test for goodness of fit.

**UNIT V**

**Curve Fitting**

Principle of least squares - working procedure for fitting curves - fitting of straight line – parabola – exponential curve – power curve.

**UNIT VI**

**Correlation and Regression**

Correlation – Correlation co-efficient – Karl Pearson’s coefficient of Correlation – Spearman’s Rank correlation coefficient – Regression – Regression equation of X on Y – Regression equation of Y on X (only linear).

**TEXT BOOKS**

* + - 1. Richard Arnold Johnson, Irwin Miller and John E.Freund, “Miller and Freund’s Probability and Statistics for Engineers”, Prentice Hall PTR.
      2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, S. Chand and Sons.

**REFERENCE BooKs**

Sheldon P. Gordon, Contemporary Statistics: A Computer Approach, McGraw Hill..

Kishore S. Trivedi, Probability Statistics with Reliability, Queuing and Computer Science Applications, Prentice Hall.

Iyengar TKV, MVSN Prasad, S. Ranganatham, Gandhi and B. Krishna, “Probability and Statistics,” S Chand.

M. R. Spiegel, J. Schiller, Probability and Statistics, Schaum Series.

**PRINCIPLES OF PROGRAMMING LANGUAGES**

**UNIT I**

**Preliminary Concepts**

Reasons for studying, concepts of programming languages, Programming domains, Language Evaluation Criteria, influences on Language design, Language categories, Programming Paradigms: Imperative, Object Oriented, functional Programming, Logic Programming. Programming Language Implementation, Compilation and Virtual Machines, Programming environments.

**UNIT II**

**Syntax and Semantics**

General Problem of describing Syntax and Semantics, formal methods of describing syntax, BNF, EBNF for common programming languages features, parse trees, ambiguous grammars, attribute grammars, Denotation semantics and axiomatic semantics for common programming language features.

**UNIT III**

**Data Types**

Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types, design and implementation uses related to these types. Names, Variable, concept of binding, type checking, strong typing, type compatibility, named constants, variable initialization. Expressions and Statements: Arithmetic relational and Boolean expressions, Short circuit evaluation mixed mode assignment, Assignment Statements, Control Structures: Statement Level, Compound Statements, Selection, Iteration, Unconditional Statements, guarded commands.

**UNIT IV**

**Subprograms and Blocks**

Fundamentals of sub-programs, Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, parameters that are sub-program names, design issues for functions user defined overloaded operators, co routines.

**UNIT V**

**Abstract Data Types**

Abstractions and encapsulation, introductions to data abstraction, design issues, language examples, C++ parameterized ADT, object oriented programming in small talk, C++, Java, C#, Ada 95, Concurrency: Subprogram level concurrency, semaphores, monitors, massage passing, Java threads, C# threads. Exception handling: Exceptions, exception Propagation, Exception handler in Ada, C++ and Java. Logic Programming Language: Introduction and overview of logic programming, basic elements of prolog, application of logic programming.

**UNIT VI**

**Functional Programming Languages**

Introduction, fundamentals of FPL, LISP, ML, Haskell, application of Functional Programming Languages and comparison of functional and imperative Languages. Scripting Language: Pragmatics, Key Concepts, Case Study: Python – Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library.

**TEXT BOOKS**

1. Robert W. Sebesta, “Concepts of Programming Languages”, 8th Edition, Pearson Education, 2008.
2. D. A. Watt, “Programming Language Design Concepts”, Wiley Dreamtech, 2007.

**REFERENCE BOOKS**

1. A.B. Tucker and R.E. Noonan, “Programming Languages”, 2nd Edition, TMH.
2. K. C Louden, “Programming Languages”, 2nd Edition, Thomson, 2003.

**DISCRETE MATHEMATICAL STRUCTURES - II**

**UNIT I**

Groups – Permutation groups, Lagrange’s theorem, Normal subgroups, Factor groups, Fundamental theorem of finite Abelian groups

**UNIT II**

Rings: Integral Domains, Ideals and factor rings Polynomial rings, factorization of polynomials, unique factorization domains, Euclidean domains

**UNIT III**

Fields, Vector spaces, Linear independence, splitting fields, irreducible polynomials Algebraic extension of fields, Finite fields

**UNIT IV**

Sylow theorems, Finite simple groups, Symmetry groups, generators and relations.

**UNIT V**

Introduction to algebraic coding theory, Linear codes, Reed-Solomon codes.

**UNIT VI**

Introduction to Galois theory, fundamental theorem of Galois theory, Solvability of polynomials by radicals, Insolvability of a quintic.

**TEXTBOOKS**

1. 1. Contemporary Abstract Algebra, Joseph A. Gallian, 4th Edition, Narosa publishing house

**COMPUTER NETWORKS LAB**

**Week 1**

1. NIC Installation & Configuration (Windows/Linux)
2. Familiarization with Networking cables (CAT5, UTP) Connectors (RJ45, T- connector), Switches, and Routers.

**Week 2**

Programs to implement error correction and detection

**Week 3**

Implementation of CRC - CCITT (16-bits)

**Week 4**

Implementation of Routing algorithms

**Week 5**

Implementation of Stop and Wait Protocol and Sliding Window Protocol

**Week 6**

Connection oriented Client server applications with TCP

**Week 7**

Connectionless Client server applications with UDP

**Week 8**

Assignment-6 Programs using RPC remote procedure call

**Week 9**

Simulate using Ns2 which consist of a three nodes point-to-point network with duplex links between them. Set the queue size vary the bandwidth and find the number of packets dropped and plot graphs using XGRAPH.

**Week 10**

Simulate using Ns2 which consist of a four node point-to-point network, and connect the links as follows: n0-n2, n1- n2 and n2-n3. Apply TCP agent between n0-n3 and UDP n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets sent by TCP/UDP and plot graphs using XGRAPH.

**Week 11**

Simulate using Ns2 which consist of the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion and plot graphs using XGRAPH..

**Week 12**

1. Design TCP iterative Client and server application to count the number of vowels present in given input sentence
2. Design TCP client and server application to transfer file

**Week 13**

Client server applications using Multi protocol server

**Week 14**

Implement a chat and mail server

**DESIGN AND ANALYSIS OF ALGORITHMS LAB**

**LIST OF EXPERIMENTS**

1. Write C programs to implement the following:
   1. Prim’s algorithm.
   2. Kruskal’s algorithm.
2. Write a C program to find optimal ordering of matrix multiplication. (Note: Use Dynamic programming method).
3. Consider the problem of eight queens on an (8x8) chessboard. Two queens are said to attack each other if they are on the same row, column, or diagonal.
4. Write a C program that implements backtracking algorithm to solve the problem i.e. place eight non-attacking queens on the board.
5. Write a C program to find the strongly connected components in a digraph.
6. Write a C program to implement file compression (and un-co impression) using Huffman’s algorithm.
7. Write a C program to implement dynamic programming algorithm to solve all pairs shortest path problem.
8. Write a C program to solve 0/1 knapsack problem using the following:
   1. Greedy algorithm.
   2. Dynamic programming algorithm.
9. Write a C program to solve 0/1 knapsack problem using the following:
10. Backtracking algorithm.
11. Branch and bound algorithm.
12. Write a C program that uses dynamic programming algorithm to solve the optimal binary search tree problem.
13. Write a C program for solving traveling sales persons problem using the following:
14. Dynamic programming algorithm.
15. The back tracking algorithm.
16. Write a C program for solving traveling salespersons problem using Branch and Bound.

**DATABASE MANAGEMENT SYSTEMS LAB**

Recommended Systems/Software Requirements:

1. Intel based desktop PC
2. Mysql /Oracle latest version Recommended
3. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
4. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS,UNION, INTERSET, Constraints.

Example: Select the roll number and name of the student who secured fourth rank in the class.

1. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
2. Queries using Conversion functions (to\_char, to\_number and to\_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next\_day, add\_months, last\_day, months\_between, least, greatest, trunc, round, to\_char, to\_date)
3. Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
4. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in

**PL/SQL block**

1. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
2. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
3. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
4. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
5. Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.
6. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
7. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

**Roadway Travels Database**

The student is expected to practice the designing, developing and querying a database in the context of example database “Roadway travel”. Students are expected to use “Mysql” database.

**"Roadway Travels"** is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad. The company wants to computerize its operations in the following areas:

• Reservations

• Ticketing

• Cancellations

**RESERVATIONS**

Reservations are directly handled by booking office. Reservations can be made 60 days in advance in either cash or credit. In case the ticket is not available, a wait listed ticket is issued to the customer. This ticket is confirmed against the cancellation.

**CANCELLATION AND MODIFICATIONS**

Cancellations are also directly handed at the booking office. Cancellation charges will be charged.

Wait listed tickets that do not get confirmed are fully refunded.

**Task 1: E-R Model**

Analyze the problem carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc.

Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Example: **Entities:**

1. BUS

2. Ticket

3. Passenger

**PRIMARY KEY ATTRIBUTES**

1. Ticket ID (Ticket Entity)

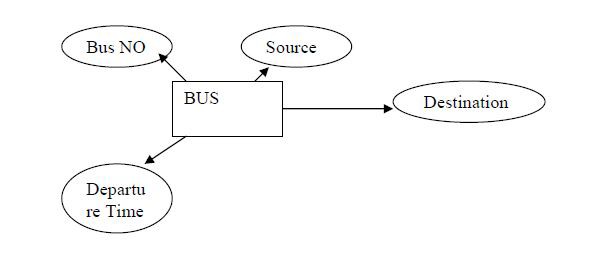
2. Passport ID (Passenger Entity)

Apart from the above mentioned entities you can identify more. The above mentioned are few.

**Task 2: Concept Design with E-R Model**

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total / partial). Try to incorporate generalization, aggregation, specialization etc wherever required.

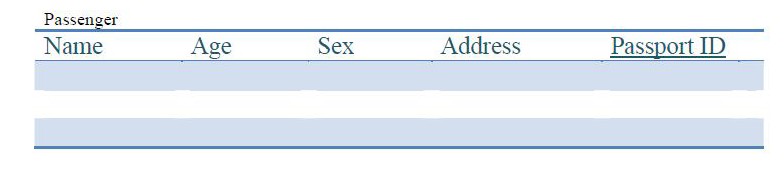
**Example: E-r diagram for bus**



**Task 3: Relational Model**

Represent all the entities (Strong, Weak) in tabular fashion.Represent relationships in a tabular 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. Different types of attributes (Composite, Multivalued, and Derived) have different way of representation.

Example: The passenger tables look as below. This is an example. You can add more attributes based on your E-R model.



**Task 4: Normalization**

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind, because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only.

**Task 5: practicing DDL, DML commands**

In this task you will learn Creating databases, How to create tables, altering the database, dropping tables and databases If not required. You will also try truncate, rename commands etc.

Example for creation of a table. CREATE TABLE Passenger (Passport id INTEGER PRIMARY KEY, Name CHAR (50) NULL,

Age Integer, Sex Char;

**Note: Detailed creation of tables is given at the end. Practicing DML commands**

DML commands are used to for managing data within schema objects. Some examples:

• SELECT - retrieve data from the a database

• INSERT - insert data into a table

• UPDATE - updates existing data within a table

• DELETE - deletes all records from a table, the space for the records remain

**Few more Examples of DML commands:**

Select \* from Bus; (selects all the attributes and display) UPDATE BUS SET Bus No = 1 WHERE BUS NO=2;

**Task 6: Querying**

In this week you are going to practice queries (along with sub queries) using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

**Practice the following Queries:**

1. Display unique PNR\_no of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Display the source and destination having journey time more than 10 hours.
5. Find the ticket numbers of the passengers whose name start with ‘A’ and ends with ‘H’.
6. Find the names of passengers whose age is between 30 and 45.
7. Display all the passengers names beginning with ‘A’
8. Display the sorted list of passengers names
9. Display the Bus numbers that travel on Sunday and Wednesday
10. Display the details of passengers who are traveling either in AC or NON\_AC(Using only IN operator)

**Task 7: Querying (continued…)**

You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and

MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

• Write a Query to display the Information present in the Passenger and cancellation tables.

**Hint:** Use UNION Operator.

• Write a Query to display different travelling options available in British Airways.

• Display the number of days in a week on which the 9W01 bus is available.

• Find number of tickets booked for each PNR\_no using GROUP BY CLAUSE. **Hint:** Use

GROUP BY on PNR\_No.

• Find the distinct PNR numbers that are present.

• Find the number of tickets booked in each class where the number of seats is greater than 1.

**Hint:** Use GROUP BY, WHERE and HAVING CLAUSES.

• Find the total number of cancelled seats.

• Write a Query to count the number of tickets for the buses, which travelled after the date '14/3/2009'. **Hint:** Use HAVING CLAUSES.

**Tables**

**BUS**

Bus No: Varchar: Pk Source :Varchar Destination :Varchar

**Passenger**

PNR\_No : Numeric(9) : PK Ticket\_No: Numeric (9) Name: Varchar(15)

Age :int (4)

Sex:Char(10) : Male / Female

PPNO: Varchar(15)

**Reservation**

PNR\_No: Numeric(9) : FK Journey\_date :datetime(8) No\_of\_seats :int (8) Address :Varchar (50)

Contact\_No: Numeric (9) -->Should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

**Cancellation**

PNR\_No: Numeric(9) : FK Journey\_date :datetime(8) No\_of\_seats :int (8) Address :Varchar (50)

Contact\_No: Numeric (9) -->Should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

**Ticket**

Ticket\_No: Numeric (9): PK Journey\_date :datetime(8) Age : int (4)

Sex: Char(10) : Male / Female

Source :Varchar Destination : Varchar Dep\_time : Varchar