

DNA PROJECT

PHASE - 1

MINI-WORLD - ZOO MANAGEMENT SYSTEM

INTRODUCTION TO THE MINI-WORLD : Our mini-world is the “Zoo Management System”. It describes about the outline of a zoo and gives a bit of knowledge of species in it.

PURPOSE OF THE DATABASE: Our main motivation to take this as our database was to help animal lovers where to visit in order to see the species they love. This database stores the information about zoos. Our database also stores the data about the animals and their species. Not only this, our database keeps track of Employees working in it.

USERS OF THE DATABASE: The data could be used by the people who are interested in knowing about zoos and who might want to analyze the logistics of it. A manager will regularly access this in order to keep track of the zoo.

APPLICATIONS OF THE DATABASE:

- The visitors can decide which zoo they can visit in order to see more number of species.
- People who are curious about knowing much more about animals can use this database. For example we can decide which animal will live for more time.
- This database can also be used by students if they are doing any research on zoos. For example, will zoos exist for more number of years.
- If any new employee joins the zoo he can use this database to analyze the zoo.

DATABASE REQUIREMENTS:

Assumptions :

- We will atleast have M number of species in a zoo.
- A species will exist in at least one zoo.
- A species will have at least one animal.
- An employee must work only in one zoo.

- One zoo will have at least M employees.
- A zoo will have one ticket counter and each ticket counter is managed by only one employee.

Strong Entity types :

- Zoo :
 - ◆ Name - Key Attribute
 - ◆ Location - Key Attribute
 - ◆ Number of Customers Visited - Derived Attribute
 - ◆ Number of Employees - Derived Attribute
 - ◆ Number of Species - Derived Attribute
- Customer:
 - ◆ Customer ID - Key Attribute
 - ◆ Phone Number - Key Attribute
 - ◆ Name - Composite Attribute
 - ◆ Gender - Simple Attribute
 - ◆ Age - simple Attribute
- Species :
 - ◆ Name - Key Attribute
 - ◆ Evolved in - Simple Attribute
 - ◆ Food - Multi valued Attribute

Weak Entity types :

- Animal :
 - ◆ Zoo Name - Partial Key (Composite Attribute)
 - ◆ Height - Simple Attribute
 - ◆ Weight - Simple Attribute
 - ◆ Color - Multi Valued Attribute
 - ◆ Age - Simple Attribute
- Employee :
 - ◆ Zoo Name - Partial Key (Composite Attribute)
 - ◆ Employee_ID - Partial Key (Composite Attribute)
 - ◆ Age - Simple Attribute

- ◆ Gender - Simple Attribute
- ◆ Salary - Simple Attribute
- Ticket Counter :
 - ◆ Zoo Name - Partial Key (Composite Attribute)
 - ◆ Ticket Number - Partial Key
 - ◆ Ticket Valid Date - Simple Attribute
 - ◆ Price for Children - Simple Attribute
 - ◆ Price for Adults - Simple Attribute

Relationship Types :

- VISITS :
 - ◆ Degree - 2
 - ◆ Participating Entity types - Customer and Zoo
(Customer VISITS Zoo)
 - ◆ Cardinality ratio - M:N
 - ◆ Constraints - Customer (0,N) - Zoo(0,N)
- CONTAINS :
 - ◆ Degree - 2
 - ◆ Participating Entity types - Zoo and Species
(Zoo CONTAINS Species)
 - ◆ Cardinality ratio - M:N
 - ◆ Constraints - Zoo(M,N) - Species(1,N)
- BELONGS_TO:
 - ◆ Degree - 2
 - ◆ Participating Entity types - Animal and Species
(Animal BELONGS_TO Species)
 - ◆ Cardinality ratio - N:1
 - ◆ Constraints - Animal(1,1) - Species(1,N)
- WORKS_IN:
 - ◆ Degree - 2
 - ◆ Participating Entity types - Employee and Zoo
(Employee WORKS_IN Zoo)
 - ◆ Cardinality ratio - N:1

- ◆ Constraints - Employee(1,1) - Zoo(M,N)

- BUYS_TICKET_FROM :

- ◆ Degree - 2
- ◆ Participating Entity types - Customer and Ticket Counter
(Customer BUYS_TICKET_FROM Ticket Counter)
- ◆ Cardinality ratio - 1:1
- ◆ Constraints - Customer(1,1) - Ticket Counter(1,1)

- TAKEN_CARE_BY:

- ◆ Degree - 2
- ◆ Participating Entity types - Species and Employee
(Species TAKEN_CARE_BY Employee)
- ◆ Cardinality ratio - M:N
- ◆ Constraints - Employee(0,N) - Species(1,N)

Degree > 2 Relationship Types :

- MANAGES :

- ◆ Degree - 3
- ◆ Participating Entity types - Among Employee ,Ticket Counter and Zoo
(Employee MANAGES Ticket Counter and Employee MANAGES Zoo)
- ◆ Cardinality ratio - 1:1:1
- ◆ Constraints - Employee(0,1) - Ticket Counter(1,1) - Zoo (1,1)

FUNCTIONAL REQUIREMENTS:

Modifications :

- INSERT - New animals,species,zoos,employees etc will be added when ever required.
- DELETE - Deletes data of any employee who leaves (or) deletes data of any animal if it passes away (or) data of a complete zoo if it gets closed by any chance.
- UPDATE - We can update the age of animals,height of them,weight of them,number of visitors visited in a day. We can even update the age and salary of an employee.

Retrieval:

- SELECTION - List of all details of an animal.
- PROJECTION QUERY - Number of visitors visiting a zoo.
- AGGREGATE - Total income of a zoo.
- SEARCH - Employees with salaries above 10k.
- ANALYSIS - Best zoo to visit.
(which have the maximum number of visitors,species,maximum income).

SUMMARY :

The mini world that we created describes the information about the ZOO MANAGEMENT SYSTEM. The Database maintains the records of the zoos. The end users can use the data abstraction provided by the mini world to meet their specific needs.