CS275 – Intro to Databases

Relations and Constraints
Chap. 3.1 – 3.4
Roadmap for Today

• Data representation and integrity constraints
• Table creation and modification
• Data creation and modification, and retrieval
Data Representation

• Relational model
  – Database is a collection of relations (tables)
  – Each table consists of rows (data) and columns (properties)
  – Relatively easy to construct complex queries

<table>
<thead>
<tr>
<th>CrewId</th>
<th>Name</th>
<th>Callsign</th>
<th>Rank</th>
<th>IsACylon</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Adama, William</td>
<td>Husker</td>
<td>Admiral</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Thrace, Kara</td>
<td>Starbuck</td>
<td>Captain</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Agathon, Karl</td>
<td>Helo</td>
<td>Captain</td>
<td>No</td>
</tr>
</tbody>
</table>
Data Representation

- A relation
  - Schema
  - Instance
Data Representation

• Relation (Table) schema
  – Name of table
  – Columns of table
    • Domain
    • Possible constraints

| CrewId(int) | Name(str) | Callsign(str) | Rank(str) | IsACylon(boolean) |
Data Representation

• Relation instance
  – Set of tuples or rows
    • No duplicate records
    • Order is not important
    • Order of columns generally does not matter
    • Domain constraints limit possible values

<table>
<thead>
<tr>
<th>CrewId</th>
<th>Name</th>
<th>Callsign</th>
<th>Rank</th>
<th>IsACylon</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Adama, William</td>
<td>Husker</td>
<td>Admiral</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Thrace, Kara</td>
<td>Starbuck</td>
<td>Captain</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Agathon, Karl</td>
<td>Helo</td>
<td>Captain</td>
<td>No</td>
</tr>
</tbody>
</table>
Data Representation

• Stats about Tables
  – Degree
    • Number of fields or columns
  – Cardinality
    • Number of tuples
A Quick Review

• What do we remember?
  – Relation
  – Relation schema
  – Relation instance
  – Tuples
  – Fields
  – Domain constraints
  – Cardinality
  – Degree
Real Queries!

- CREATE TABLE Galactica_Crew(
  cid int,
  name varchar(255),
  callsign varchar(255),
  rank varchar(127),
  isacylon boolean
)
CREATE TABLE Galactica_Crew(
  cid int,
  name varchar(255),
  callsign varchar(255),
  rank varchar(127),
  isacylon boolean
)

INSERT INTO Galactica_Crew(cid, name, callsign, rank, isacylon)
VALUES(4, "Adama, Lee", "Apollo", "Major", FALSE)

UPDATE Galactica_Crew GC
SET GC.rank = "Retired"
WHERE GC.name == "Lee, Adama"
Integrity Constraints

• An Integrity Constraint (IC)
  – Condition specified on the schema
  – Restricts data that can be stored
  – Enforced by the DBMS
Integrity Constraints

• Key constraints
  – Unique identifier

• Candidate keys
  – Set of attributes that uniquely determine row
  – No subset of attributes determines the row
  – Every table has one
Integrity Constraints

- CREATE TABLE Galactica_Crew(
  cid int,
  name varchar(255),
  callsign varchar(255),
  rank varchar(127),
  isacylon boolean,
  UNIQUE (name, callsign),
  PRIMARY_KEY(cid)
)
Integrity Constraints

• Foreign key
  – Why go to the trouble of marking unique columns and primary keys?
  – We can use them as foreign keys
  – Lets us cascade changes
Integrity Constraints

• CREATE TABLE Duty_Assignments(
    aid int,
    cid int,
    duty varchar(255),
    FOREIGN KEY(cid ) REFERENCES Galactica_Crew,
    PRIMARY_KEY(aid)
)
General Constraints

• Business rules that are not common across databases
  – Only pilots who have flown > 100 missions can be commanders.
How Can We Violate ICs

• Insertion
  – Domain
  – Unique
  – Primary key
  – Foreign key
How Can We Violate ICs

• Update
  – Domain
  – Unique
  – Primary key
  – Foreign key
How Can We Violate ICs

• Deletion
  – Foreign key
Foreign Keys

- CREATE TABLE Duty_Assignments(
  aid int,
  cid int,
  duty varchar(255),
  FOREIGN KEY(cid ) REFERENCES Galactica_Crew,
  PRIMARY_KEY(aid)
)
Foreign Keys

• What happens when we delete
  – NO ACTION
    • Reject the deletion
  – CASCADE
    • If we delete a crew member, we delete their duty assignments
  – SET NULL
    • If we delete a crew member we set their duty assignment references to NULL
  – SET DEFAULT
    • If we delete a crew member we set their duty assignment references to some default value
Foreign Keys

• Update
  – Same as delete
Selection

- SELECT * FROM Galactica_Crew
- SELECT GC.name FROM Galactica_Crew GC WHERE GC.rank == “Admrial”
Questions?

• ?