BEGINNING T-SQL

Jen McCown

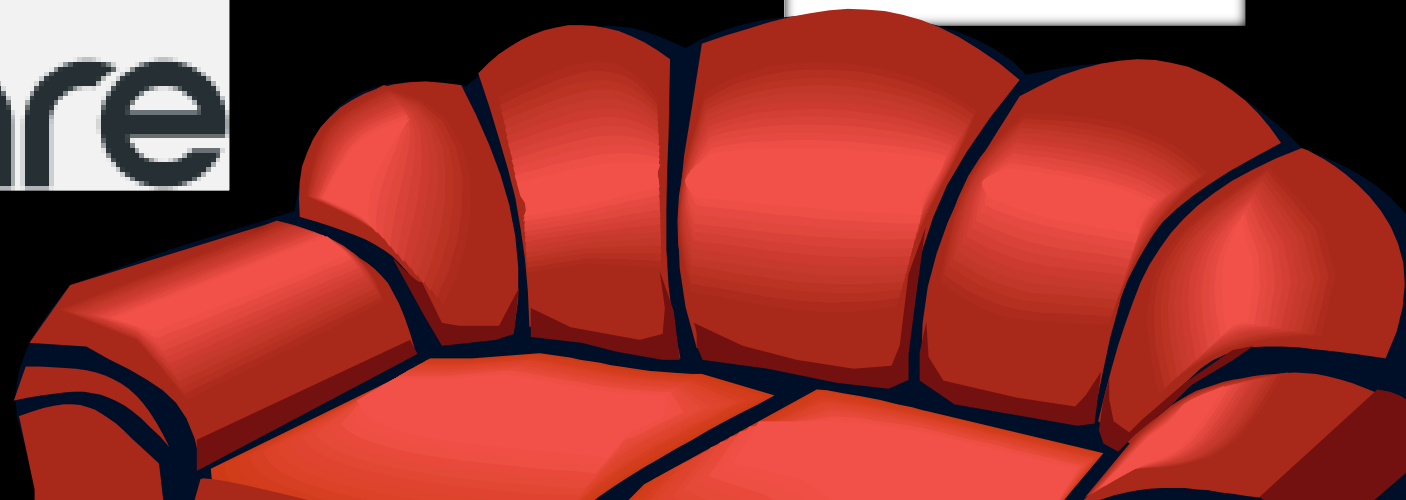
MidnightSQL Consulting, LLC
MinionWare, LLC

MIDNIGHT SQL

MidnightDBA
by
ITBoo

Microsoft®
CERTIFIED
Master

minionware



FIRST: GET READY

1. What to model?
2. What is T-SQL?
3. Books Online (BOL)
4. Transactions

WHAT TO MODEL?

- **What kind of data should we store in OUR database?**

Past suggestions include: goats, socks, plants, whiskey, candy, mafia hits, insults, toothpaste flavor ratings, hair counts, hubcap styles, Dr. Who factoids, book sales…

WHAT IS T-SQL?

- **SQL**
  - Special purpose programming language for database management system.
  - ANSI standard language.
- **Transact-SQL (T-SQL)**
  - Same thing, mostly.
  - *Proprietary* language for Microsoft SQL Server.

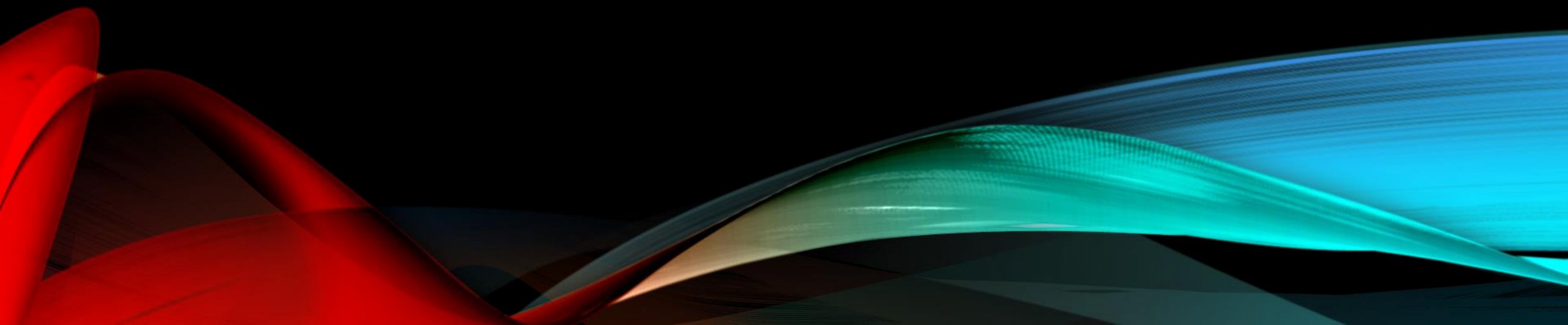
BOOKS ONLINE (BOL)

- **The official word.**
- **The reference.**
- **For realz.**
- **At Technet.Microsoft.com**

TRANSACTIONS

- **Transaction = a unit of work.** A transaction must completely, or it swims. No partial complete or rollback.
- **Any single T-SQL statement** runs in an *implicit transaction*.
- **You can explicitly define** *explicit transactions*. One use: grouping multiple statements into a single transaction.

DEMO: TRANSACTIONS



NEXT: START A TABLE

5. Tables
    - Demo: CREATE TABLE
    - Demo: What do we want to know?
6. Data types
    - Demo: Add data types
    - Demo: Char vs. Varchar
    - Demo: Add data types

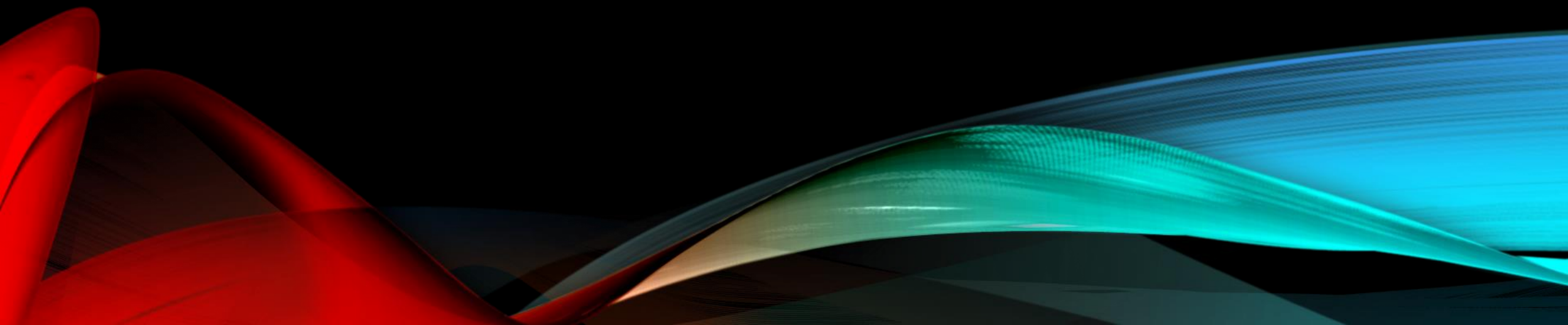
TABLES

- Has a name
- Stores data, rather like a spreadsheet: columns and rows
- Each column has a name and data type
- Each row is unique

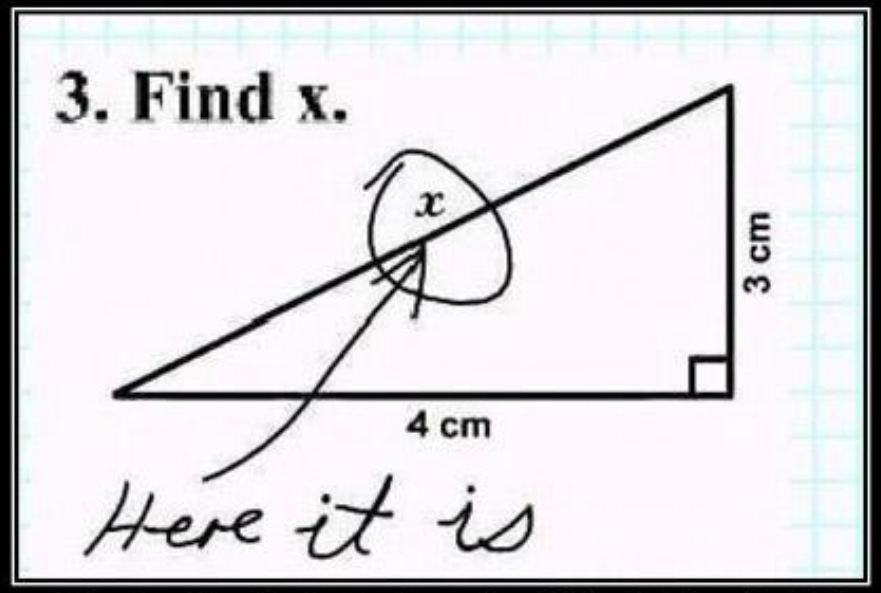
RoomID	BuildingID	RoomName	SizeSqFt
1	1	Living Room	200
2	1	MBR	200
3	1	bath1	200
4	1	bath2	200
5	1	kid room 1	200
6	1	kid room 2	200
7	1	kitchen	200
8	1	laundry ro...	200

DEMO: CREATE TABLE

DEMO: WHAT DO WE WANT TO KNOW?



DATA TYPES


3. Find x.
x
3 cm
4 cm
Here it is
SIMPLICITY
The simplest solutions are often the cleverest
They are also usually wrong

DATA TYPES

- Data type determines what you can do with the data.
- Use *appropriate* data types.

Value	Type of Data	SQL Data Type
"Susan"	String	Varchar
197	Integer	Int
March 20, 1983	Date	Date

DATA TYPES: NUMERIC

SQL Datatype	Description	Example
tinyint, smallint, int, bigint	Integer data	0, -180, 1250467
decimal, numeric	Fixed precision numbers	0.0004, 1.0, 3028.17
float, real	Non fixed precision number (like a variable decimal)	0.0004, 1.0, 3028.17
smallmoney, money	Currency data	1604.38, 0, -12.11

DATA TYPES: DATE

SQL Datatype	Description	Example
smalldatetime, datetime	Date and time data	2010-03-20 11:20:03.107, 2010-04-07 00:00:00
date	Date data	2010-03-20
time	Time data	11:20:03.1071974
datetime2	Date and time data (expanded datetime)	2010-03-20 11:20:03.1071974
datetimeoffset		

DATA TYPES: STRING

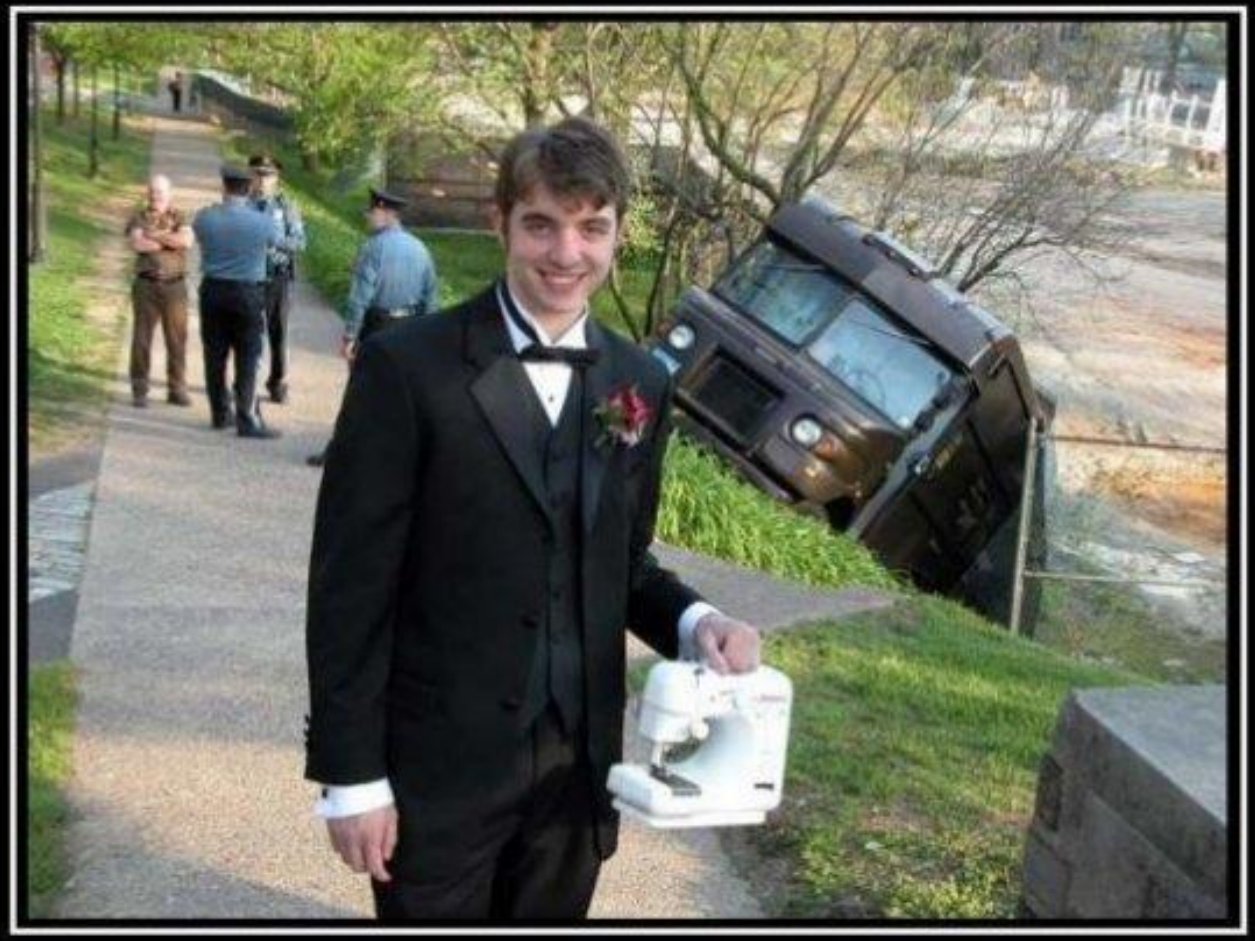
SQL Datatype	Description	Example
char	Fixed length character data	“Sue ”, “Jennifer”
varchar	Variable length character data	“Sue”, “Jennifer”
nchar	Fixed length Unicode character data	“Sue ”, “Jennifer”
nvarchar	Variable length Unicode character data	“Sue”, “Jennifer”

DEMO: ADD DATA TYPES

DEMO: CHAR VS VARCHAR

NEXT: FINISH A TABLE

7. NULLs
    - Demo: NULLs
    - Demo: Add NULL / NOT NULL
    - Demo: Create the table!



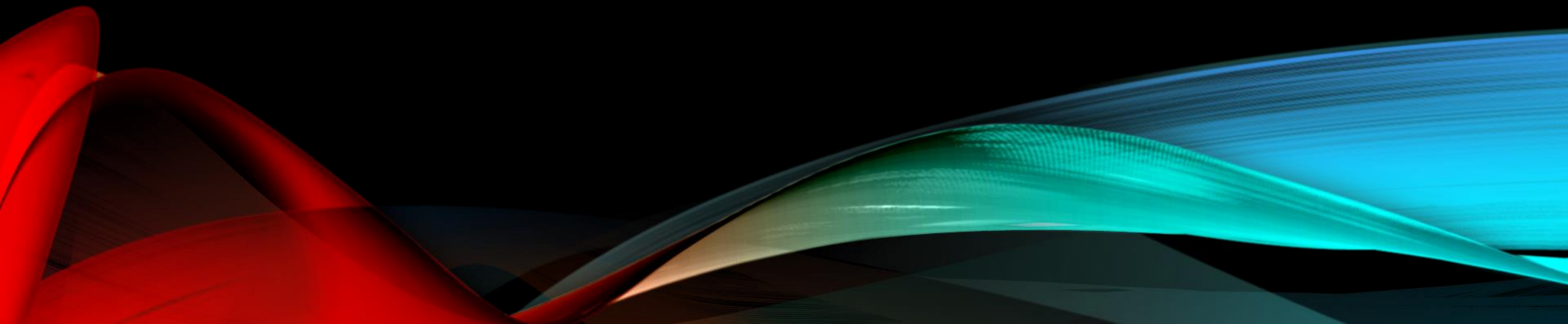
UNEXPLAINABLE

This picture can not be explained.

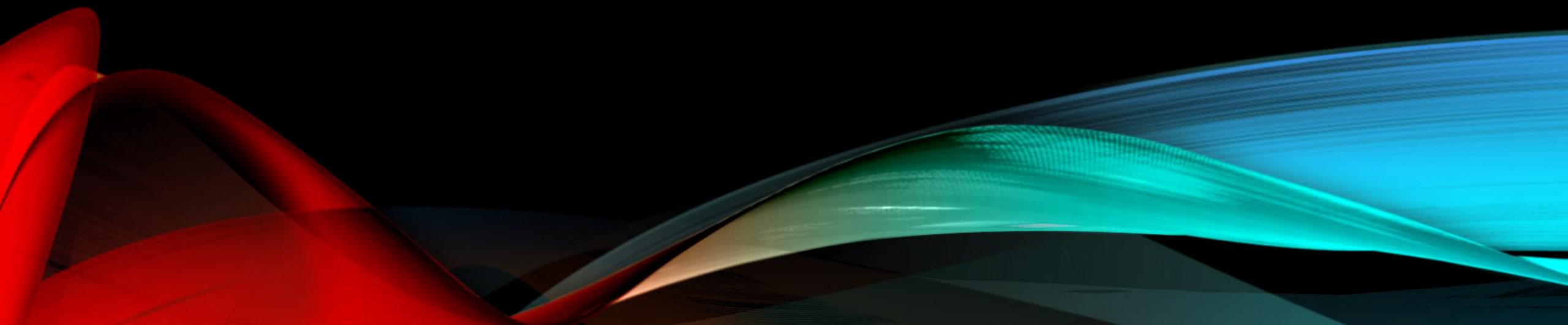
NULL: NO KNOWN VALUE

- NULL means “unknown value”
- Nothing can be = NULL
- Nothing can be <> NULL
- We can only test to see if the field IS NULL or IS NOT NULL

DEMO: NULLS



DEMO: CREATE THE TABLE



STATEMENTS!

- SELECT
- INSERT
- UPDATE
- DELETE

SELECT

- SELECT *columns* FROM TableName
  - SELECT retrieves data.
- SELECT * FROM TableName
  - Retrieves all columns. But don't use it!
- WHERE
  - The WHERE clause limits what rows we get back.
  - WHERE value = 1, <> 1
  - WHERE value IN ()
  - WHERE value LIKE 'string%'

INSERT

- Inserts a row into a table
- Use a column list!
- You can insert multiple rows with one statement.

UPDATE…WHERE

- UPDATE modifies data already in the table
- Example:

```
UPDATE Table1
SET isHungry = 0
WHERE myName = ‘Jen’
```

DELETE

- Delete data from the table
- DELETE … WHERE

CONSTRAINTS

CAPTAIN JAMES T. KIRK

I'M SORRY, I CAN'T HEAR YOU OVER THE SOUND OF HOW AWESOME I AM.

PRIMARY KEY (PK) CONSTRAINT

- A *constraint* is a rule for the table.
- PK is the column(s) that uniquely identifies a row
- 1 PK per table
- PK doesn't allow NULL
- Example:

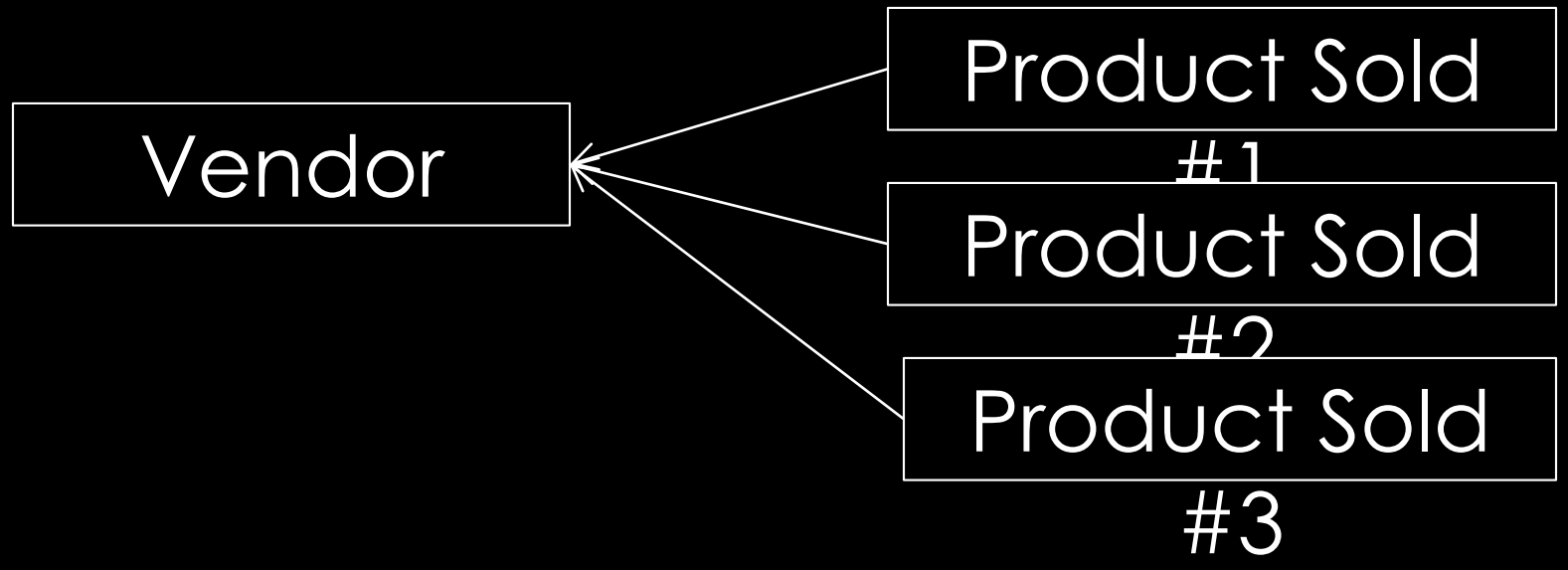
  ALTER TABLE *tableName*
  ADD CONSTRAINT *constrName*
  PRIMARY KEY (*columnNames)*

PRIMARY KEY (PK) CONSTRAINT

- Natural vs. surrogate PKs
- IDENTITY()
- GUIDs

FOREIGN KEY (FK) CONSTRAINT

- An FK links two tables
- Prevents orphaned records
- Link FK on *ChildTable* to PK on *ParentTable*



JOIN

- INNER JOIN
- OUTER JOIN: Full, Left, Right
- CROSS JOIN (rarely used)

BONUS!

BONUS!

- GetDate()
- Varchar vs Nvarchar
  - Varchar can't handle Unicode. For example, this won't work:
    DECLARE @reg VARCHAR(10) = ' θЂب';
    SELEC @reg;
  - Nvarchar can handle Unicode. For example, this WILL work:
    DECLARE @uni NVARCHAR(10) = N' θЂب';
    SELEC @uni;
- Sp_help
- Sys.sql_modules
- Indexes!

Tutorials: www.MidnightDBA.com
Consulting: www.MidnightSQL.com
Software: www.MinionWare.net