DENORMALIZATION

To Normalize or Not To Normalize…

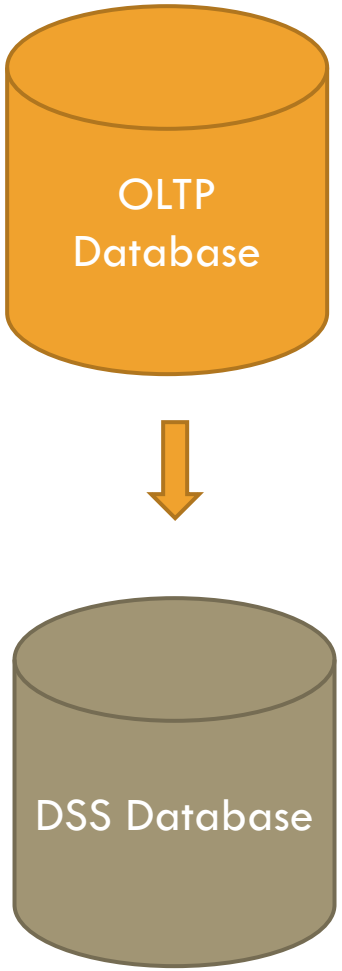
- Advantages of Normalization
  - Eliminate modification anomalies
  - Reduce duplicate data
- Disadvantages
  - More complicated SQL required to query a normalized database
  - Complicated queries can reduce DBMS performance in some cases

When not to Normalize

- Scenario 1: Database is read only
  - No modification anomalies can occur
- Scenario 2: Selected data never changes
  - Example: Zip codes

Two Types of Databases

- Online Transaction Processing (OLTP)
  - Contain operational data
  - Supports business applications
  - Updated live by applications
- Decision Support System (DSS)
  - Populated periodically from OLTP
  - Supports reporting and business intelligence apps
  - Read-only data



To Normalize or Not

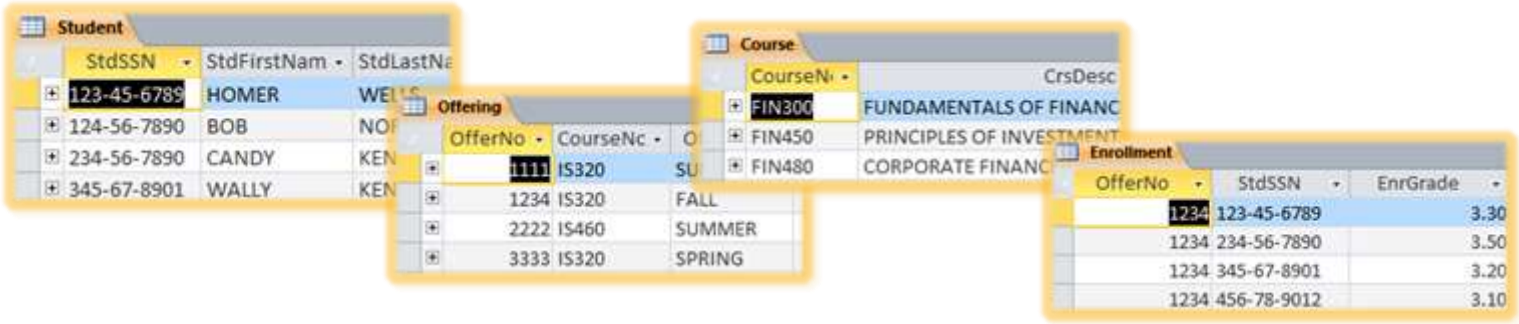
- Normalized databases contain minimal data redundancy
- Tradeoff: Normalization leads to
  - More tables
  - More complexity
  - Lower query performance
- Normalization is vital for OLTP databases
  - Data redundancy in transactional databases is a disaster waiting to happen
- What about DSS databases?

Decision Support Databases

- DSS databases are designed to
  - Optimize query performance
  - Simplify query design
- How it works:
  - DSS populated nightly with data extracted from OLTP databases
  - DSS tables result from merging several operational tables using a join query
    - This produces a denormalized design
  - End users query DSS databases; do not update
  - Modification anomalies cannot occur

DSS Example

- Operational Database:



Student

StdSSN	StdFirstNam	StdLastNa
123-45-6789	HOMER	WEL
124-56-7890	BOB	NO
234-56-7890	CANDY	KEN
345-67-8901	WALLY	KEN

Offering

OfferNo	CourseNc	O
1111	IS320	SU
1234	IS320	FALL
2222	IS460	SUMMER
3333	IS320	SPRING

Course

CourseN	CrsDesc
FIN300	FUNDAMENTALS OF FINANC
FIN450	PRINCIPLES OF INVESTMENT
FIN480	CORPORATE FINANC

Enrollment

OfferNo	StdSSN	EnrGrade
1234	123-45-6789	3.30
1234	234-56-7890	3.50
1234	345-67-8901	3.20
1234	456-78-9012	3.10

- DSS Database:

StdSSN	StdFirstNam	StdLastNam	OfferNo	EnrGrade	CourseN	
123-45-6789	HOMER	WELLS	1234	3.30	IS320	FUNDAME
234-56-7890	CANDY	KENDALL	1234	3.50	IS320	FUNDAME
345-67-8901	WALLY	KENDALL	1234	3.20	IS320	FUNDAME
456-78-9012	JOE	ESTRADA	1234	3.10	IS320	FUNDAME
567-89-0123	MARIAH	DODGE	1234	3.80	IS320	FUNDAME
678-90-1234	TESS	DODGE	1234	3.40	IS320	FUNDAME

Views and Normalization

- Normalization introduces database design complexity
- Create views to tame the complexity and present a simple ("denormalized") view of data to query designers
- Views address the complexity imposed by normalization, but not the performance issues

Creating a View

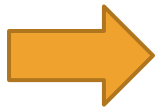
```
CREATE VIEW CustomerOrders AS
    SELECT OrderID, OrderDate, CustName
    FROM Orders INNER JOIN Customers
            ON Orders.CustID = Customers.CustID
```

Orders

OrderID	OrderDate	CustID
1	2017-02-03	1
2	2017-03-04	2
3	2017-03-08	2

Customers

CustID	CustName
1	Fred Jones
2	Freda Anderson
3	Amy Harrison



CustomerOrders

OrderID	OrderDate	CustName
1	2017-02-03	Fred Jones
2	2017-03-04	Freda Anderson
3	2017-03-08	Freda Anderson