The importance of data normalization

CREATING POSTGRESQL DATABASES

SQL

Darryl Reeves

Industry Assistant Professor, New York University

Example 1: redundant data

- Data redundancy can be problematic

```
CREATE TABLE loan (
    borrower_id INTEGER REFERENCES borrower(id),
    bank_name VARCHAR(50) DEFAULT NULL,
    ...
);
```

```
CREATE TABLE bank (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) DEFAULT NULL,
    ...
);
```

Example 1: redundant data

```
CREATE TABLE loan (
    borrower_id INTEGER REFERENCES borrower(id),
    bank_name VARCHAR(50) DEFAULT NULL,
    ...
);
```

```
CREATE TABLE bank (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) DEFAULT NULL,
    ...
);
```

- Problem 1: Different banks/same name
- Problem 2: Name changes

Example 1: redundant data

```
CREATE TABLE loan (
    borrower_id INTEGER REFERENCES borrower(id),
    bank_id INTEGER REFERENCES bank(id),
    ...
);
```

- Banks share name with distinct ids
- Updates to bank names will only affect bank table

Example 2: consolidating records

`applicant`

id	name
1	Jane Simmmons
2	Rick Demps
3	Pam Jones

`borrower`

id	name
1	Jack Smith
2	Sara Williams
3	Jennifer Valdez

Example 2: consolidating records

`applicant`

id	name
1	Jane Simmmons
2	Rick Demps
3	Pam Jones

`borrower`

id	name
1	Jack Smith
2	Sara Williams
3	Jennifer Valdez
4	Pam Jones

Example 2: consolidating records

`applicant`

id	name
1	Jane Simmmons
2	Rick Demps
3	Pam Jones

`borrower`

id	name
1	Jack Smith
2	Sara Williams
3	Jennifer Valdez

Example 2: consolidating records

`applicant`

id	name
1	Jane Simmmons
2	Rick Demps

`borrower`

id	name
1	Jack Smith
2	Sara Williams
3	Jennifer Valdez
4	Pam Jones

Example 2: consolidating records

```
CREATE TABLE borrower (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL
);
```

Example 2: consolidating records

```
CREATE TABLE borrower (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    approved BOOLEAN DEFAULT NULL
);
```

- `approved` is `NULL` => applicant
- `approved` is `true` => borrower
- `approved` is `false` => denied application

Why normalize data?

- Reduces data duplication
- Increases data consistency
- Improves data organization

Let's practice!

1st Normal Form

CREATING POSTGRESQL DATABASES

SQL

Darryl Reeves

Industry Assistant Professor, New York University

Example: maintaining student records

```
CREATE TABLE student (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    courses VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

- Update errors
- Insertion errors
- Deletion errors

Example: duplicated data after update

id	name	courses	home_room
122	Susan Roth	Algebra I, Physics, Spanish II	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: duplicated data after update

id	name	courses	home_room
122	Susan Roth	Algebra I, Chemistry, Spanish II	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: duplicated data after update

id	name	courses	home_room
122	Susan Roth	Algebra I, Chemistry, Spanish II, Chemistry	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: insertions with column restrictions

```
CREATE TABLE student (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    courses VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

id	name	courses	home_room
122	Susan Roth	Algebra I, Physics, Spanish II	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: insertions with column restrictions

```
CREATE TABLE student (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    courses VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

id	name	courses	home_room
122	Susan Roth	Algebra I, Physics, Spanish II	101
413	Robert Cruz	History, Geometry, Biology, French Literature	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: data integrity impacted by deleting records

id	name	courses	home_room
122	Susan Roth	Algebra I, Physics, Spanish II	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	English III, Chemistry, Algebra II	102

Example: data integrity impacted by deleting records

id	name	courses	home_room
122	Susan Roth	Algebra I, Physics, Spanish II	101
413	Robert Cruz	History, Geometry, Biology	204
613	Thomas Wright	???	102

Satisfying 1st Normal Form (1NF)

- 1NF Requirement:
  - Table values must be atomic

Example: student table satisfying 1NF

```
CREATE TABLE student (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    courses VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

Example: student table satisfying 1NF

```
CREATE TABLE student (
    id INTEGER,
    name VARCHAR(50) NOT NULL,
    courses VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

Example: student table satisfying 1NF

```
CREATE TABLE student (
    id INTEGER,
    name VARCHAR(50) NOT NULL,
    course VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

Example: student table satisfying 1NF

id	name	course	home_room
122	Susan Roth	Algebra I	101
122	Susan Roth	Physics	101
122	Susan Roth	Spanish II	101
413	Robert Cruz	History	204
413	Robert Cruz	Geometry	204
413	Robert Cruz	Biology	204

Example: student table satisfying 1NF

```
CREATE TABLE student (
    id INTEGER,
    name VARCHAR(50) NOT NULL,
    course VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

Example: student table satisfying 1NF

```
CREATE TABLE student (
    student_id INTEGER,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    course VARCHAR(50) NOT NULL,
    home_room SMALLINT NOT NULL
);
```

Example: student table satisfying 1NF

id	first_name	last_name	course	home_room
122	Susan	Roth	Algebra I	101
122	Susan	Roth	Physics	101
122	Susan	Roth	Spanish II	101
413	Robert	Cruz	History	204
413	Robert	Cruz	Geometry	204
413	Robert	Cruz	Biology	204

Let's practice!

CREATING POSTGRESQL DATABASES

2nd Normal Form

CREATING POSTGRESQL DATABASES

SQL

Darryl Reeves

Industry Assistant Professor, New York University

Example: school textbooks

```
CREATE TABLE textbook (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    publisher_name VARCHAR(100) NOT NULL,
    publisher_site VARCHAR(50),
    quantity SMALLINT NOT NULL DEFAULT 0
);
```

Example: school textbooks

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22

Example: inconsistency from updating url

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22

Example: inconsistency from updating url

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.newabc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22

Example: adding publisher without textbook

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22

Example: adding publisher without textbook

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22
??	??	New Horizons	www.nhorizon.com	??

Example: removing a textbook

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27
112	Statistical Concepts	Martin House	www.mh.com	22

Example: removing a textbook

id	title	publisher_name	publisher_site	quantity
23	Introductory Algebra: 1st Edition	ABC Publishing	www.abc.com	32
74	Calculus Foundations	ABC Publishing	www.abc.com	27

- Publisher requires separate table
- Data anomalies from insertions and deletions

Satisfying 2nd Normal Form (2NF)

- 1NF is satisfied
- All non-key columns are dependent on the table's `PRIMARY KEY`

Example: textbooks and publishers in 2NF

```
CREATE TABLE textbook (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    publisher_name VARCHAR(100) NOT NULL,
    publisher_site VARCHAR(50),
    quantity SMALLINT NOT NULL DEFAULT 0
);
```

Example: textbooks and publishers in 2NF

```
CREATE TABLE textbook (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    quantity SMALLINT NOT NULL DEFAULT 0,
);
```

```
CREATE TABLE publisher (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    site VARCHAR(50)
);
```

Example: textbooks and publishers in 2NF

```
CREATE TABLE textbook (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    quantity SMALLINT NOT NULL DEFAULT 0,
    publisher_id INTEGER REFERENCES publisher(id)
);
```

```
CREATE TABLE publisher (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    site VARCHAR(50)
);
```

Let's practice!

CREATING POSTGRESQL DATABASES

3rd Normal Form

CREATING POSTGRESQL DATABASES

SQL

Darryl Reeves

Industry Assistant Professor, New York University

Defining 3rd Normal Form

Requirements

- 2NF is satisfied
- No "transitive dependencies" exist
  - i.e., All non-key columns are only dependent on the `PRIMARY KEY`

Transitive dependencies

- Involve 3 columns in table
- Columns X, Y, Z
- column X -> column Y
- column Y -> column Z
- column X -> column Z

Example: course room assignments

id	name	teacher	num
157	Algebra	Maggie Winters	244
162	Physics	Maggie Winters	244
321	Spanish I	Jeremy Smith	309
497	History I	Sarah Williams	313
613	Spanish II	Jeremy Smith	309

- course name -> teacher
- teacher -> room number
- course name -> room number

Example: course room assignments

id	name	teacher	num
157	Algebra	Maggie Winters	244
162	Physics	Maggie Winters	244
321	Spanish I	Jeremy Smith	309
497	History I	Sarah Williams	313
613	Spanish II	Jeremy Smith	309

- course name -> teacher
- teacher -> room number
- course name -> room number
  (transitive dependency)

Example: course room assignments

id	name	teacher	num
157	Algebra	Maggie Winters	244
162	Physics	Maggie Winters	244
321	Spanish I	Jeremy Smith	309
497	History I	Sarah Williams	313
613	Spanish II	Jeremy Smith	309

1. Updating room number

Example: course room assignments

id	name	teacher	num
157	Algebra	Maggie Winters	244
162	Physics	Maggie Winters	244
321	Spanish I	Jeremy Smith	309
497	History I	Sarah Williams	313
613	Spanish II	Jeremy Smith	309

1. Updating room number
2. Adding new teachers

Example: course room assignments

id	name	teacher	num
157	Algebra	Maggie Winters	244
162	Physics	Maggie Winters	244
321	Spanish I	Jeremy Smith	309
497	History I	Sarah Williams	313
613	Spanish II	Jeremy Smith	309

1. Updating room number
2. Adding new teachers
3. Deleting all courses for a teacher

Example: course room assignments

How do we change the structure of our data in order to alleviate these potential problems?

Example: course room assignments

`teacher` table

id	name	room_num
1	Maggie Winters	244
2	Jeremy Smith	309
3	Sarah Williams	313

Example: course room assignments

`teacher` table

id	name	room_num
1	Maggie Winters	244
2	Jeremy Smith	309
3	Sarah Williams	313

`course_assignment` table

id	name	teacher_id
157	Algebra	1
162	Physics	1
321	Spanish I	2
497	History I	3
613	Spanish II	2

Let's practice!

CREATING POSTGRESQL DATABASES