Integrity or correctness of data

• Would like data to be “accurate” or “correct” at all times

EMP

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>White</td>
<td>52</td>
</tr>
<tr>
<td>Green</td>
<td>3421</td>
</tr>
<tr>
<td>Gray</td>
<td>1</td>
</tr>
</tbody>
</table>
Integrity or consistency constraints

- Predicates data must satisfy
- Examples:
  - x is key of relation R
  - x → y holds in R
  - Domain(x) = {Red, Blue, Green}
  - a is valid index for attribute x of R
  - no employee should make more than twice the average salary

Constraints (as we use here) may not capture “full correctness”

Example 1  Transaction constraints
- When salary is updated,
  new salary > old salary
- When account record is deleted,
  balance = 0
Constraints (as we use here) may not capture “full correctness”

Example 2  Database should reflect real world

Transaction: collection of actions that preserve consistency
Big assumption:

If T starts with consistent state + T executes in isolation
⇒ T leaves consistent state

Correctness in the case of failure (informally)

- If we stop running transactions, DB left consistent
- Each transaction sees a consistent DB
How can constraints be violated?

• Transaction bug
• DBMS bug
• Hardware failure
  – e.g., disk crash alters balance of account
• Data sharing
  – e.g.: T1: give 10% raise to programmers
  T2: change programmers ⇒ systems analysts

Will not consider:

• How to write correct transactions
• How to write correct DBMS
• Constraint checking & repair
  – That is, solutions studied here do not need to know constraints