1. **Data Quality (Exam exercise)**

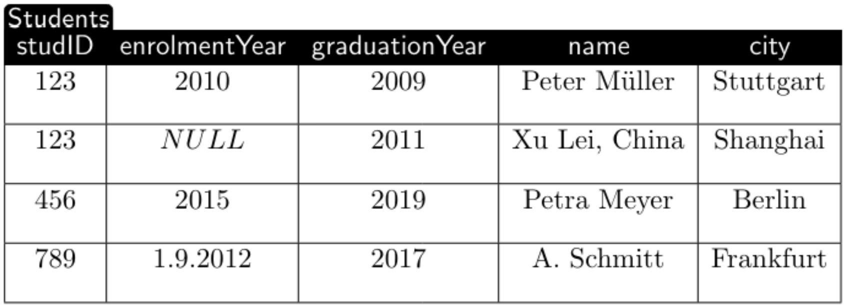
(a) In class, we discussed in detail a semantic-oriented classification of quality criteria, namely the requirement survey. Which of the following information quality criteria does not fit the category of technical criteria? Provide a brief description of the selected IQ criterium.

- Latency
- Completeness
- Timeliness
- Security

Completeness does not fit the category of technical criteria.

**Latency:** is the amount of time in seconds from issuing the query until the first data item reaches the user
**Completeness:** is the extent to which data is not missing and is of sufficient breadth, depth, and scope for the task at hand.
**Timeliness:** is the extent to which the age of the data is appropriate for the task at hand
**Security:** is the extent to which access to data is restricted appropriately to maintain its security

(b) The following table presents multiple data errors that can arise within one data source. Identify the errors by circling them and assigning them a unique number. Then, use the provided space to provide, for each numbered error, the type of error (schema level or entity level) and its name or description based on the data error classification we discussed in class.



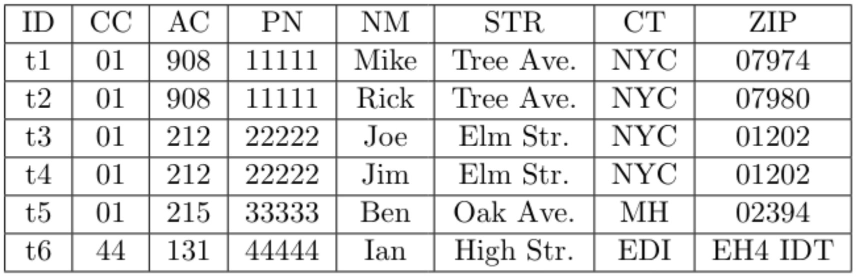
Students

| studID | enrolmentYear | graduationYear | name | city |
|---|---|---|---|---|
| 123 | 2010 | 2009 | Peter Müller | Stuttgart |
| 123 | *NULL* | 2011 | Xu Lei, China | Shanghai |
| 456 | 2015 | 2019 | Petra Meyer | Berlin |
| 789 | 1.9.2012 | 2017 | A. Schmitt | Frankfurt |

- **studID:** duplicate values (Schema Level: uniqueness violation)
- **enrolmentYear:** NULL Value (Entity Level: missing values), Wrong date format (Schema Level: invalid value)
- **graduationYear:** Can not graduate before enrolment (Schema Level: invalid value)
- **name:** Interrelated values (Entity Level: embedded value), Wrong name format (Schema Level: invalid value)

## 2. Conditional Functional Dependencies

In addition to the traditional functional dependencies (FDs), there are also conditional functional dependencies (CFDs). Based on the short discussion in the lecture, this task will cover CFDs in detail. We will use the following schema, which specifies customers using attributes for country code (CC), area code (AC), phone number (PN), name (NM), street (STR), city (CT), and zip code (ZIP). (For details about CFDs see Bohannon, P., Fan, W., Geerts, F., Jia, X., Kementsietsidis, A: *Conditional functional dependencies for data cleaning.* ICDE 2007.)



| ID | CC | AC | PN | NM | STR | CT | ZIP |
|---|---|---|---|---|---|---|---|
| t1 | 01 | 908 | 11111 | Mike | Tree Ave. | NYC | 07974 |
| t2 | 01 | 908 | 11111 | Rick | Tree Ave. | NYC | 07980 |
| t3 | 01 | 212 | 22222 | Joe | Elm Str. | NYC | 01202 |
| t4 | 01 | 212 | 22222 | Jim | Elm Str. | NYC | 01202 |
| t5 | 01 | 215 | 33333 | Ben | Oak Ave. | MH | 02394 |
| t6 | 44 | 131 | 44444 | Ian | High Str. | EDI | EH4 IDT |

(a) Give a definition of CFDs. Think of a scenario where CFDs might be more suitable than FDs.

CDFs are an extension of FDs. They describe dependencies that only apply under certain conditions or for certain subsets of data.

Some dependencies must only be checked under some conditions, which is less expensive.

(b) Use the following background knowledge about the data to create CFDs:

For customers in the UK (country code 44), the zip code determines the street.

If the country code is 01 and the area code 908, then if two tuples have the same phone number, they also must have the same street, zip code values, and city.

For all tuples with area code 212, which also have the country code 01 and the same phone number, the the city must be NYC and they must have the same zip code values and the same street.

For all tuples in the US (country code 01) and with area code 215, the city must be PHI.

1. {country code = 44, zip code} -> street

2. {Country Code = 01, Area Code = 908} -> Phone Number -> {street, zip code, city}

3. {Country Code =01, Area Code 212, Phone Number} -> {City = NYC, zip code, street}

4. {Country Code = 01, Area Code 215} -> {City = PHI}

(c) Check the data for tuples that violate the CFDs. What is a special case that does not occur when using FDs?

- t1 and t2 violates 2. They have different ZIPs.
- t5 violates 4. It has a different City than PHI

Answer:

(d) When using a set of CFDs we have to make sure that they are consistent, which means that they should not contradict each other. Make sure that the CFDs you created are consistent.

Our Statements are already consistent.

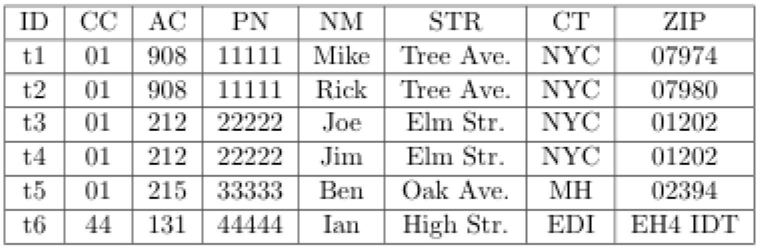
(e) How could you use CFDs in the context of data cleaning?

To find and correct:

- Redundancies
- Anomalies
- Data Errors
- Ensure Integrity

### 3. Data Profiling: Partitioning TODO!!

There are different ways to check a relation for functional dependencies. Using the example data below, this task will focus on partitioning.



| ID | CC | AC | PN | NM | STR | CT | ZIP |
|---|---|---|---|---|---|---|---|
| t1 | 01 | 908 | 11111 | Mike | Tree Ave. | NYC | 07974 |
| t2 | 01 | 908 | 11111 | Rick | Tree Ave. | NYC | 07980 |
| t3 | 01 | 212 | 22222 | Joe | Elm Str. | NYC | 01202 |
| t4 | 01 | 212 | 22222 | Jim | Elm Str. | NYC | 01202 |
| t5 | 01 | 215 | 33333 | Ben | Oak Ave. | MH | 02394 |
| t6 | 44 | 131 | 44444 | Ian | High Str. | EDI | EH4 IDT |

(a) What is partition refinement and why can it be used to detect functional dependencies?

It partitions relational databases with the aim to determine whether certain attributes are independent from each other.

(b) Given our example data compute the stripped partitions. The partitions over more than one attribute should be calculated using the partition product algorithm.

(c) Which functional dependencies can you find using the stripped partitions? Can you also detect a key?

(d) Discuss the complexity of partitioning with basic and stripped partitions.

(e) Based on e(X) can you provide a formula to compute the minimal number of data edits in column (A) such that (A) becomes a key.