ER - DIAGRAM

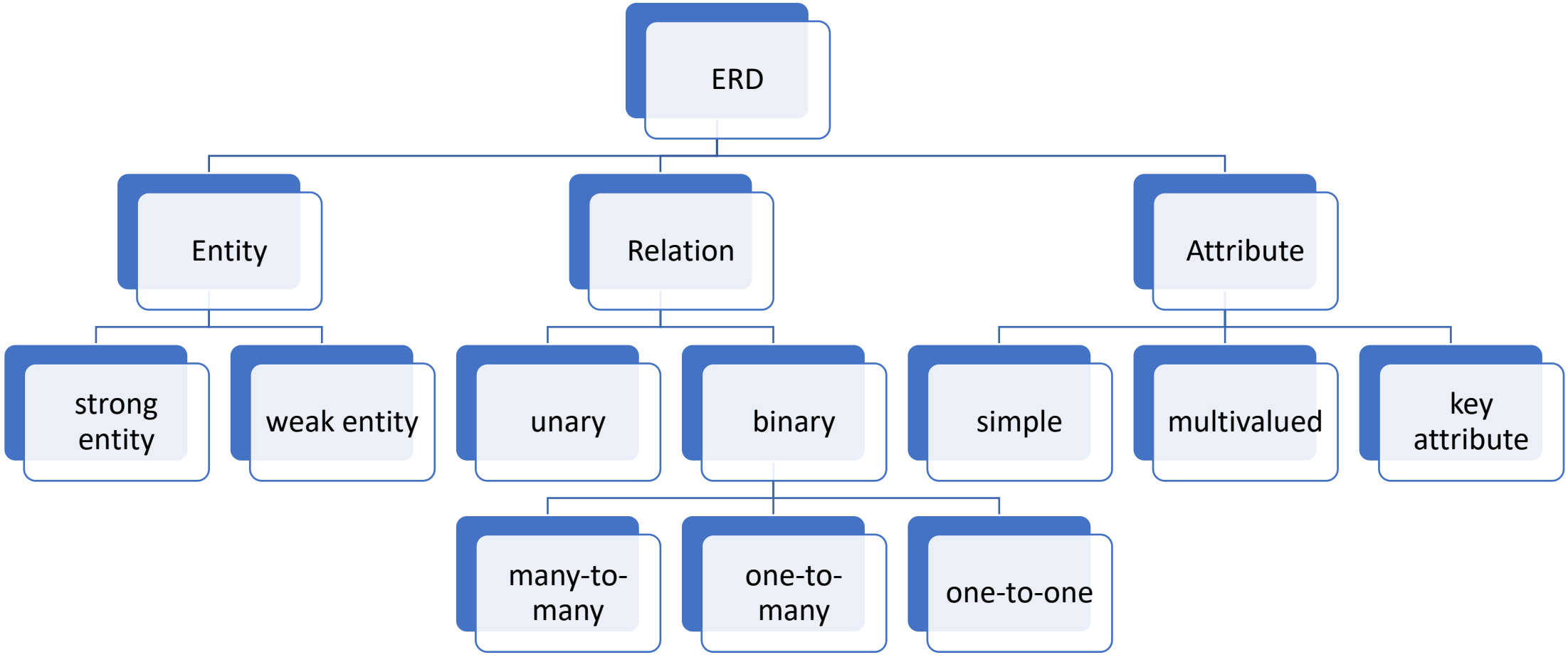
LECTURE 2: Databases

ER - Diagram

- User story/requirement analysis ➔ **ER** ➔relational database schema.
- Easy to translate into relational tables.

- High-level design.
- Suitable for structured systems.

ERD - components



ER - Diagram

ENTITY

person, place, activity, event, concept, real world object etc.

usually a noun

RELATION

ATTRIBUTE

ER - Diagram

ENTITY

person, place, activity, event, concept, real world object etc.

usually a noun

RELATION

links entities (unary, binary, ternary).

usually a verb

ATTRIBUTE

ER - Diagram

ENTITY

person, place, activity, event, concept, real world object etc.

usually a noun

RELATION

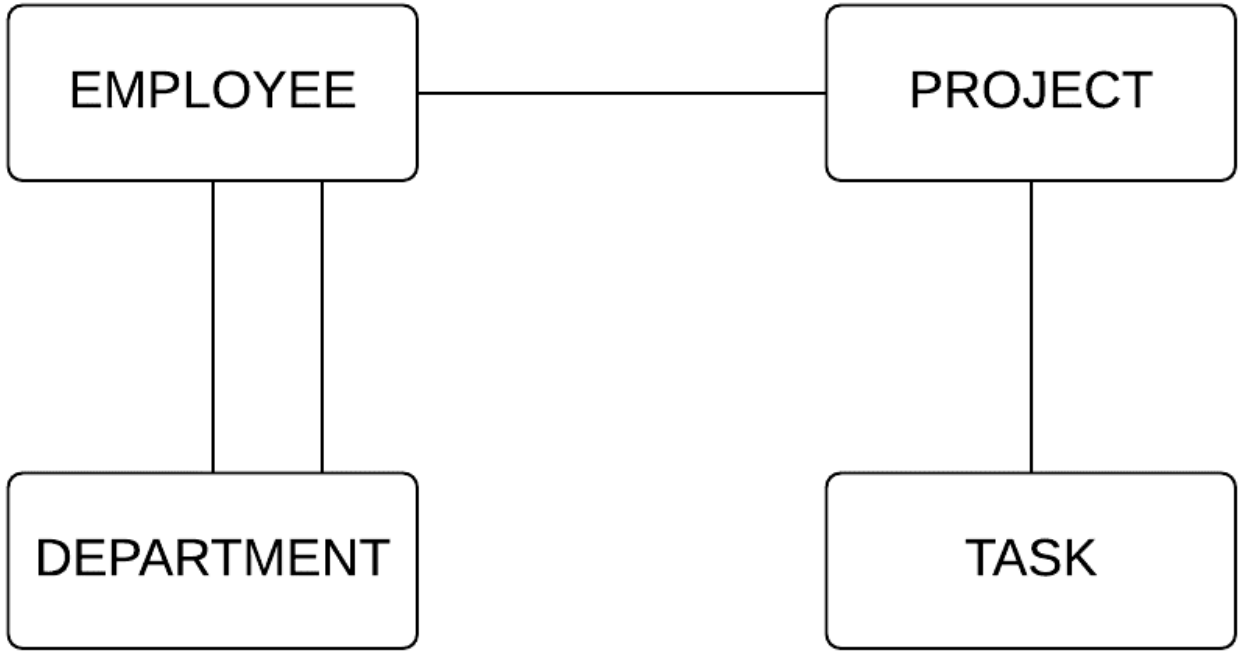
links entities (unary, binary, ternary).

usually a verb

ATTRIBUTE

describe entities or relations

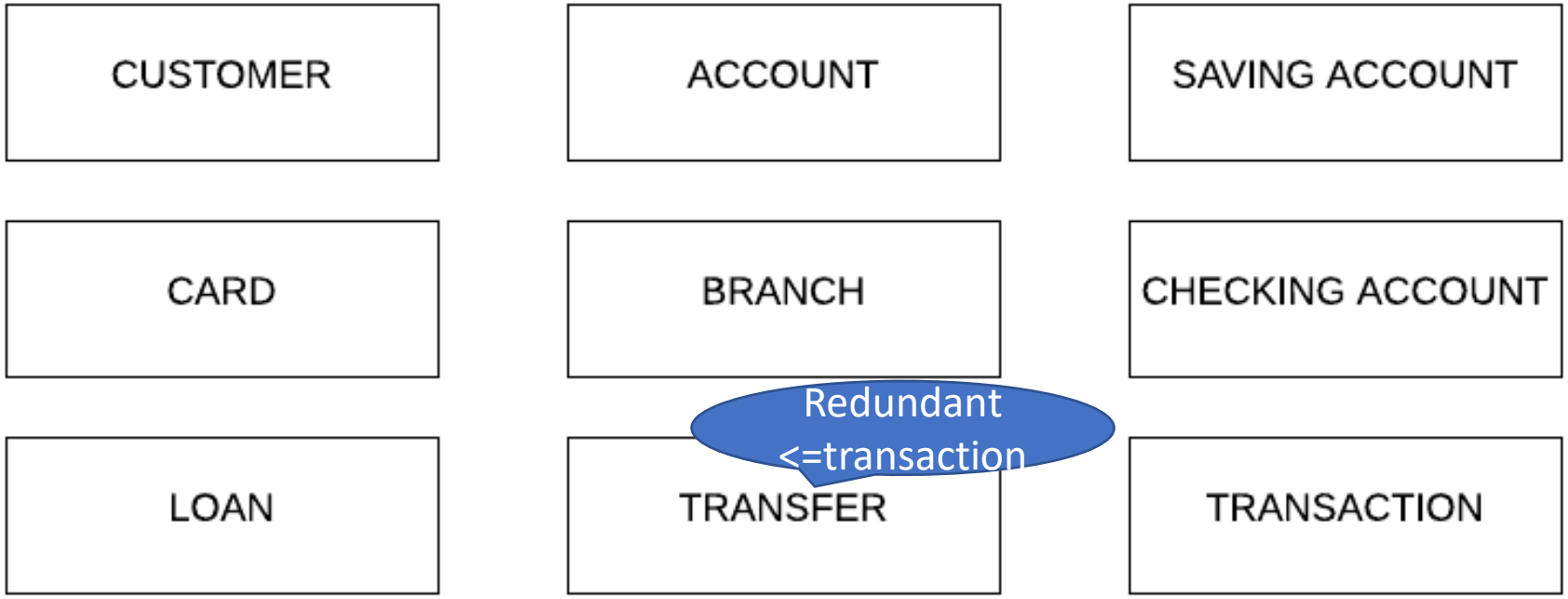
Entities

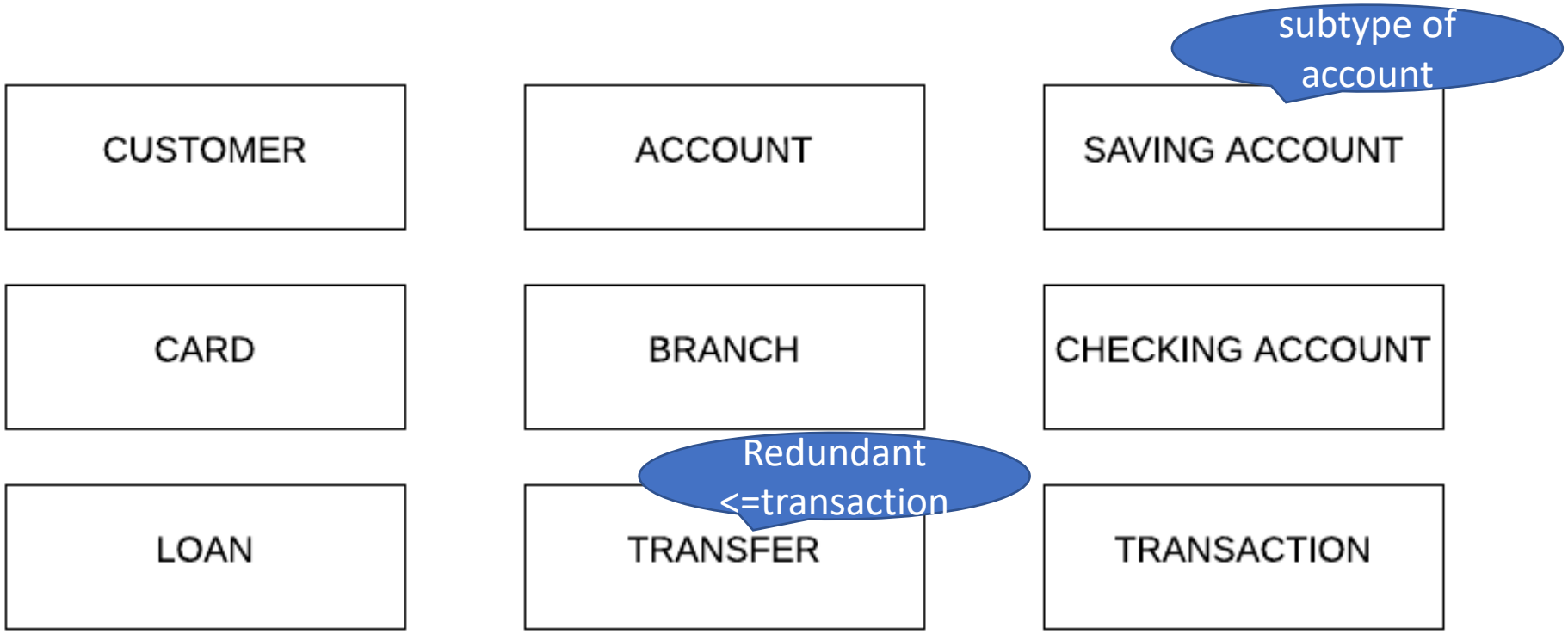


Banking (1)Entities

- A customer opens a saving account or a checking account, at a bank branch. He may also access loans. For each checking account he has a card. Periodically he may withdraw money from his account or partially pay his loans. He may also transfer money from one account to another.

- Please answer www.menti.com 8045 7859 Q5, Q6


CUSTOMER
ACCOUNT
SAVING ACCOUNT
CARD
BRANCH
CHECKING ACCOUNT
Redundant
<=transaction
LOAN
TRANSFER
TRANSACTION


subtype of account
CUSTOMER
ACCOUNT
SAVING ACCOUNT
CARD
BRANCH
CHECKING ACCOUNT
Redundant <=transaction
LOAN
TRANSFER
TRANSACTION

CUSTOMER	ACCOUNT	SAVING ACCOUNT
CARD	BRANCH	CHECKING ACCOUNT
LOAN	TRANSFER	TRANSACTION

!!!not all beneficiaries (missing from story) are customers of the same bank

Entities

- Unique names, uppercase characters
- Graphical representation: rectangles

- Relational database: entity ➔ table (line & columns)
- Primary key: attribute or group of attributes that uniquely identifies entity instances

Candidate key

- *Relation* in relational model <> *relationship* in ERD
- Relation <-> set of tuples <-> tables <-> attributes of relation are columns

- Minimal *superkey* or o *relation* (minimal set of attributes) such that:
  1) There are no two distinct tuples sharing the same values for the *superkey* (unique)
  2) No proper subsets of the superkey has property (1)

- Attributes of the superkey are called prime attributes.
- Attributes that does not occur in any superkey are non-prime attributes.
- If there are no null values, since a relation is a *set* of tuples, each relation has at least one candidate key, i.e the set of all its attributes.

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

There are no candidate key with one attribute

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Unique keys: ~~(BUYER_ID, PRICE)~~

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Unique keys: ~~(BUYER_ID, HALL_NO)~~

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Unique keys: (BUYER_ID, DATE) (BUYER_ID, VIP) (PRICE, DATE) (HALL_NO, DATE) (HALL_NO, TYPE)
(BUYER_ID, PRICE, HALL_NO) (BUYER_ID, HALL_NO, DATE)
etc -- all sets with 3, 4 or 5 attributes

Candidate keys

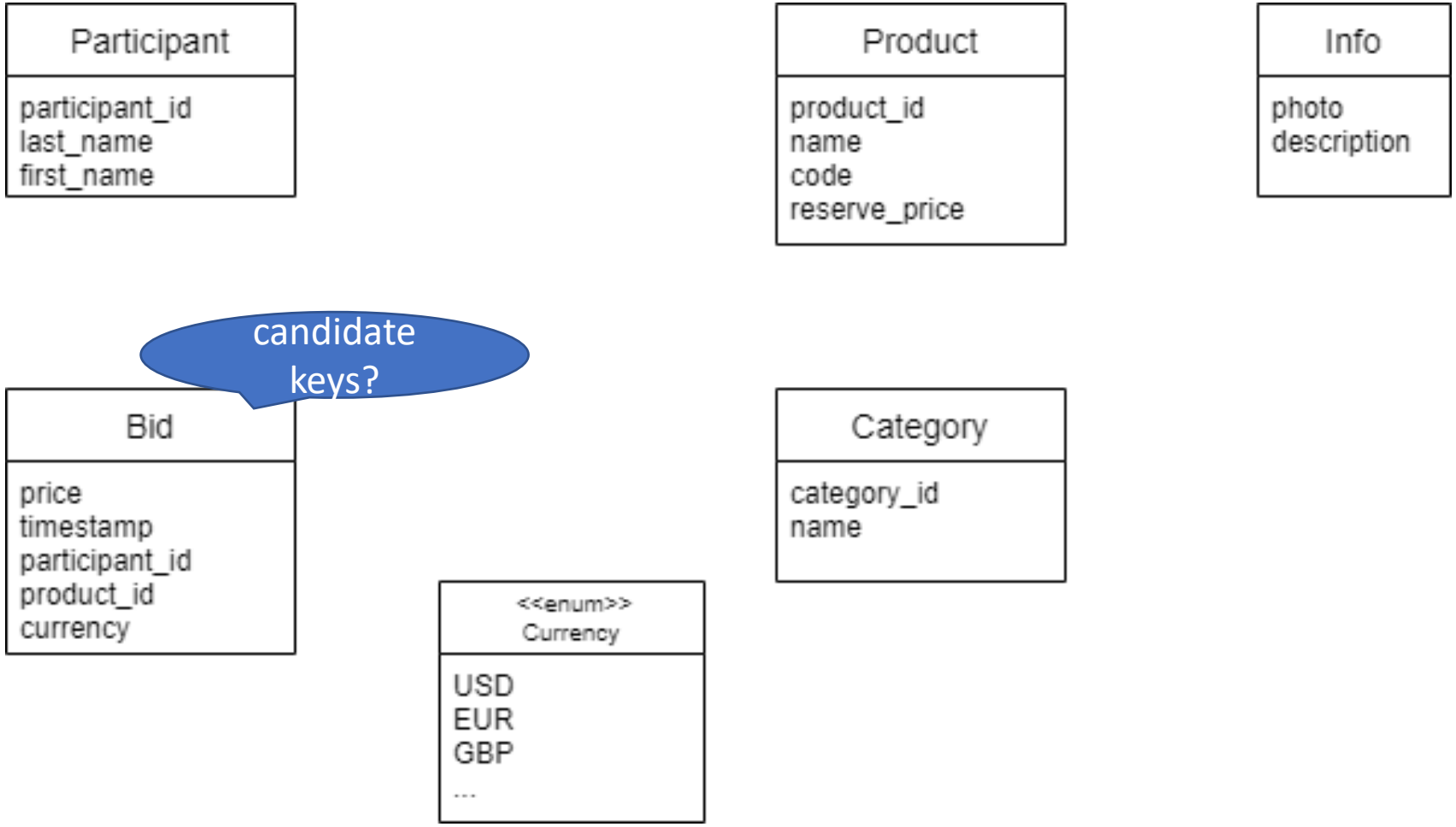
BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Candidate keys: (BUYER_ID, DATE) (BUYER_ID, VIP) (PRICE, DATE) (HALL_NO, DATE) (HALL_NO, TYPE) (BUYER_ID, PRICE, HALL_NO) (BUYER_ID, PRICE, TYPE) , ~~(PRICE_DATE, DATE, TYPE)~~

Candidate keys

BUYER_ID	PRICE	HALL_NO	DATE	TYPE
1	150	Coliseum	08/03/22	VIP
1	150	Lyttelton	14/04/22	A
2	140	Olivier	01/05/22	A
2	90	Coliseum	04/06/22	B
2	220	Lyttelton	08/03/22	VIP
3	140	Olivier	14/04/22	B
3	220	Olivier	20/03/22	VIP

Candidate keys: (BUYER_ID, DATE) (BUYER_ID, VIP) (PRICE, DATE) (HALL_NO, DATE) (HALL_NO, TYPE) (BUYER_ID, PRICE, HALL_NO) (BUYER_ID, PRICE, TYPE) ,



Please answer www.menti.com 44 32 42 0 Q8

Primary key

- ***Unique*** identifier.
- ***not null*** must be known at any moment.
- Simple, no ambiguities.
- Stable.
- Must be a candidate key

- Composed keys may be replaced with an *artificial key (surrogate key)*.
- In many RDBMS we may use autoincremented values.

Primary key UUID/GUID

- universally unique identifier 128-bit
- Probability of collision (that a UUID is duplicated) is negligible.
- No need to change when merging to databases. (example: store on local database, merge when connected to central database)
- Known before the insertion of a new row, without querying the database.

- Please answer www.menti.com 8045 7859 Q7

Primary key

- Undersize/oversize?.
- Has foreign keys? (how many lines are affected?)
- Is it stable?
- Is it simple?

- Please answer www.menti.com 8045 7859 Q7

Populating a Primary Key

- Identity automatically assigns a unique sequence number to each row inserted.
- Sequence or functions (functions that generate uuid).
- Examples:
  - Postgres SERIAL (populate with sequence values)
  - MySql `AUTO_INCREMENT`.
  - Oracle Sequence
  - My Sql Server IDENTITY

  - Mongo ObjectID -- UUID

Primary key UUID/GUID

- universally unique identifier 128-bit
- Not the best solution for clusters (sequential UUIDs might be used).

- Types:
  - Type 1 : 4 bytes + 2 bytes + 2 bytes + 2 bytes + 6 bytes = time + node
  - Type 4 : 122 bits randomly generated, 6 bits reserved for version and variant.

  - Bit for type

    type 1 2ad1db02-2ff0-11eb-adc1-0242ac120002
    type 4 a7bc2d72-7153-44a1-83df-d03dd298cf53

Primary key UUID/GUID

- universally unique identifier 128-bit
- Not the best solution for clusters (sequential UUIDs might be used).

- Types:
  - Type 1 : 4 bytes + 2 bytes + 2 bytes + 2 bytes + 6 bytes = time + node
  - Type 4 : 122 bits randomly generated, 6 bits reserved for version and variant.

  - Bit for type

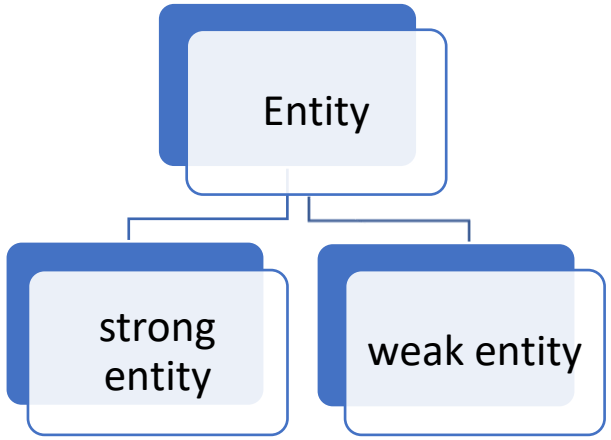
    type 1 2ad1db02-2ff0-11eb-adc1-0242ac120002
    type 4 a7bc2d72-7153-44a1-83df-d03dd298cf53

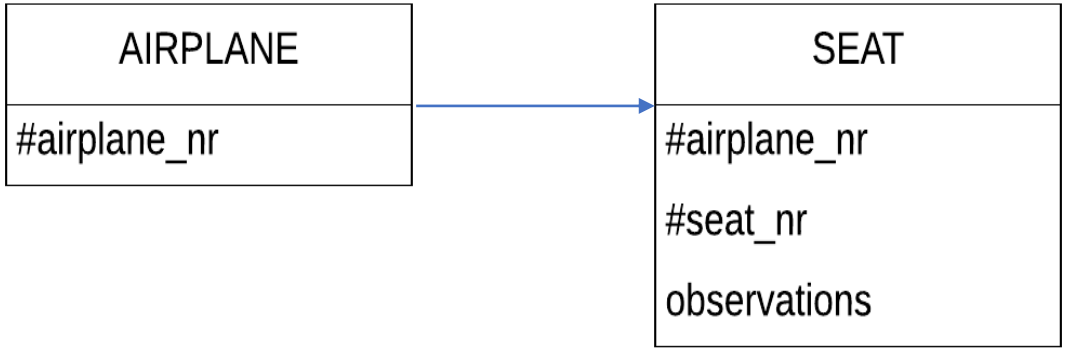
Airline (1)Entities

- The airline has one or more airplanes. An airplane has a model number, and capacity. Each flight is carried out by airplanes. An airplane is uniquely identified by its Registration_no and a flight is identified by its Flight_no. A passenger can book a ticket for a flight.

Please answer www.menti.com 8045 7859 Q10, Q11

Entities

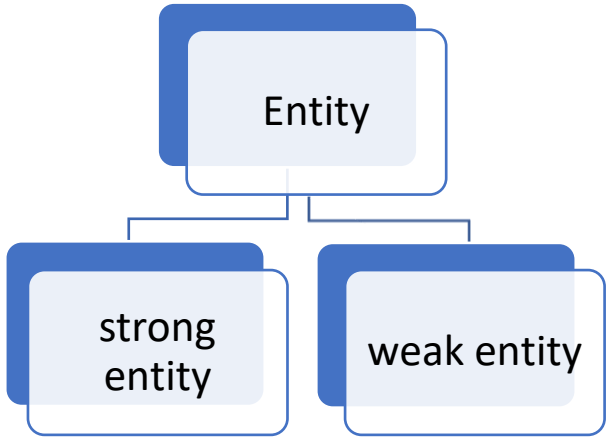


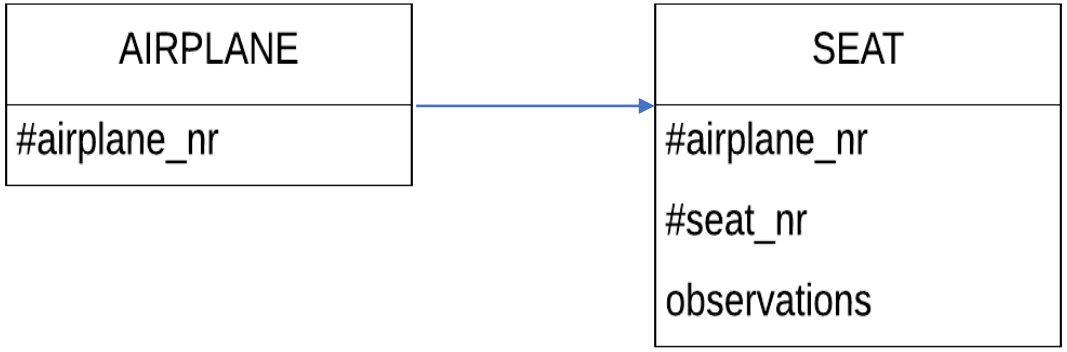


- Weak entity is an entity that depends on another entity.
- The primary key of a weak entity contains the primary key of the strong entity that it depends on + description/partial key.

Please answer www.menti.com 8045 7859 Q12, 13

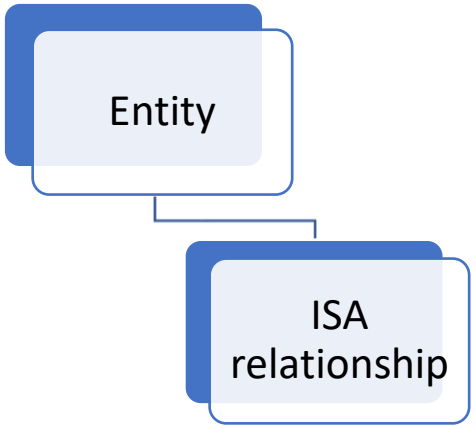
Entities

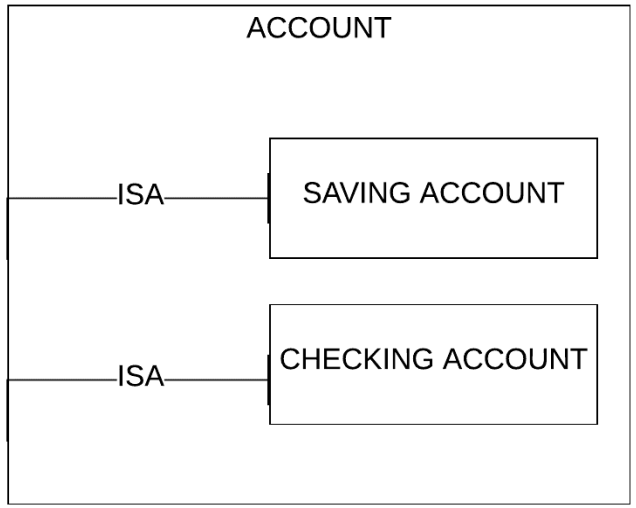




- Weak entity is an entity that depends on another entity.
- The primary key of a weak entity contains the primary key of the strong entity that it depends on + description/partial key.

Entities





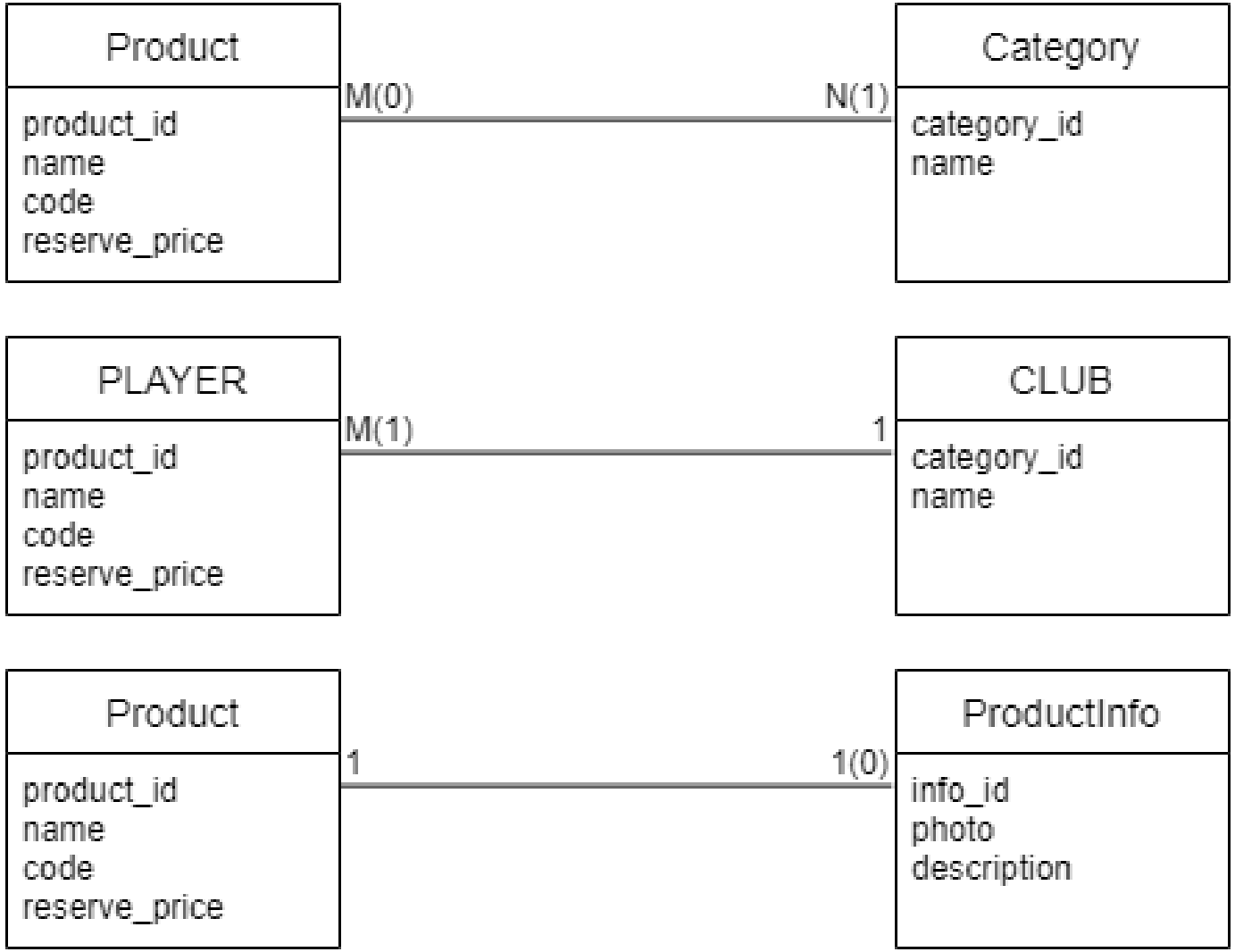
- A sub-entity has the same key as the *super*-entity and all its attributes and relationships.

Relationship

- Association between two or more entities (binary, ternary etc.)
- Relationship ➔ column (foreign key) or table.
- Graphical representation: oriented arc.

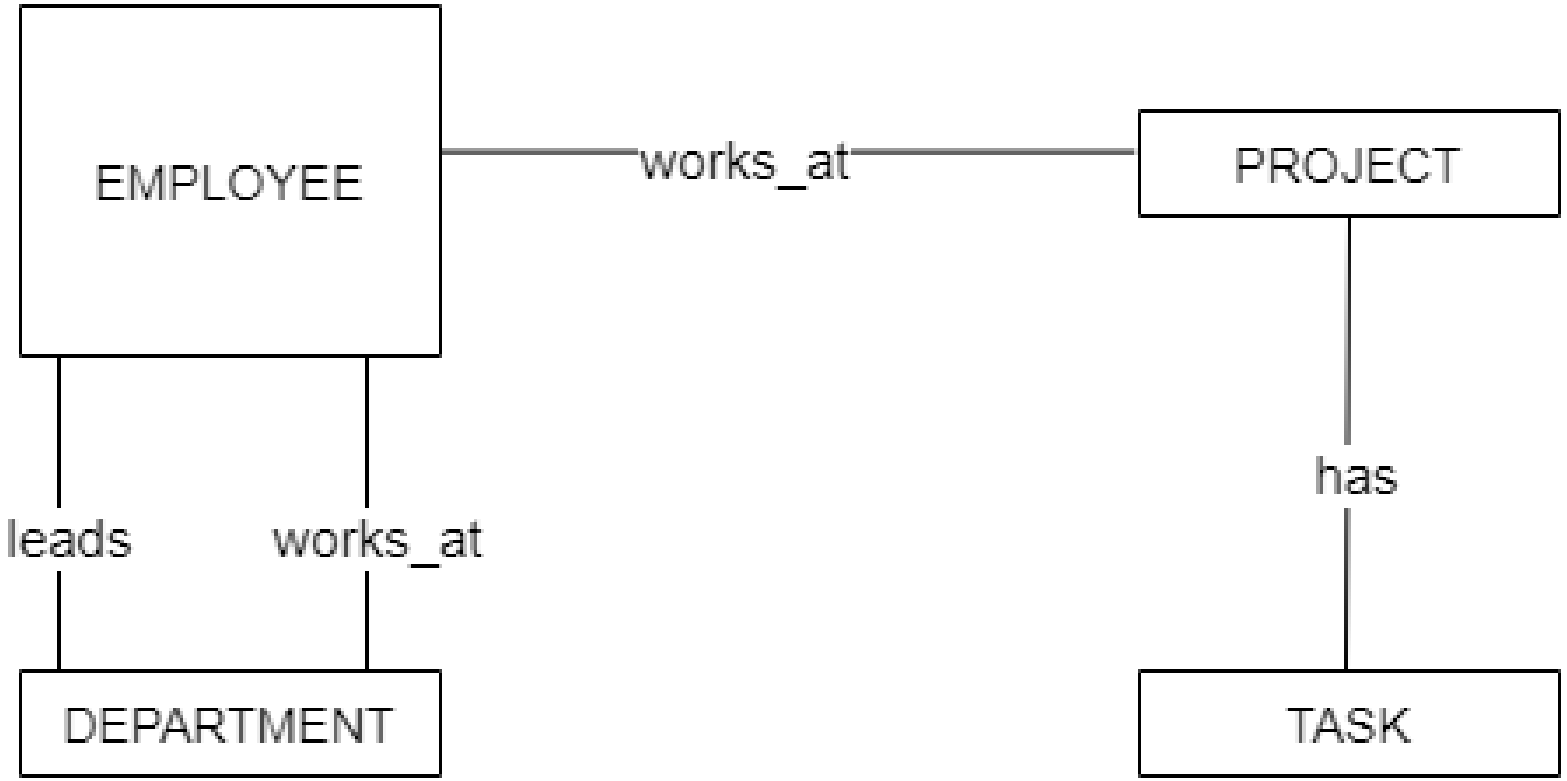
- Two relationships with the same name link different entities.

- Cardinality defines the numerical attributes of the relationship between two entities: MANDATORY (min) OPTIONAL (max)


Product
product_id
name
code
reserve_price
M(0)
N(1)
Category
category_id
name
PLAYER
product_id
name
code
reserve_price
M(1)
1
CLUB
category_id
name
Product
product_id
name
code
reserve_price
1
1(0)
ProductInfo
info_id
photo
description

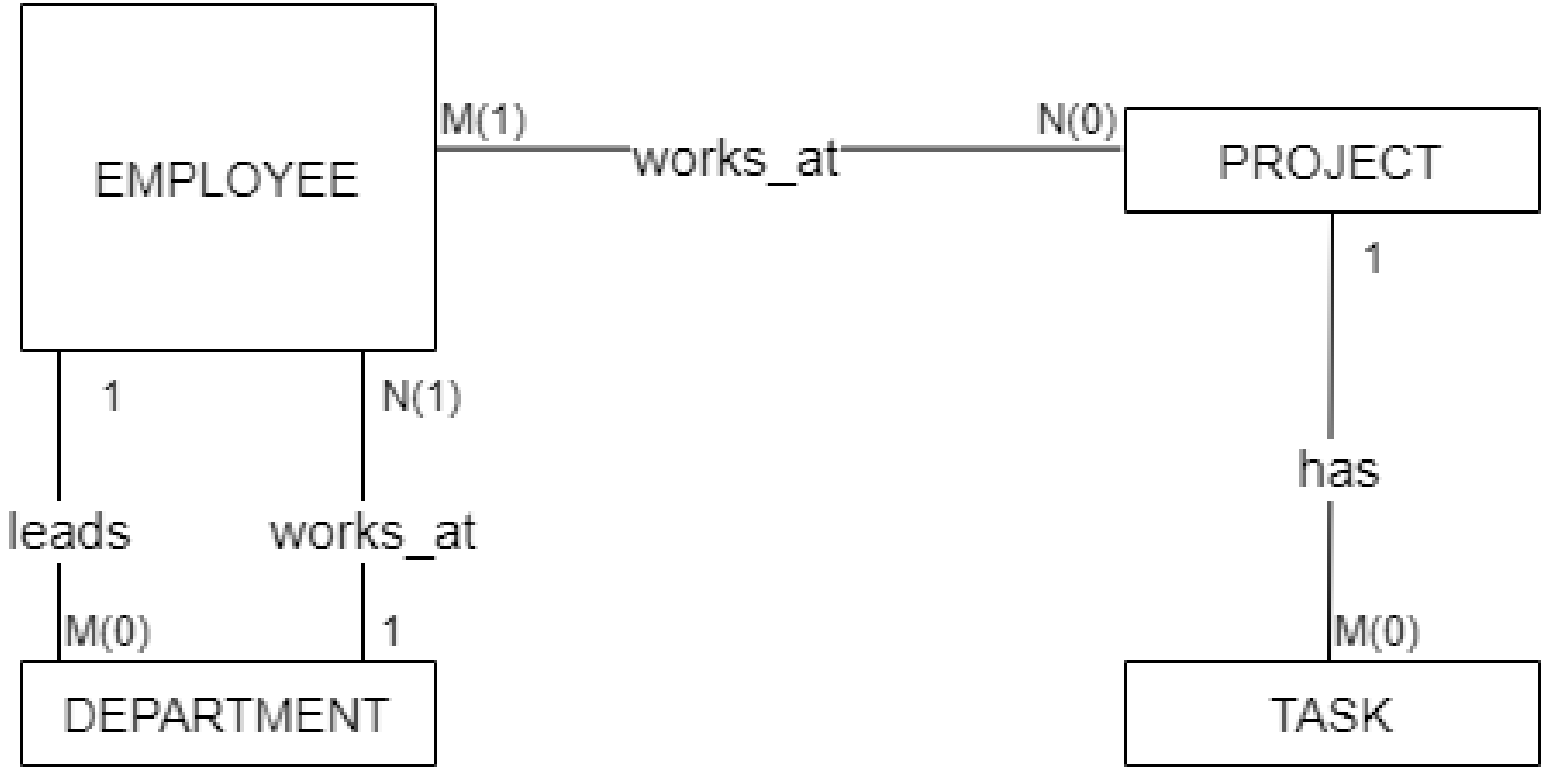
Relationship cardinality

- MANDATORY (must)
- OPTIONAL (may)



Relationship cardinality

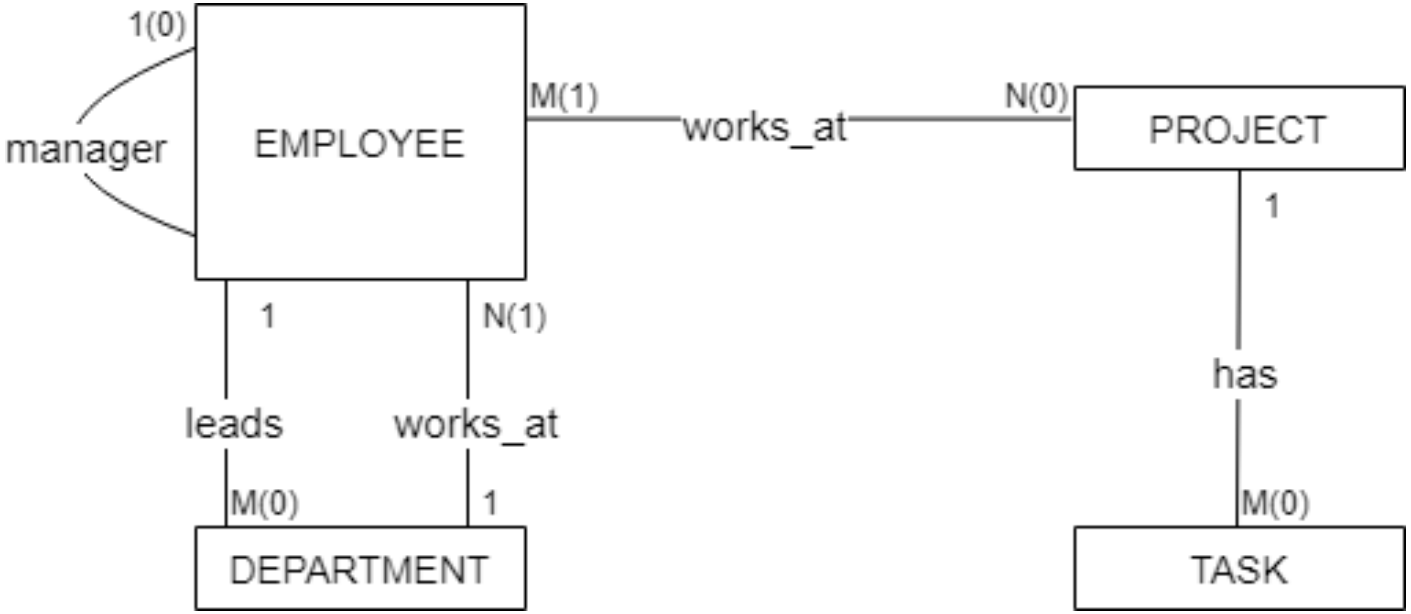
- MANDATORY (must)
- OPTIONAL (may)

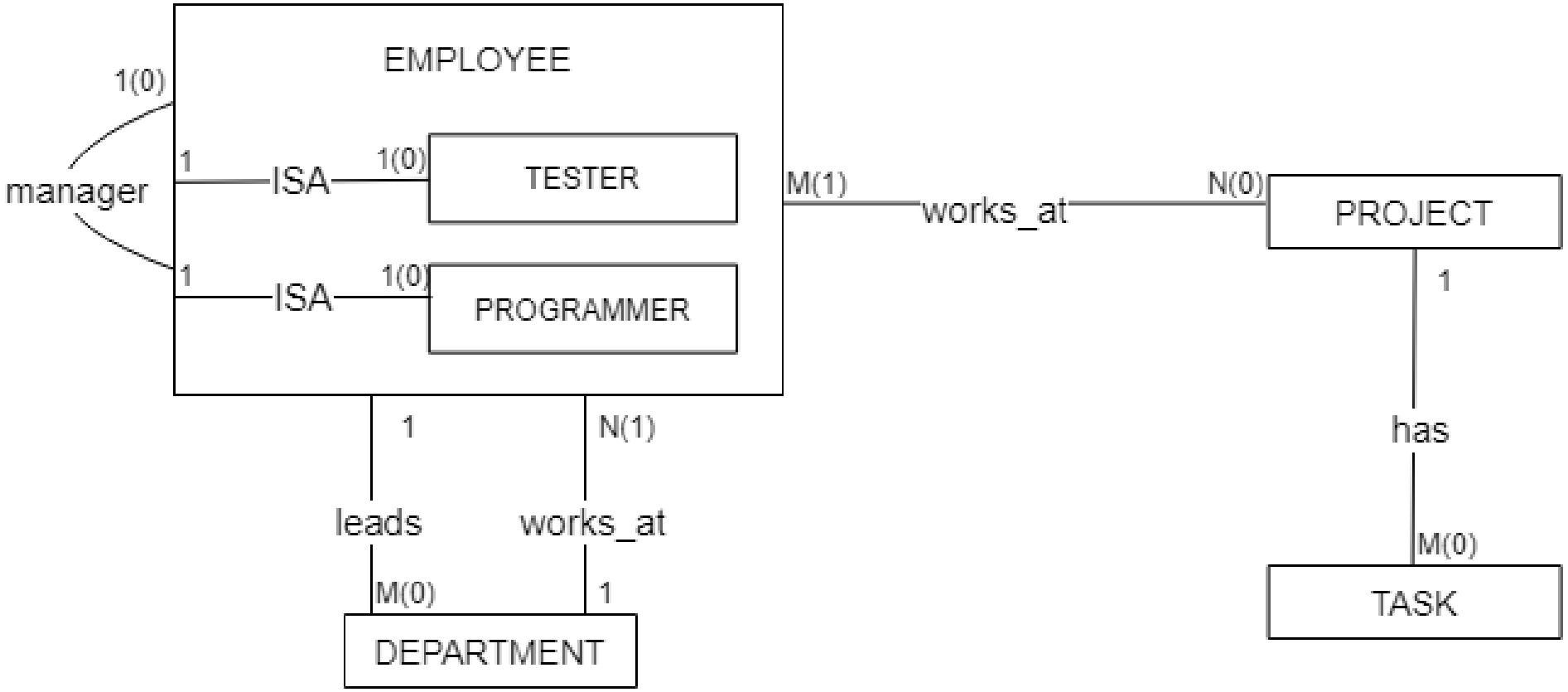


Relationship cardinality

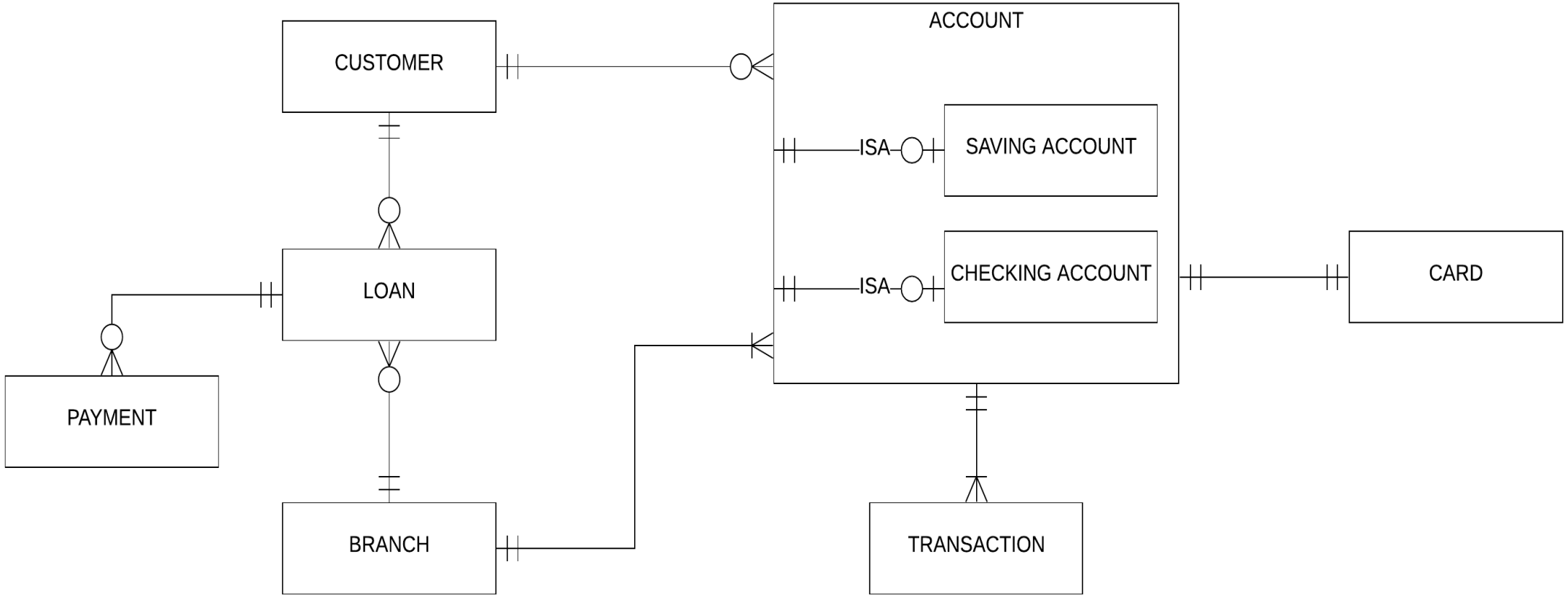
- Reflexive relationship

  unary relationship.




EMPLOYEE
1(0)
manager
1
ISA
1(0)
TESTER
1
ISA
1(0)
PROGRAMMER
M(1)
works_at
N(0)
PROJECT
1
has
M(0)
TASK
1
N(1)
leads
works_at
M(0)
1
DEPARTMENT

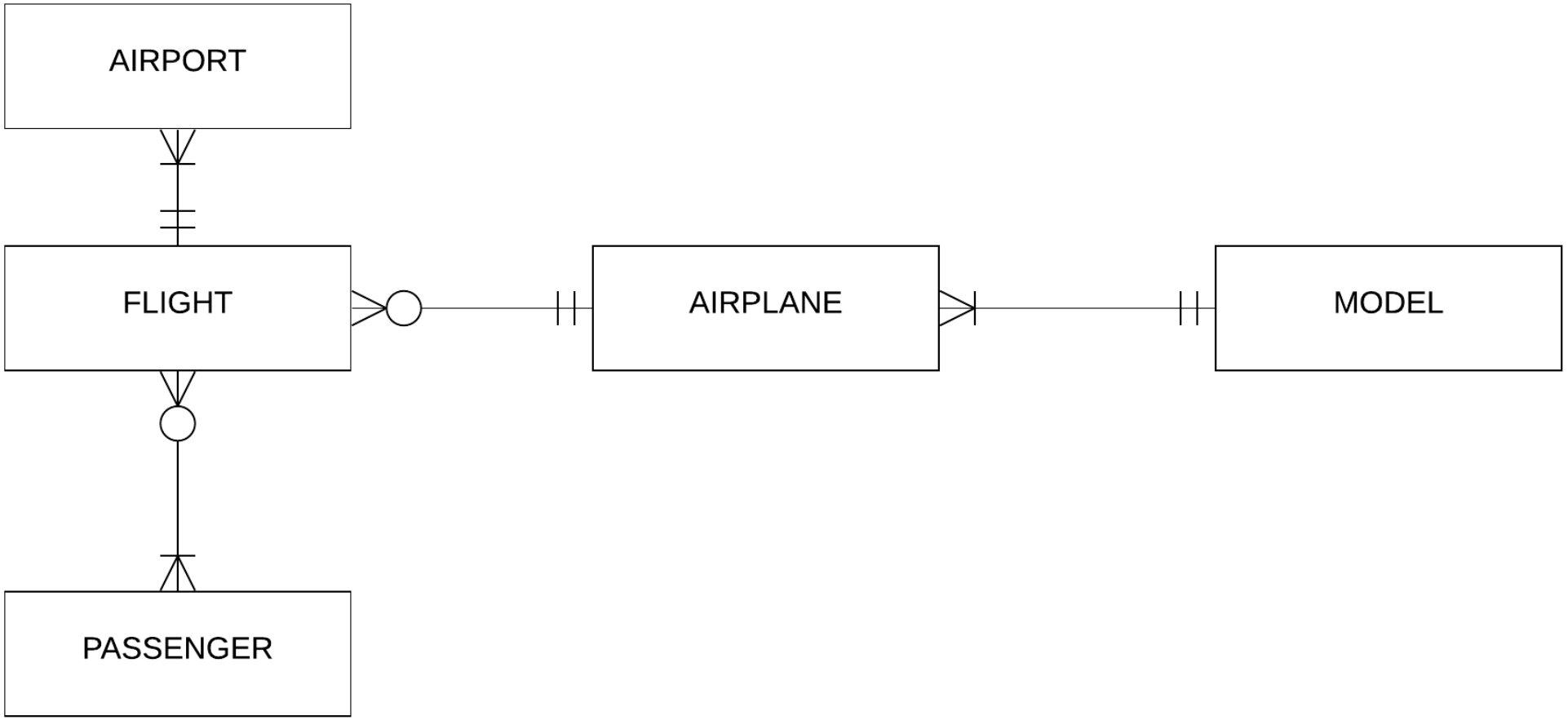
Banking Relationships



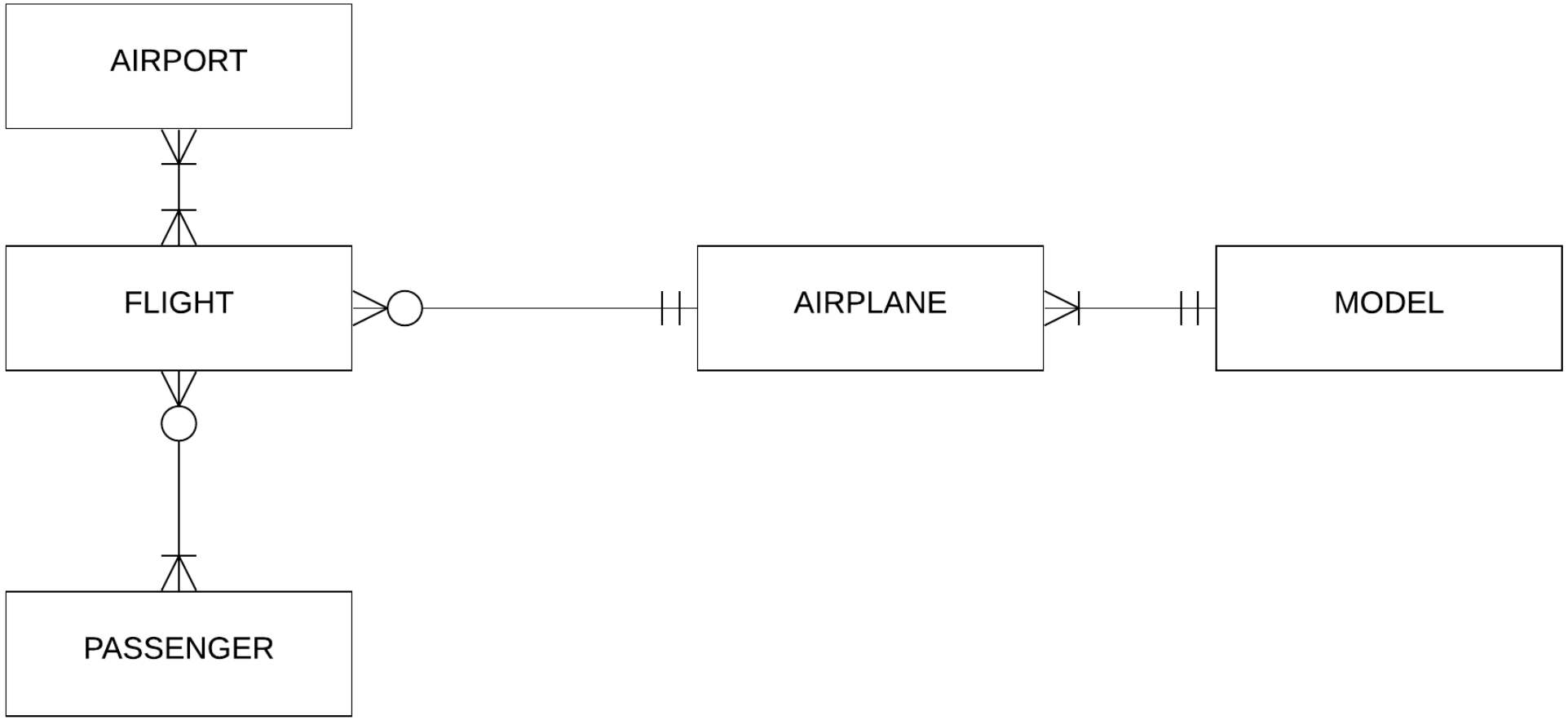
Airline Relationships

- The airline has one or more airplanes. An airplane has a model number, and capacity. Each flight is carried out by airplanes. An airplane is uniquely identified by its Registration_No and a flight is identified by its Flight_No. A passenger can book a ticket for a flight.

Airline



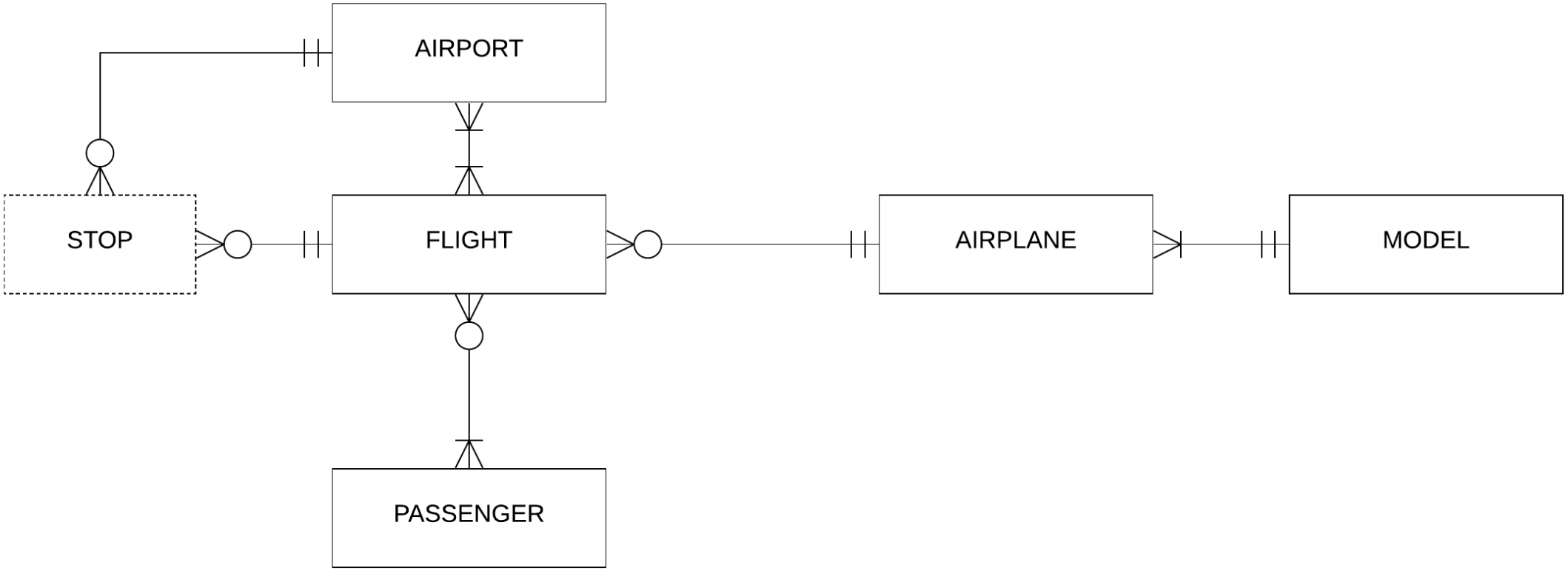
Airline



Airline Relationships

- The airline has one or more airplanes. An airplane has a model number, and capacity. Each flight is carried out by airplanes. An airplane is uniquely identified by its Registration_No and a flight is identified by its Flight_No. A passenger can book a ticket for a flight. A flight may have one or more stops.

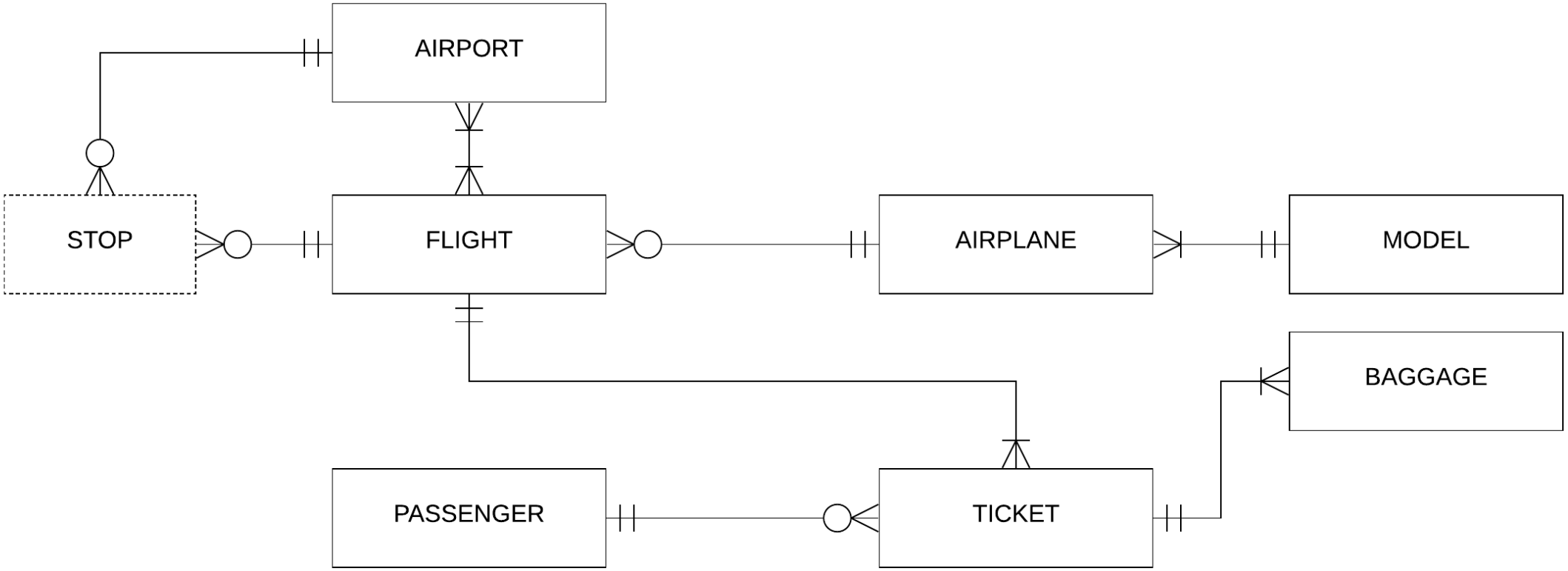
Airline



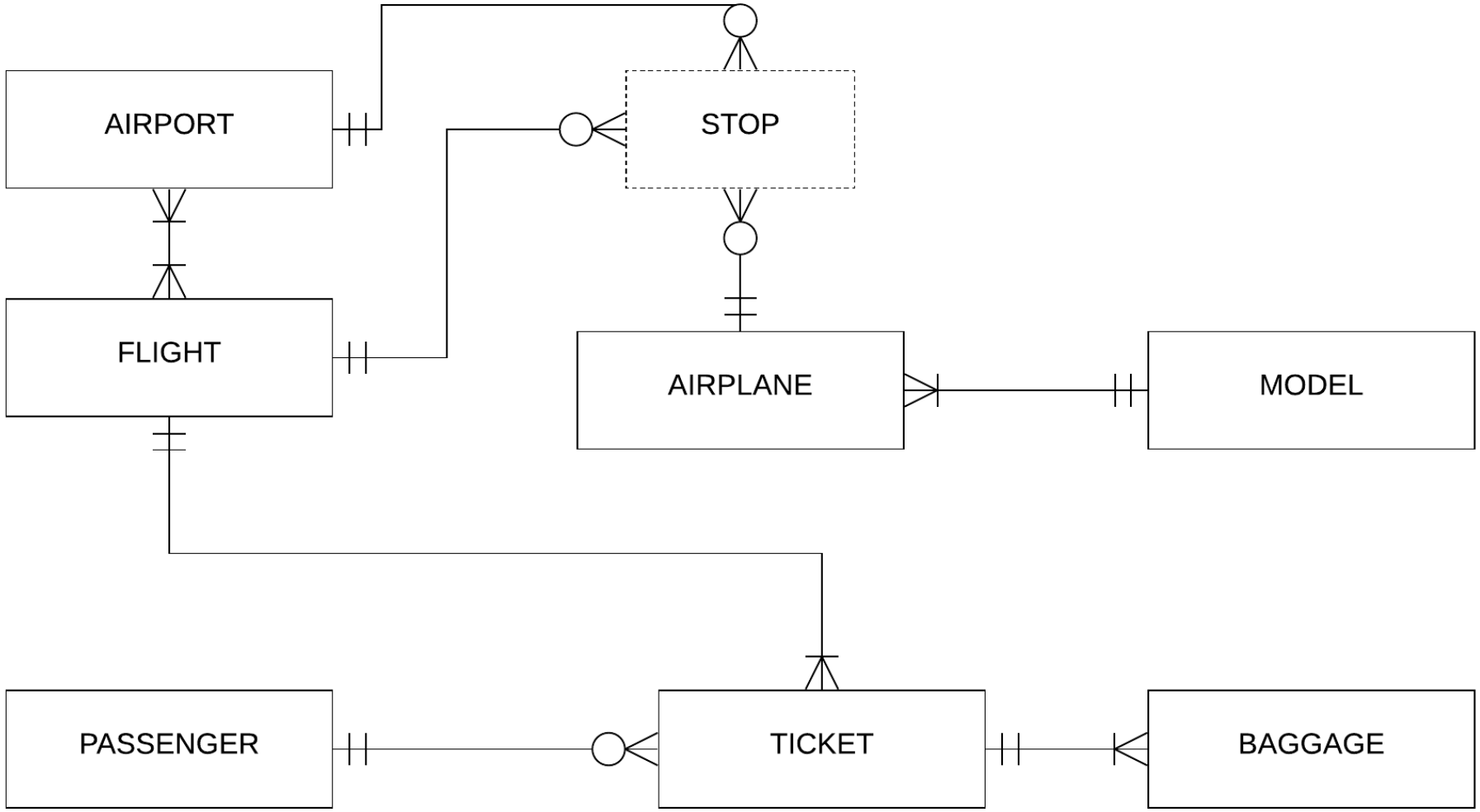
Airline Relationships

- The airline has one or more airplanes. An airplane has a model number, and capacity. Each flight is carried out by airplanes. An airplane is uniquely identified by its Registration_No and a flight is identified by its Flight_No. A passenger can book a ticket for a flight. A flight may have one or more stops. The passenger will pay for extra baggage.

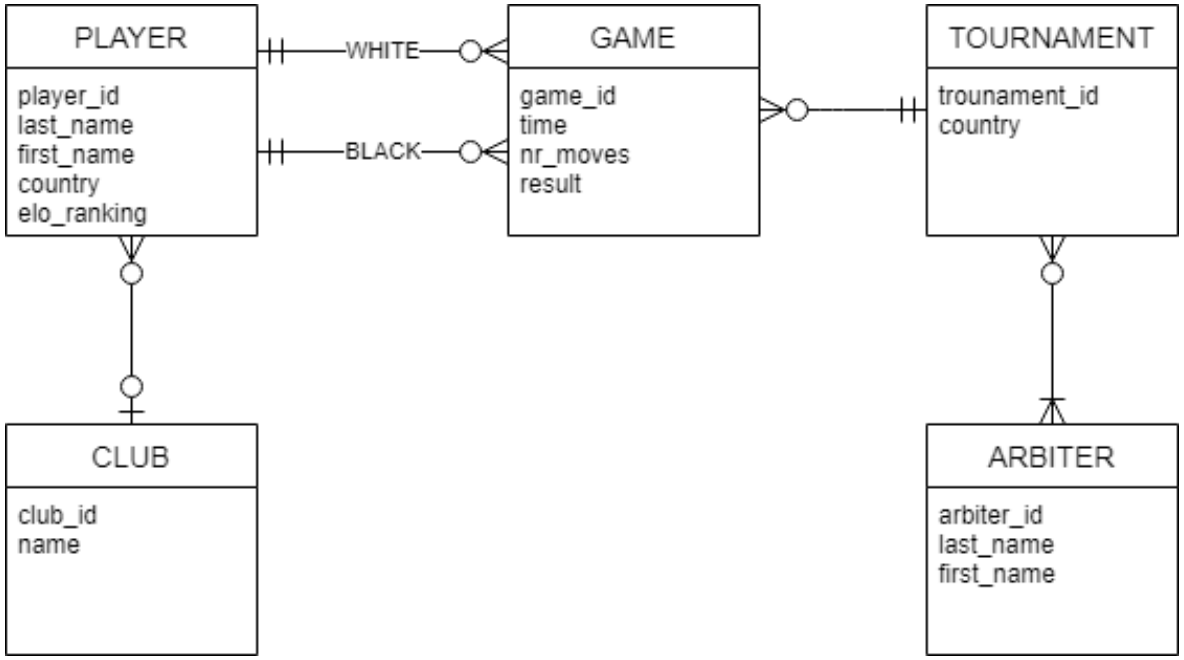
Airline


AIRPORT
STOP
FLIGHT
AIRPLANE
MODEL
BAGGAGE
PASSENGER
TICKET

Airline

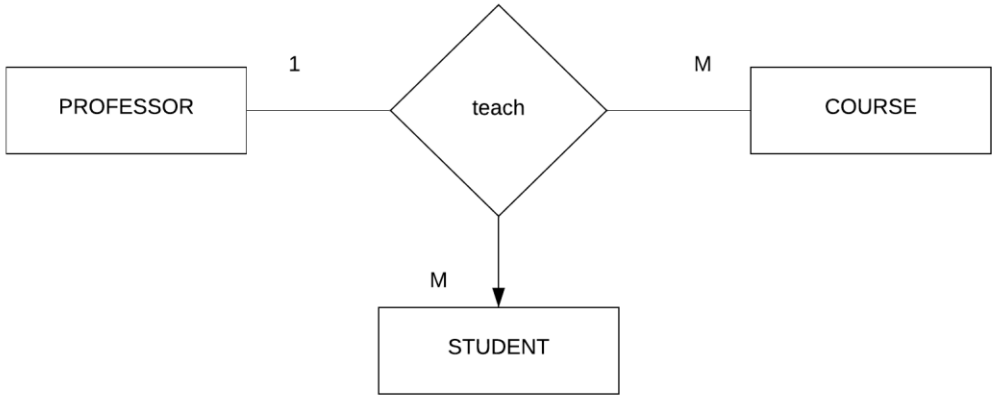

AIRPORT
STOP
FLIGHT
AIRPLANE
MODEL
PASSENGER
TICKET
BAGGAGE

Chess


PLAYER
player_id
last_name
first_name
country
elo_ranking
WHITE
BLACK
GAME
game_id
time
nr_moves
result
TOURNAMENT
trounament_id
country
CLUB
club_id
name
ARBITER
arbiter_id
last_name
first_name

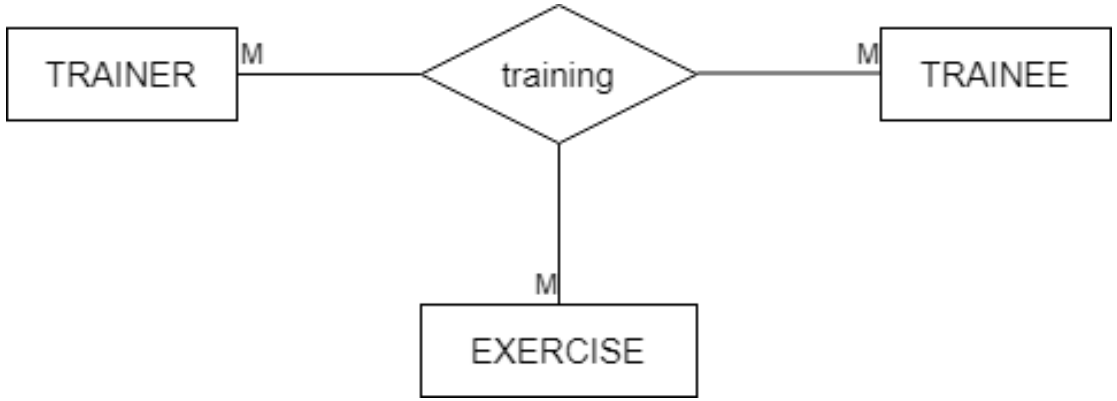
Ternary relationships

- Relationship binding simultaneously 3 entities.



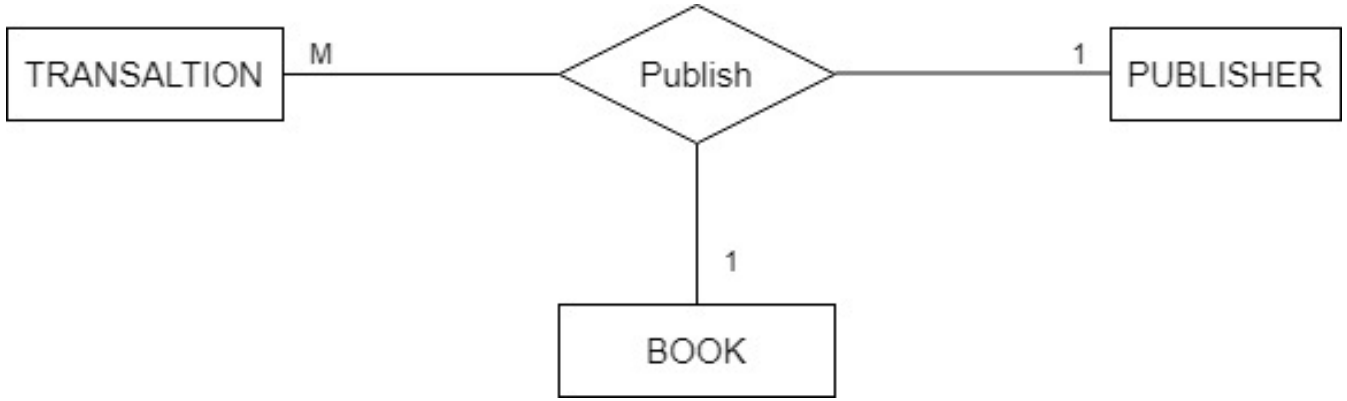
Ternary relationships

- Relationship binding simultaneously 3 entities.



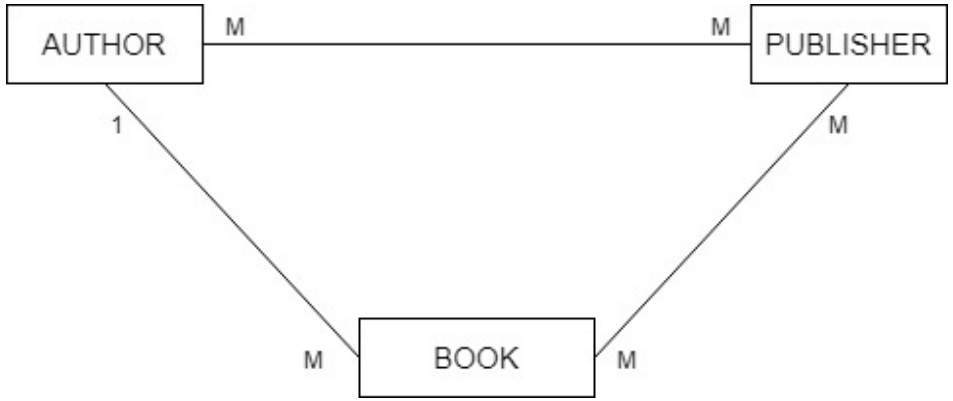
Ternary relationships

- Ternary or three binary?



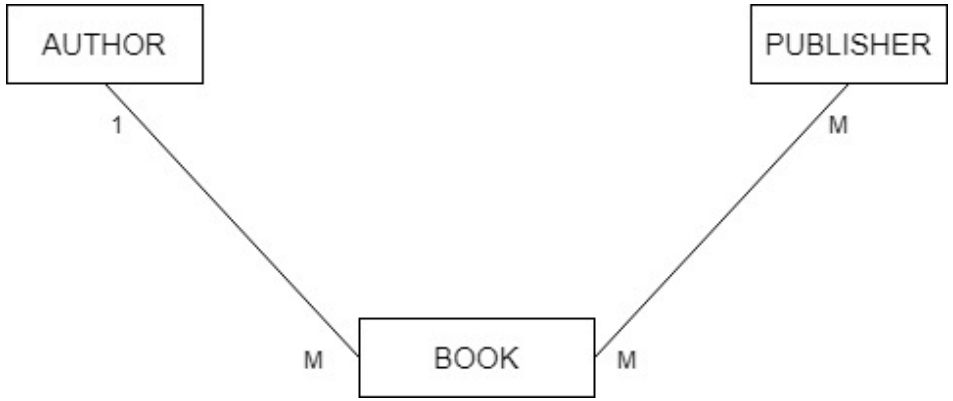
Ternary relationships

- Ternary or three binary?



Ternary relationships

- Ternary or three binary?



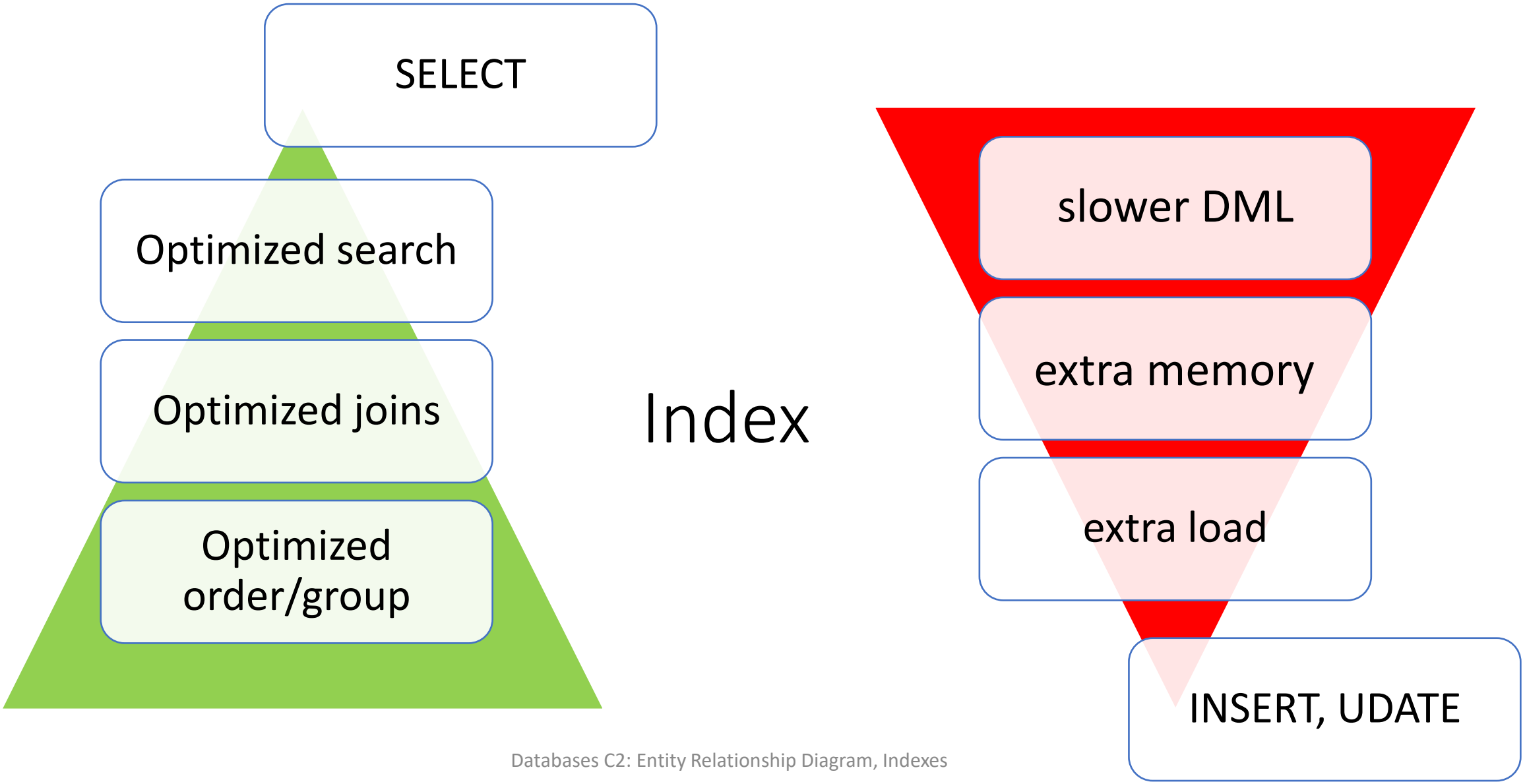
Indexes

Indexes

- Maps search key to data using specific data structures.

- Optimized search.
- Optimized joins (lookup in more than one table)
- Optimized order/group

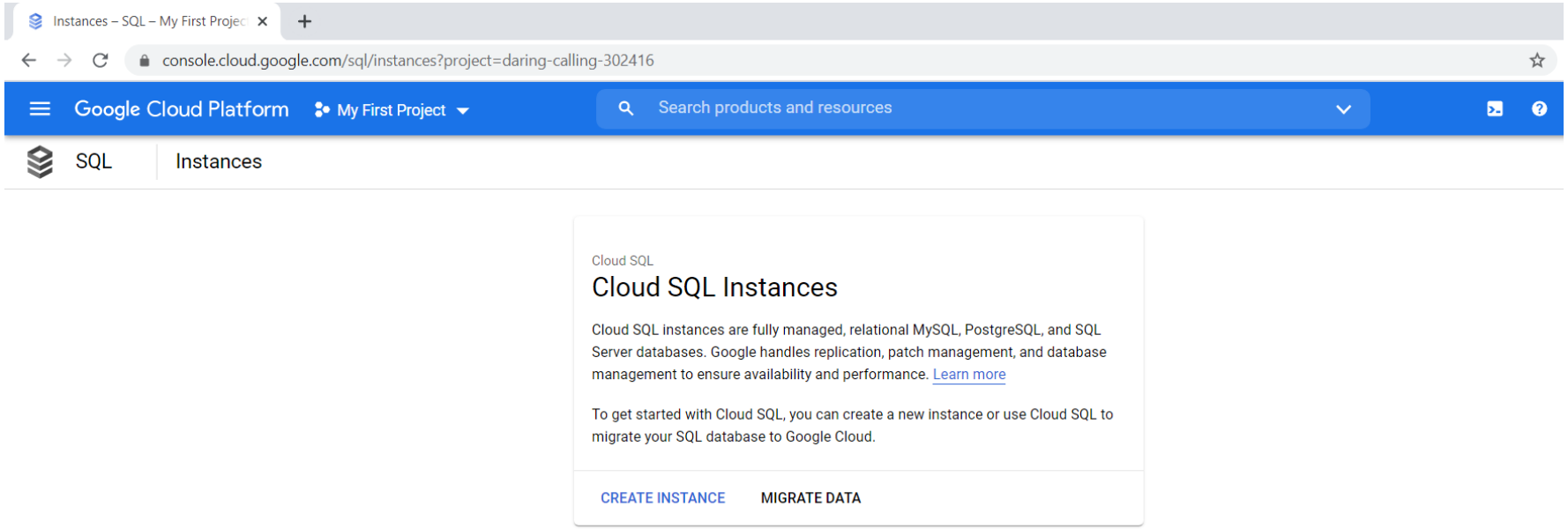
- slower DML (insert and update operations).
- extra memory


SELECT
Optimized search
Optimized joins
Optimized order/group
Index
slower DML
extra memory
extra load
INSERT, UDATE

Indexes demo

In Google Cloud go to **SQl – CREATE INSTANCE**.

Choose MySQL. PostgreSQL and SQLServer are also available.

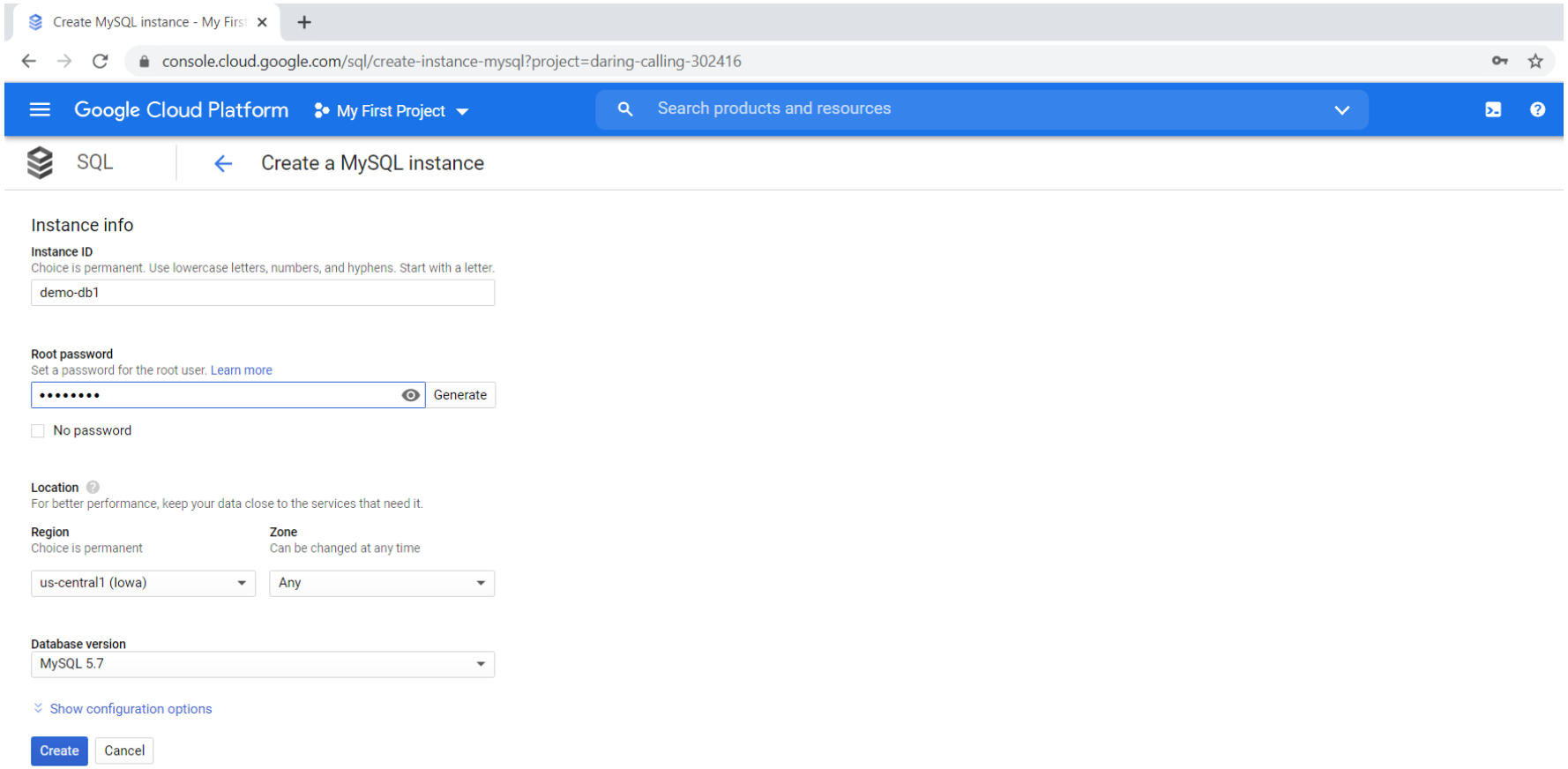


Indexes demo

Choose Instance ID and root password.

Wait for the instance to be created.

Connects with Cloud Shell and Cloud Proxy are available.

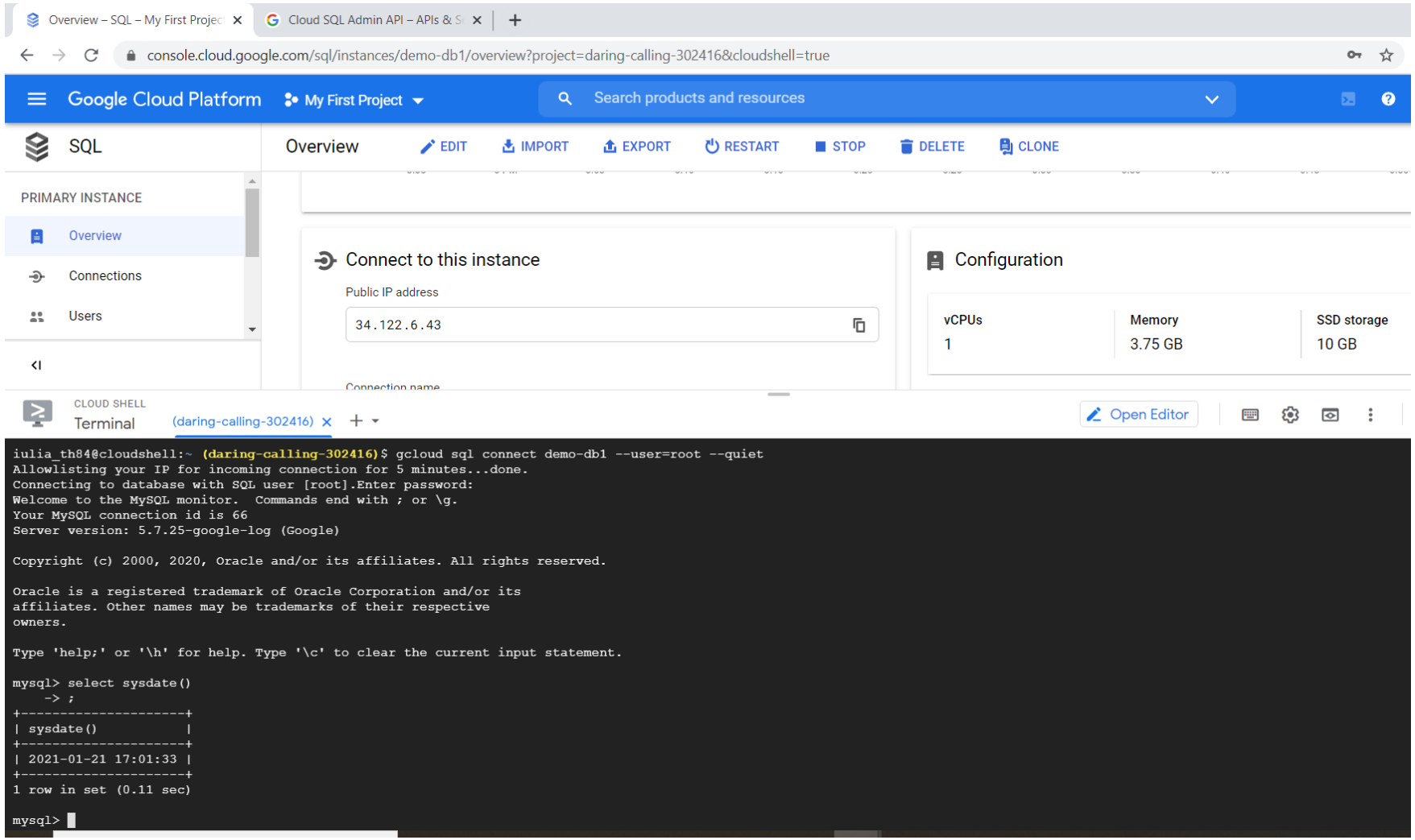


Indexes demo

Enable Cloud Admin and try connecting with cloud console.

To connect with Google Proxy follow these steps:

https://cloud.google.com/sql/docs/mysql/connect-admin-proxy

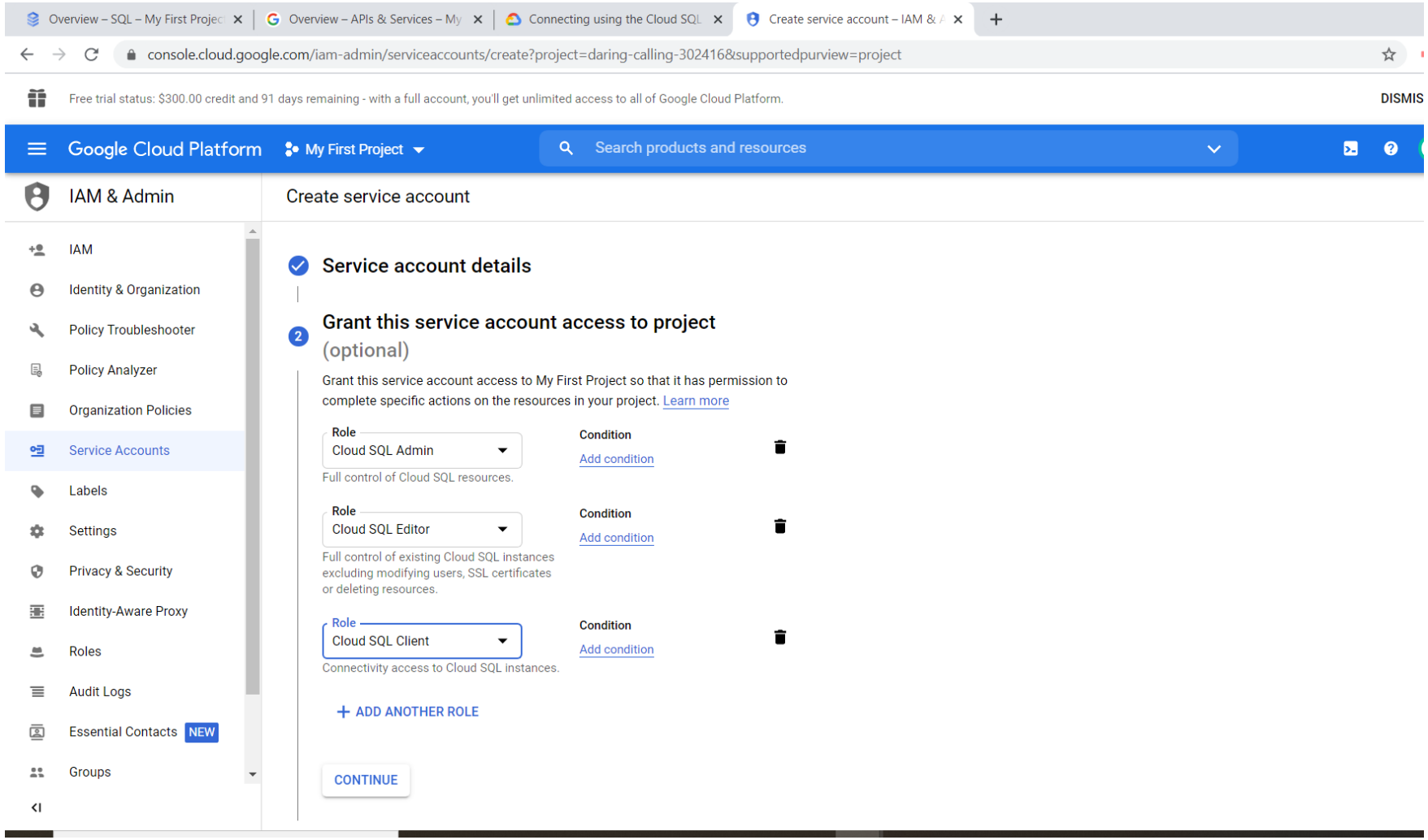


Indexes demo

To connect with Google Proxy follow these steps:

https://cloud.google.com/sql/docs/mysql/connect-admin-proxy

Create a service account and store the key.

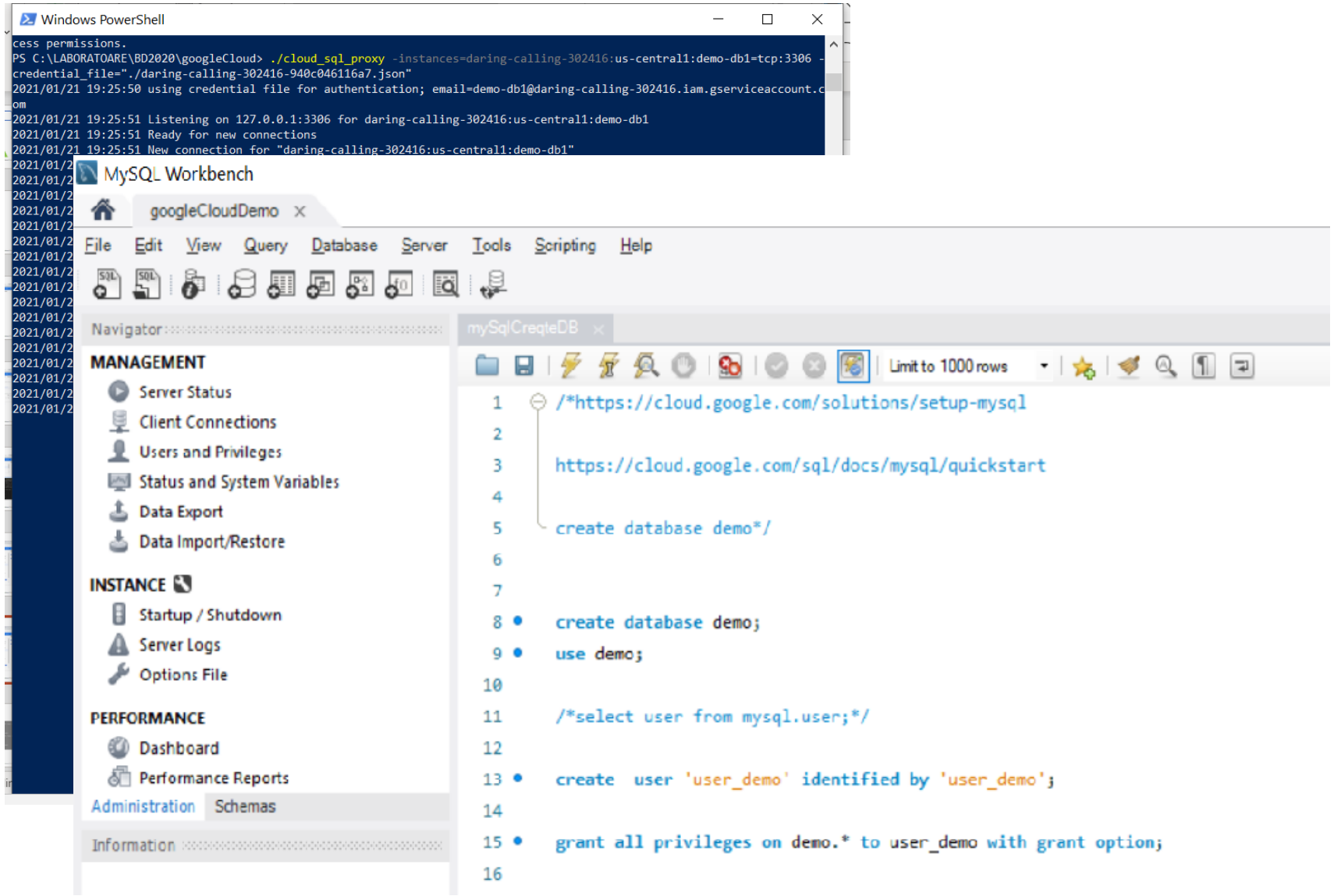


Indexes demo

Start cloud_sql_proxy:

./cloud_sql_proxy -instances=**daring-calling-302416:us-central1:demo-db1**=tcp:3306 -credential_file=**"./daring-calling-302416-940c046116a7.json"**

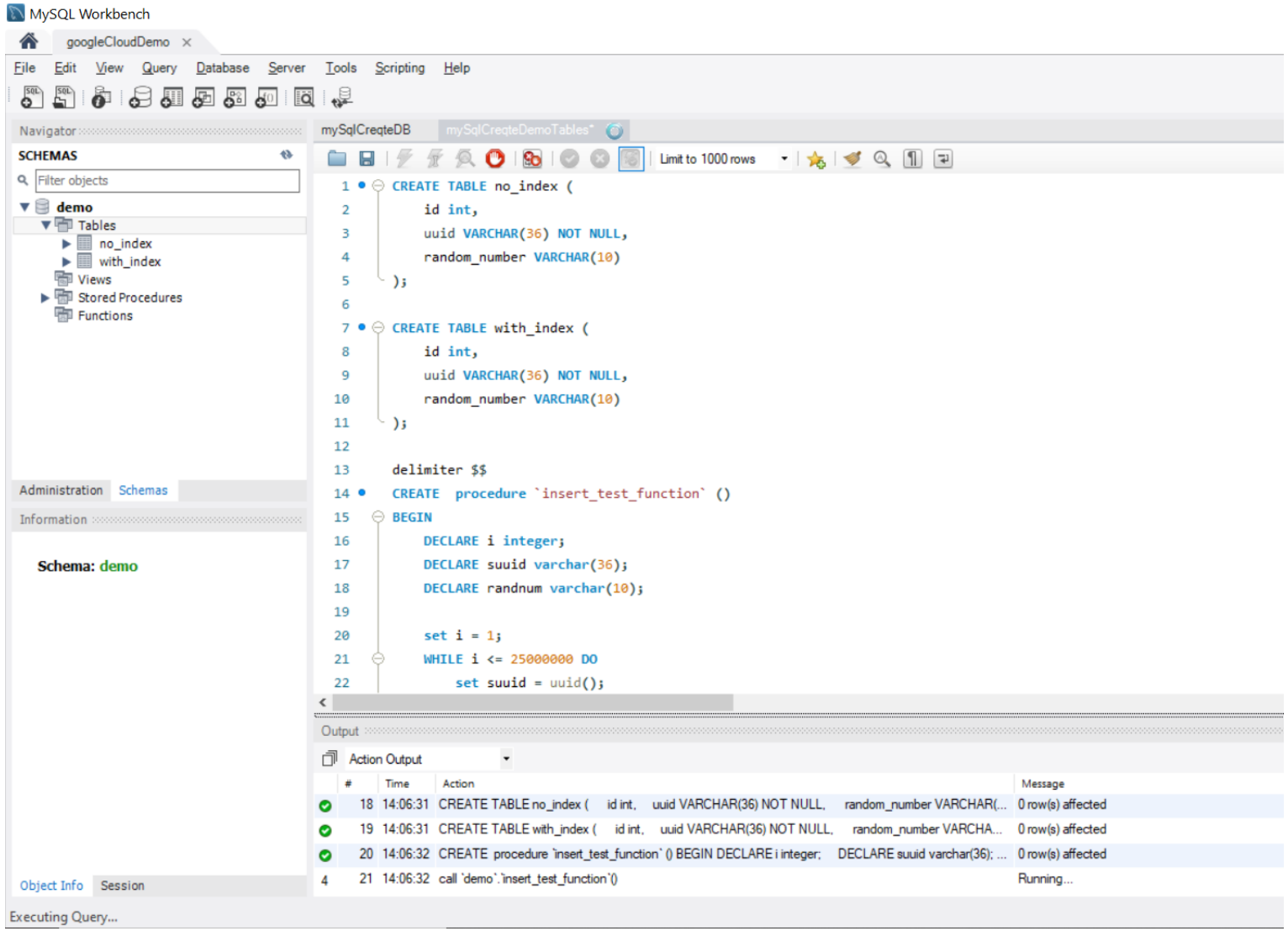
Connect with MySql Workbench with root user and run **mySqlCreqteDB.sql**.



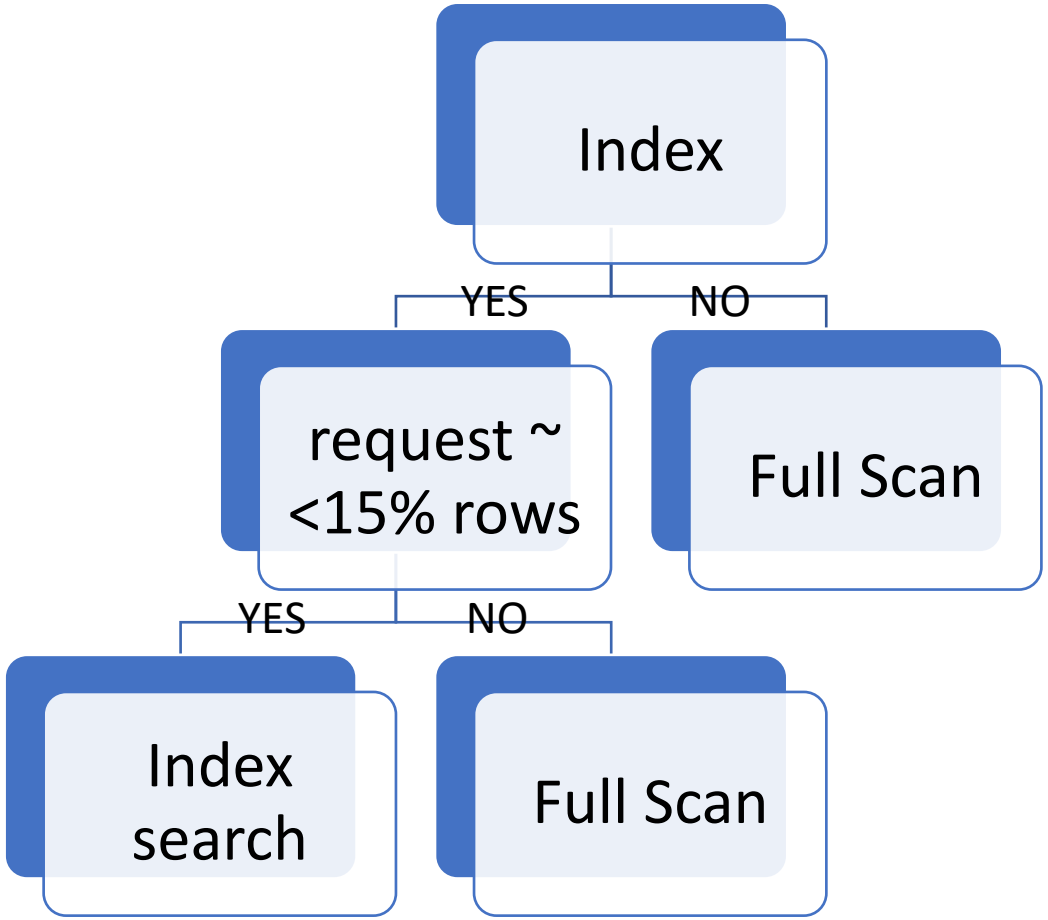
Indexes demo

Connect with MySql Workbench with user **demo** and run **mySqlCreqteDemoTables.sql**

and **C2AddIndexCompareTimes.sql**



Sql Optimizer



Autogenerated columns

- MySQL auto-generated index (key):
  - DB_ROW_ID increases monotonically as new rows are inserted.
  - DB_ROLL_PTR roll pointer, points to log record.
  - DB_TRX_ID last transaction that updated or inserted the row.

- Oracle rowid:
  - Pseudo column 18 characters = 10 + 4 + 4 (block, row, file).
  - Store and return row address in hexadecimal format (string).
  - Unique identifier for each row.
  - Immutable.

Autogenerated columns

- Oracle rowid:
  - Used in where clause to select/update/delete a row.
- Oracle rownum:
  - Sequential number in which oracle has fetched the row, before ordering the result
  - Temporary generated along with a select statement.
- Mongo
  - ObjectID (timestamp 4Bytes + random 5Bytes + Count 3Bytes).

Index

- Data structure that optimize search.
- Automatically created when a primary key is defined.

Primay key

- Constraint imposed on insert/update behavior.
- NotNull & Unique.

MySQL

```
SHOW EXTENDED INDEX FROM index_test;
```

MySQL

```
select * from  information_schema.statistics
where table_name = 'index_test1’
        and index_name = 'primary';
```

Oracle

```
select * from user_indexes
where table_name = 'INDEX_TEST';
```

Oracle

```
select * from user_constraints
where table_name = 'INDEX_TEST';
```