Relational Data Model

Definitions

- An entity class is a collection of entity instances that have a common structure. For example, a whole collection of student records could form an entity class.
- An entity instance represents a particular object of interest that is to be represented and tracked. For example, a student record is an entity instance that represents an individual student.
- An attribute represents a piece of interesting information, or a measurable fact, about the instances of an entitity class. For example, year of first registration is a fact about students that might be represented as an attribute of all the instances of the entity class student
- A domain is a set of values that can be assigned to an attribute; for example, the attribute birthday coudl be given values from the domain date.
- A relationship is an association between entities. Entities are often identified by nouns in a requirements specification, and relationships by verbs. For example, owns might form a relationship between entities person and vehicle. Relationships can be described as relationships between entity classes, or between entity instances.
- Mathematically, a relation consists of a heading, which is a subset of the Cartesian product of a set of (attribute name, domain) pairs, and a body, which contains (attribute name, value) pairs. For example, the entity class student could be represented as a heading, (student number, integer), (student name, text) and a body containing values like (student number, 123),(student name, Bloggs) A relation is implemented as a table in a relational database.
- A candidate key is a minimal set of attributes that identifies each individual row in a table (each tuple in a relation). For example, suppose there was a relation Slotroom,day,time in a timetabling application. Then room,day,time or class,day,time would serve as alternative candidate keys for the relation.
- The primary key is the candidate key that has been nominated to identify individual rows in a table. For example, in the timetabling relation above, room,day,time would be likely to form a suitable primary key because class is likely to change.

Database Integrity

ACID

- Atomicity - something is either done completely, or not done at all. The state of doing it is not visible outside the database.
- Consistency - The database is in a legal state at all times. When a transaction occurs, it can not break the rules. These rules are about integrity, what is allowed and what is not allowed in certain locations of the database.
- Isolation - There can be more than one transaction occurring at the same time. A certain transaction will not see changes made by other transactions.
- Durability - When a transaction is done, it will be committed. After it is committed, it can no longer be undone.

NoSQL

Semistructured Data + XML

FLOWR

FOR, LET, WHERE, ORDER BY, RETURN.

Security

Areas of concern for data security:

Confidentiality	Ensure only authorised parties have access to data	Use Encryption, authorisations and authentication
Integrity	Ensure data is not modified by authorised or unauthorised parties. and is not corrupt by system limitations	Use checksums, hashes and digital signatures
Availability	Ensure the data is accessible when needed	backups, redundancies, attack mitigation

Symmetric vs Asymmetric

Symmetric: Going one way is the same as going the other. Eg encrypting and decrypting use the same key so must be kept super secure!
Asymmetric: Going one way is not the same as the other. Eg two keys are needed, one to encrypt and one to decrypt. Typically one is "Private" and one is "Public". Knowing one key does not allow you to go the other way.

DES - The Data Encryption Standard

- From IBM Early 1970's, Published and standardized in 80's
- 56-bit Keys (with additional 8 bits of parity)
- Fast to encrypt/decrypt in hardware and software
- superseeded by TDES and AES
- Uses sequence of Permutations and Substitutions, repeated 16 times
- Produces:
  - A Product Cipher - Repeating simple ciphers can produce more complex one
  - A Bloc Cipher - Operates on a block of data (can be adapted to streams)
  - A Symmetric-key Cipher - Encrypting and Decrypting keys are the same