Databases

International Baccalaureate · IB Diploma · Computer Science · SL · Vocabulary in context

Word	Example
database /ˈdeɪtəbeɪs/	Scientists put DNA sequences into a **database** that anyone can search. DNA
database management system /ˈdeɪtəbeɪs ˈmænɪdʒmənt ˈsɪstəm/	A **database management system** stores and controls the data.
data redundancy /ˈdeɪtə rɪˈdʌndənsi/	**Data redundancy** means the same data is stored twice.
inconsistency	An **inconsistency** happens when the same fact is stored differently in two places.
relational database /rɪˈleɪʃənl ˈdeɪtəbeɪs/	A **relational database** has linked tables.
table /ˈteɪbl/	The STUDENT **table** has one row for each student. STUDENT
record /ˈrekɔːd/	Each **record** in the school database holds one student's details.
field /fiːld/	The **field** "Name" stores each student's name. "Name"
primary key /ˈpraɪməri kiː/	StudentID is the **primary key**: no two students share it. StudentID StudentID。
foreign key /ˈfɒrən kiː/	ClassID in the STUDENT table is a **foreign key**. STUDENT ClassID
composite key /ˈkɒmpəzɪt kiː/	A **composite key** is made of two fields together.

exercises.lol exercises.com.cn

Word	Example
referential integrity /ˌrefəˈrenʃl ɪnˈtegrɪti/	**Referential integrity** stops a foreign key pointing to nothing.
entity /ˈentɪti/	A student is an **entity** in the school database.
entity-relationship diagram /ˈentɪti rɪˈleɪʃənʃɪp ˈdaɪəgræm/	An **entity-relationship diagram** shows how the tables link.
link table /lɪŋk ˈteɪbl/	A **link table** breaks a many-to-many relationship into two.
normalisation /ˌnɔːməlaɪˈzeɪʃn/	**Normalisation** removes repeated data from tables.
first normal form	In **first normal form**, each field holds only one value.
second normal form	**Second normal form** removes data that depends on only part of the key.
third normal form	**Third normal form** removes data that depends on a non-key column.
SQL /ˌes kjuː ˈel/	**SQL** is a language for asking a database questions.
join /dʒɔɪn/	A **join** puts data from two tables together.
aggregate function /ˈægrɪgeɪt ˈfʌŋkʃn/	SUM and COUNT are each an **aggregate function**. SUM COUNT
transaction /trænˈsækʃn/	A **transaction** either finishes completely or not at all.