Inserting Data

Name:	Class:	Date:

This worksheet follows on from 'A Nice New Table' worksheet

Now it's time to insert data into the tables teachers, students, and subjects.

This is done using

INSERT INTO *table(field_1, field_2,…) VALUES (val_1, val_2,…);*

There's some constraints on the data we put into a table using INSERT statements.

Constraint	**Example**	**Reason**
Data should match the datatype	If you have a field: trainer_name varchar(15) The data below will work: "Mitchel" This data sadly won't: Fin-tim-lin-bin-whin-bim-lim-bus-stop-F'tang-F'tang-Olé-Biscuitbarrel (Look it up)	The field trainer_name was given a maximum of 15 characters. This is a constraint based upon data type and could be an issue due to the limited character count.
You cannot give a value to an AUTO_INCREMENT	student_number AUTO_INCREMENT Cannot be input: INSERT INTO students (student_number, …_ VALUES (1, …)	As student_number's values are being set by the RDMS we cannot choose the number we want. The risk is we pick a pre-existing number or skip a record number.

Inserting Data

Constraint	Example	Reason
Putting a value for an ENUM field to something other than those in the list.	ENUM ('Database', 'Web Development', 'Software Development') 'Web Development' will work. 'Wed Bevelopment' Will not.	ENUM create a list of acceptable inputs that cannot be altered during INSERT. It is also common practice for one value to be the default: DEFAULT 'Database'
Having a set range of values for the field such as age ranges.	NOT NULL CHECK (student_age >=18 AND student_age <100) A student of age 3 would not enter. A student of 36 would.	Age restrictions can exist for many reasons. In many systems you would unlikely allow ages that could not exist 1000, or - 50. It could be argues that storing an age is in itself a poor choice as ages change.
The value is a FOREIGN KEY.	teacher_id is a FK in the subjects table. If teacher_id 10 does not exist in the teachers table, it will not accept the INSERT.	When you try to INSERT a foreign key the value must exist as a primary key in the related table, otherwise the relationship cannot be made.

Inserting Data

Now we can try to add data to a table.

Task 1.1

We can add data to a table in different ways.

A single record method can be done as follows:

INSERT INTO teachers(trainer_id, trainer_name, trainer_dob, salary) VALUES (1, "Zak", '1996-09-20',20000.3);

The fields have been declared in order: (trainer_id, trainer_name, trainer_dob, salary)

Then the values: (1, "Zak", '1996-09-20',20000.3);

Task 1.2

Add the following records:

2, 'Tim','1994-01-01',15000.2

3, 'Christain','1993-01-01',35900.3

Task 1.3

You can also add multiple records in the same insert by separating the records with commas.

INSERT INTO *table* (*fields…)* VALUES (*values)*, (*values*);

Try it with the data below:

(4, 'Richard','1969-01-01',100900.3), (5, 'Waqas', '1922-01-01', 50000.2)

Inserting Data

Challenges

There's more ways to INSERT data but you need to make sure you get it right.

Challenge 1

Both students and subjects have AUTO_INCREMENT on the primary key.

This means you cannot provide this value when you do the INSERT.

Add the following data to students:

18, 10540523, 'Harry', 'Biker'

Look at your students table schema. The first value (18) is not the student_number#

Challenge 2

Add a record into the subjects column this follows the constraints for that table.

Here's a recap of the constraints:

subject_id - PK AUTO_INCREMENT

trainer_id FK (Must have a matching record in teachers)

subject_name ENUM('Database', 'Web development;, 'Software Development')

Challenge 3

Insert three more records for the students and subjects tables.

Remember to follow the constraints.

Challenge 4

What happens when you use INSERT but do not specify column names?

INSERT INTO teachers VALUES (6, 'Narayan', '1982-01-01', 100500.1);

Try with students and subjects to see if it works.