The SQL UPDATE Statement

TCL's COMMIT and ROLLBACK

Transaction Control Language

- **the COMMIT statement**
  - saves the transaction in the database
  - changes cannot be undone
- **the ROLLBACK clause**
  - allows you to take a step back
  - the last change(s) made will not count
  - reverts to the last non-committed state

TCL’s COMMIT and ROLLBACK

the COMMIT statement

- saves the transaction in the database
- changes cannot be undone

TCL's COMMIT and ROLLBACK

the COMMIT statement

- saves the transaction in the database
- changes cannot be undone

used to save the state of the data in the database at the moment of its execution

TCL's COMMIT and ROLLBACK

the COMMIT statement

- saves the transaction in the database
- changes cannot be undone

used to save the state of the data in the database at the moment of its execution

the ROLLBACK clause

- allows you to take a step back
- the last change(s) made will not count
- reverts to the last non-committed state

TCL's COMMIT and ROLLBACK

the COMMIT statement

- saves the transaction in the database
- changes cannot be undone

used to save the state of the data in the database at the moment of its execution

the ROLLBACK clause

- allows you to take a step back
- the last change(s) made will not count
- reverts to the last non-committed state

it will refer to the state corresponding to the last time you executed COMMIT

TCL’s COMMIT and ROLLBACK

TCL's COMMIT and ROLLBACK

```
COMMIT;
```

1

TCL's COMMIT and ROLLBACK

COMMIT; COMMIT;

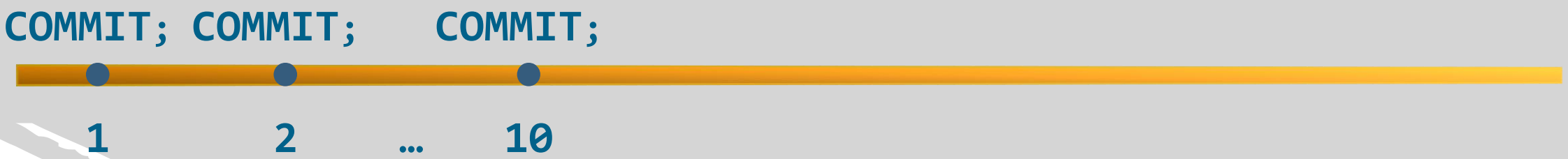
1 2

TCL's COMMIT and ROLLBACK

COMMIT; COMMIT;

1 2 …

TCL's COMMIT and ROLLBACK



TCL's COMMIT and ROLLBACK

TCL's COMMIT and ROLLBACK

- ROLLBACK will have an effect *on the last execution* you have performed

TCL's COMMIT and ROLLBACK

- ROLLBACK will have an effect *on the last execution* you have performed

TCL's COMMIT and ROLLBACK

- ROLLBACK will have an effect *on the last execution* you have performed

TCL's COMMIT and ROLLBACK

- ROLLBACK will have an effect *on the last execution* you have performed

TCL's COMMIT and ROLLBACK

- ROLLBACK will have an effect *on the last execution* you have performed
- you cannot restore data to a state corresponding to an earlier COMMIT

The UPDATE Statement

The UPDATE Statement

- the UPDATE Statement

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

SQL

```
UPDATE table_name
SET column_1 = value_1, column_2 = value_2 …
WHERE conditions;
```

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

SQL

```
UPDATE table_name
SET column_1 = value_1, column_2 = value_2 …
WHERE conditions;
```

- we do not have to update each value of the record of interest

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

SQL

```
UPDATE table_name
SET column_1 = value_1, column_2 = value_2 …
WHERE conditions;
```

- we do not have to update each value of the record of interest
- we can still say we have updated the specific record

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

SQL

```
UPDATE table_name
SET column_1 = value_1, column_2 = value_2 …
WHERE conditions;
```

The UPDATE Statement

the UPDATE Statement

used to update the values of existing records in a table

SQL

```
UPDATE table_name
SET column_1 = value_1, column_2 = value_2 …
WHERE conditions;
```

- if you don’t provide a WHERE *condition*, all rows of the table will be updated