Select operations in SQL

# Various select operations are possible in SQL

1. Selecting rows

2. Selecting columns

3. Selecting rows and columns

4. Updating rows and columns

Selecting rows: Selecting rows by condition
Identify transactions greater than $100

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Select rows where `PurchAmount > 100`
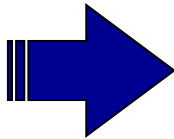
Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
...	...	...	...	...

R data.table   `transactions[PurchAmount > 100, ]`

R data.table method [1]

SQL in R   `dbGetQuery(con, "SELECT * FROM transactions`
                              `WHERE PurchAmount > 100;")`
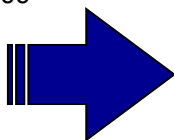
[2] `WHERE` specifies the condition, which rows to select.

Selecting rows: Selecting rows by condition
Identify transactions greater than $100

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Select rows where `PurchAmount > 100`

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
...	...	...	...	...

R data.table   `transactions[PurchAmount > 100, ]`

❶ R data.table method
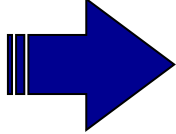
SQL in R   `dbGetQuery(con, "SELECT * FROM transactions`
                              `WHERE PurchAmount > 100;")`

❷ `WHERE` specifies the condition, which rows to select.

Selecting columns: Select multiple columns by column name

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Select rows where `PurchAmount > 100`

Customer	TransDate	PurchAmount
149332	15-11-05 00:00	199.95
172951	29-08-08 00:00	199.95
120621	19-10-07 00:00	99.95
149236	14-11-05 00:00	39.95
149236	12-06-07 00:00	79.95
...	...	...

R data.table

```
transactions[,list(Customer, TransDate, PurchAmount)]
```
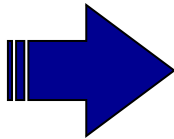
SQL in R

```
dbGetQuery(con, "SELECT Customer, TransDate, PurchAmount
                FROM transactions;")
```

Specifies the columns that are queried after `SELECT`

Selecting rows and columns: Combine operations to select by row and column

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Select rows where `PurchAmount > 100`

TransDate	Cost
15-11-05 00:00	107.00
29-08-08 00:00	108.00
…	…

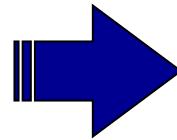**R data.table** `transactions[PurchAmount > 100, list(TransDate, Cost)]`

SQL in R

```
dbGetQuery(con, "SELECT TransDate, Cost
                 FROM transactions
                 WHERE PurchAmount > 100;")
```
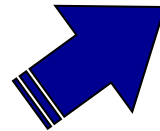
<u>Updating rows and columns</u>: Appending rows into a table (1/2)

transactions

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...
80801	19.09.2014	1	89.95	29.30

New row

Customer	TransDate	Quantity	PurchAmount	Cost
80801	19-09-14 00:00	1	89.95	29.30

Updating rows and columns: Appending rows into a table (1/2)

transactions

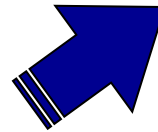| Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...
80801	19.09.2014	1	89.95	29.30

New row

Customer	TransDate	Quantity	PurchAmount	Cost
80801	19-09-14 00:00	1	89.95	29.30

Updating rows and columns: Appending rows into a table (1/2)

transactions

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...
80801	19.09.2014	1	89.95	29.30

New row

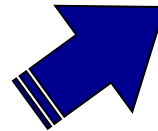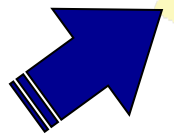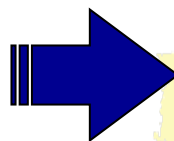| Customer	TransDate	Quantity	PurchAmount	Cost
80801	19-09-14 00:00	1	89.95	29.30

Updating rows and columns: Appending rows into a table (1/2)

transactions

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
...	...	...	...	...
80801	19.09.2014	1	89.95	29.30

New row

Customer	TransDate	Quantity	PurchAmount	Cost
80801	19-09-14 00:00	1	89.95	29.30

Updating rows and columns: Appending rows into a table (2/2)

R data.table
```
rbind(transactions,data.table(Customer=80801,TransDate="2014-
09-19 00:00",Quantity=1,PurchAmount=89.95,Cost=29.30,
TransID=227998739,TransKey=40000))
```

> ❶ INSERT INTO specifies the relevant table and columns. For missing columns, SQL will automatically insert NAs.

SQL in R
```
dbGetQuery(con,"INSERT INTO transactions(Customer, TransDate,
                                         Quantity, PurchAmount,
                                         Cost,
                                         TransID, TransKey)
          VALUES (80801,'2014-09-19 00:00',1,89.95,
          29.30, 227998739, 40000);")
```

> ❷ VALUES specifies the data that should be appended

*NOTE: the output "*data frame with 0 columns and 0 rows*" in the R console can be safely ignored.

Updating rows and columns: Appending rows into a table (2/2)

R data.table

```
rbind(transactions,data.table(Customer=80801,TransDate="2014-
09-19 00:00",Quantity=1,PurchAmount=89.95,Cost=29.30,
TransID=227998739,TransKey=40000))
```

① `INSERT  INTO` specifies the relevant table and columns. For missing columns, SQL will automatically insert `NA`s.

SQL in R

```
dbGetQuery(con,"INSERT INTO transactions(Customer, TransDate,
                                          Quantity, PurchAmount,
                                          Cost,
                                          TransID, TransKey)
```

② `VALUES` specifies the data that should be appended

```
VALUES (80801,'2014-09-19 00:00',1,89.95,
29.30, 227998739, 40000);")
```

*NOTE: the output "*data frame with 0 columns and 0 rows*" in the R console can be safely ignored.

<u>Updating rows and columns</u>: Appending rows into a table (2/2)

R data.table
```
rbind(transactions,data.table(Customer=80801,TransDate="2014-
09-19 00:00",Quantity=1,PurchAmount=89.95,Cost=29.30,
TransID=227998739,TransKey=40000))
```

❶ `INSERT INTO` specifies the relevant table and columns. For missing columns, SQL will automatically insert `NA`s.

SQL in R
```
dbGetQuery(con,"INSERT INTO transactions(Customer, TransDate,
                                          Quantity, PurchAmount,
                                          Cost,
                                          TransID, TransKey)
```

❷ `VALUES` specifies the data that should be appended
```
VALUES (80801,'2014-09-19 00:00',1,89.95,
        29.30, 227998739, 40000);")
```

*NOTE: the output "*data frame with 0 columns and 0 rows*" in the R console can be safely ignored.

Updating rows and columns: Appending rows into a table (2/2)

R data.table
```
rbind(transactions,data.table(Customer=80801,TransDate="2014-
09-19 00:00",Quantity=1,PurchAmount=89.95,Cost=29.30,
TransID=227998739,TransKey=40000))
```

> INSERT INTO specifies the relevant table and columns. For missing columns, SQL will automatically insert NAs. ❶

SQL in R
```
dbGetQuery(con,"INSERT INTO transactions(Customer, TransDate,
                                         Quantity, PurchAmount,
                                         Cost,
                                         TransID, TransKey)
```

> VALUES specifies the data that should be appended ❷

```
              VALUES (80801,'2014-09-19 00:00',1,89.95,
              29.30, 227998739, 40000);")
```

*NOTE: the output "*data frame with 0 columns and 0 rows*" in the R console can be safely ignored.

Updating rows and columns: Appending rows into a table (2/2)

R data.table
```
rbind(transactions,data.table(Customer=80801,TransDate="2014-
09-19 00:00",Quantity=1,PurchAmount=89.95,Cost=29.30,
TransID=227998739,TransKey=40000))
```

❶ INSERT INTO specifies the relevant table and columns. For missing columns, SQL will automatically insert NAs.

SQL in R
```
dbGetQuery(con,"INSERT INTO transactions(Customer, TransDate,
                                         Quantity, PurchAmount,
                                         Cost,
                                         TransID, TransKey)
                VALUES (80801,'2014-09-19 00:00',1,89.95,
                29.30, 227998739, 40000);")
```
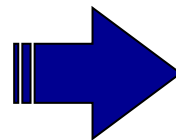
❷ VALUES specifies the data that should be appended

*NOTE: the output "*data frame with 0 columns and 0 rows*" in the R console can be safely ignored.

Updating rows and columns: Adding columns to a table

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Add column
`NewCol`

Customer	…	PurchAmount	Cost	NewCol
149332	…	199.95	107.00	107
172951	…	199.95	108.00	108
120621	…	99.95	49.00	49
149236	…	39.95	18.95	19
149236	…	79.95	35.00	35
…	…	…	…	…

R data.table

```
transactions[, NewCol := round(Cost)]
```

❶ First, add a new, empty column to the table with the `ALTER TABLE` & `ADD` commands
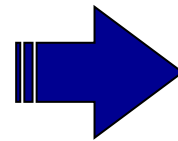
SQL in R

```
dbGetQuery(con, "ALTER TABLE transactions ADD NewCol;")
dbGetQuery(con, "UPDATE transactions
                 SET NewCol=ROUND(Cost,0);")
```

❷ In a second step, specify the values of the new column with the `UPDATE` & `SET` commands

Updating rows and columns: Adding columns to a table

Customer	TransDate	Quantity	PurchAmount	Cost
149332	15-11-05 00:00	1	199.95	107.00
172951	29-08-08 00:00	1	199.95	108.00
120621	19-10-07 00:00	1	99.95	49.00
149236	14-11-05 00:00	1	39.95	18.95
149236	12-06-07 00:00	1	79.95	35.00
…	…	…	…	…

Add column
`NewCol`

Customer	…	PurchAmount	Cost	NewCol
149332	…	199.95	107.00	107
172951	…	199.95	108.00	108
120621	…	99.95	49.00	49
149236	…	39.95	18.95	19
149236	…	79.95	35.00	35
…	…	…	…	…

R data.table
```
transactions[, NewCol := round(Cost)]
```

❶ First, add a new, empty column to the table with the `ALTER TABLE` & `ADD` commands

SQL in R
```
dbGetQuery(con, "ALTER TABLE transactions ADD NewCol;")
dbGetQuery(con, "UPDATE transactions
                 SET NewCol=ROUND(Cost,0);")
```

❷ In a second step, specify the values of the new column with the `UPDATE` & `SET` commands