

Relations and Functions

In maths and databases

Set Theory: Ordered Pairs

One of the reasons for studying sets is that we can represent functions using sets. Before defining functions we define a larger set of entities called [relations](#).

For this we need to define an [ordered pair](#) which is simply a pair of values selected from the same or different sets, where the ordering of the values is significant.

Reminder: Ordered Pairs

For example:

If the first value of each pair is an element of the set $\{1, 2\}$ and the second is an element of the set $\{3, 4\}$ then we can generate the following set of ordered pairs:

$\{(1,3), (1,4), (2,3), (2,4)\}$ (looks like a set of co-ordinates)

If we select from $\{3, 4\}$ first and $\{1, 2\}$ second we get:

$\{(3, 1), (4,1), (3,2), (4,2)\}$ which is a different set.

Set Theory: Product Operator

Product

We can generate a set of ordered pairs through use of the product operator \times .

That is, if $A = \{1,2\}$ and $B = \{3,4,5\}$ then:

$A \times B = \{(1,3),(1,4),(1,5),(2,3),(2,4),(2,5)\}$ and

$B \times A = \{(3,1),(3,2),(4,1),(4,2),(5,1),(5,2)\}$

Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation is simply a collection of associations between elements of one set and elements of another (possibly the same) set.

For example:

`isGreaterThan` associates integers with integers.
`isAPlaceIn` associates towns/cities of the world with countries
`isABrotherOf` associates male people with students.

Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation can be represented by a set of ordered pairs.

For example:

`isGreaterThan` $\{(3,1),(5,1),(3,2),(7,-2),(3,0),...\}$
`isAPlaceIn` $\{(London,England),(Edinburgh,Scotland),$
 $(Paris,France),(Paris,USA),...\}$
`isABrotherOf` $\{ ??? \}$

Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation can be represented pictorially as two sets and a rule / mapping:

`isGreaterThan`

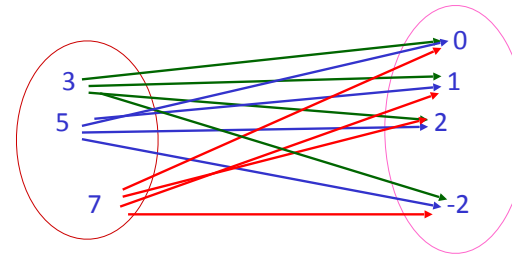


Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation can be represented pictorially as two sets and a rule / mapping:

`isGreaterThan`

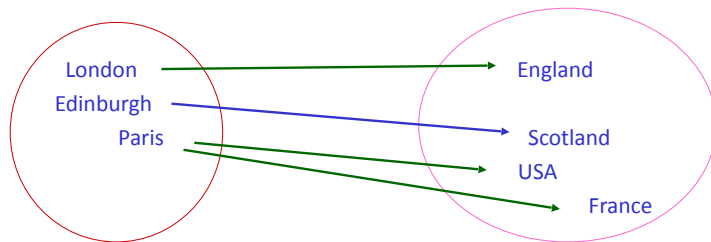


Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation can be represented pictorially as two sets and a rule / mapping:

`isAPlaceIn`



Mathematical Entities which can be Defined Using Sets: Relations

Relation

A relation can be defined in three parts:

- The input set (or sets) **DOMAIN**
- The output set **CODOMAIN**
- The rule which associates a value from the domain to a value in the codomain.

For example:

Domain:	Integers
Codomain:	Integers
Rule:	<code>isLargerThan</code>

Exercises

Is $\{(3, 4), (4, 5), (5, 6)\}$ a relation?

Is $\{(1, 2), (2, 5, 7), (4, 5)\}$ a relation?

Is $\{(1, 1)\}$ a relation?

Write down the domain, codomain and rule for $\{(Eastenders, BBC1)\}$

Write down some other values for $\{(Eastenders, BBC1)\}$

Exercises

Write down the domain and codomain for $\{(1, 3), (2, 4), (3, 5), (5, 3)\}$

Express `myPostcode` as a relation

Make up 3 relations from the 'real' world

Mathematical Entities which can be Defined Using Sets: Functions

Function

A function is simply a collection of **unique** associations between elements of one (or more) sets and elements of another set.

For example:

<code>isEqualTo</code>	associates integers with integers.
<code>myFingerprints</code>	associates a person with their fingerprints.
<code>myKUIDNo</code>	associates students with KUID

Mathematical Entities which can be Defined Using Sets: Functions

Function

A function can be represented by a set of ordered pairs. Since each element of the first set can only be associated with **one** element of the second set, each first set element must appear only once.

For example:

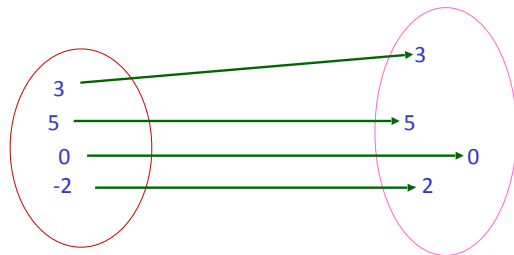
<code>isEqualTo</code>	$\{(3,3),(5,5),(-2,-2),(0,0),...\}$
<code>myKUIDNo</code>	$\{(Fred X, k1239955),(Abdul Y, k6177178),...\}$

Mathematical Entities which can be Defined Using Sets: Functions

Function

A function can be represented pictorially as follows:

`isEqualTo`

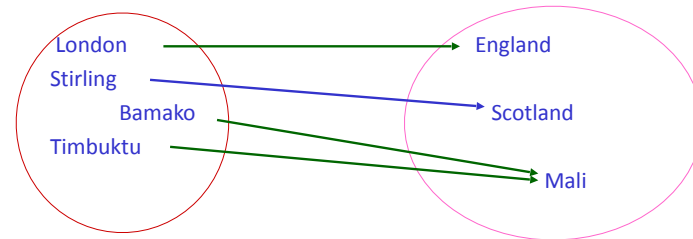


Mathematical Entities which can be Defined Using Sets: Relations

Function

A function can be represented pictorially as follows:

`isATownIn`



Mathematical Entities which can be Defined Using Sets: Functions

Function

A function can be defined in three parts:

- i. The input set (or sets) **DOMAIN**
- ii. The output set **CODOMAIN**
- iii. The rule which associates a value from the domain to a value in the codomain.

For example:

Domain: Integers
Codomain: Integers
Rule: `isEqualTo`

Mathematical Entities which can be Defined Using Sets: Functions

Function

A function can be defined in three parts:

- i. The input set (or sets) **DOMAIN**
- ii. The output set **CODOMAIN**
- iii. The rule which associates a value from the domain to a value in the codomain.

For example:

Domain: Strings
Codomain: Strings
Rule: `isEqualTo`

Mathematical Entities which can be Defined Using Sets: Functions

More examples:

Domain: Towns In Europe
Codomain: Countries in Europe
Rule: `isTownIn`

Domain: Students
Codomain: KU Student ID Numbers
Rule: `hasIDNo`

Domain: Students
Codomain: {Earth, Saturn}
Rule: `livesOnThePlanet`

Mathematical Entities which can be Defined Using Sets: Functions

Function Range

The set of values that a function may return is called the **range** of a function. The range is always a **subset** of the codomain.

For example (given the function definitions from the previous slide):

The range of `isATownIn` is the same as the codomain.
The range of `hasIDNo` is the set of ID numbers of students.
The range of `livesOnThePlanet` is the set {Earth}.

Exercises

Are the following functions or relations?

isAMotherOf	myBreakfastToday
myTelephoneNo	myBrothersName
myPostcode	x^2
myCurrentGPS	$\text{sqrroot } x$
myDNA	myFavTVprogramme
CapitalCities	myDateOfBirth

Exercises

For the 3 relations you wrote down earlier

Are any of them functions?

Write down 3 functions

Functions in SQL

Transform a value or set of values using some rule

Built into the SQL standard

Problem: Microsoft uses many of its own function names to maintain compatibility with Excel and Word functions

Functions Categories

String	concatenation, length, substring
Arithmetic	max, min, power, round, trunc
Date	add, subtract dates
Aggregate or group	average, sum, count

String Function - String Concatenation & Combines fields with additional text if required

`select "Employee "&ename&"
earns £"&sal
from emp`

Expr1000
Employee MARCH earns £18000
Employee BYRNE earns £26000
Employee BELL earns £22500
Employee BIRD earns £39750
Employee AHMAD earns £22500
Employee COX earns £38500
Employee POLLARD earns £34500
Employee REES earns £40000
Employee PARKER earns £60000
Employee TURNER earns £25000
Employee HAYES earns £21000
Employee CASSY earns £19500
Employee GIBSON earns £40000
Employee BLACK earns £23000

String Function - Substring

`mid (string, starting point, no of chars)` - returns part of a string

`select ename, mid (ename, 2, 4)
from emp`

ename	Expr1001
MARCH	ARCH
BYRNE	YRNE
BELL	ELL
BIRD	IRD
AHMAD	HMAD
COX	OX
POLLARD	OLLA
REES	EES
PARKER	ARKE
TURNER	URNE
HAYES	AYES
CASSY	ASSY
GIBSON	IBSO
BLACK	LACK

Arithmetic Function - min

`min ()` - returns smallest value in a column

`select min (sal)
from emp`

Expr1000
18000

SAL
18000
26000
22500
39750
22500
38500
34500
40000
60000
25000
21000
19500
40000
23000

Arithmetic Function - max

`max ()` - returns largest value in a column

`select max (sal)
from emp`

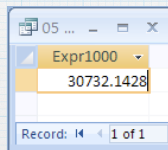
Expr1000
60000

SAL
18000
26000
22500
39750
22500
38500
34500
40000
60000
25000
21000
19500
40000
23000

Aggregate Function - avg

avg () - returns mean value in a column

```
select avg (sal)
from emp
```



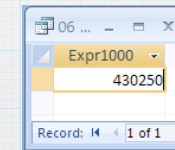
A screenshot of a SQL query result window titled '05 ...'. It shows a single row with the value '30732.1428' in a column labeled 'Expr1000'. The status bar at the bottom indicates 'Record: 14 | 1 of 1'.

SAL
18000
26000
22500
39750
22500
38500
34500
40000
60000
25000
21000
19500
40000
23000

Aggregate Function - sum

sum () - returns total of all values in a column

```
select sum (sal)
from emp
```



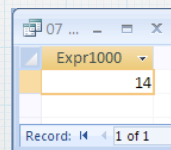
A screenshot of a SQL query result window titled '06 ...'. It shows a single row with the value '430250' in a column labeled 'Expr1000'. The status bar at the bottom indicates 'Record: 14 | 1 of 1'.

SAL
18000
26000
22500
39750
22500
38500
34500
40000
60000
25000
21000
19500
40000
23000

Aggregate Function - count

count () - returns total number of values in a column

```
select count (sal)
from emp
```



A screenshot of a SQL query result window titled '07 ...'. It shows a single row with the value '14' in a column labeled 'Expr1000'. The status bar at the bottom indicates 'Record: 14 | 1 of 1'.

SAL
18000
26000
22500
39750
22500
38500
34500
40000
60000
25000
21000
19500
40000
23000

Exercises Work out the following using SQL

What is the average manager salary?

How many people earn less than 30000?

Whats the lowest admin salary?

Who was the most recently employed person?

Who was the first employed person?

How many people work in London?

What will our total salary cost be?

What are the total salary cost for each department?

Who earns less than the average salary?