Understanding the Data Operations

Satyam Govila

Data Manipulation and Operations

- Manipulation of data is the process of manipulating or changing information to make it more organised and readable.
- Data Manipulation can help us make sure that data which is regularly being added in our database is structured, easily understandable and stored consistently.
- It helps us to create more value and insights from the raw data.

Data Validation Checks

Data validation is the process of ensuring data has undergone data cleansing to ensure they have data quality i.e. proper checks for correctness, meaningfulness, and security of data that are input to the system, through some validation rules.

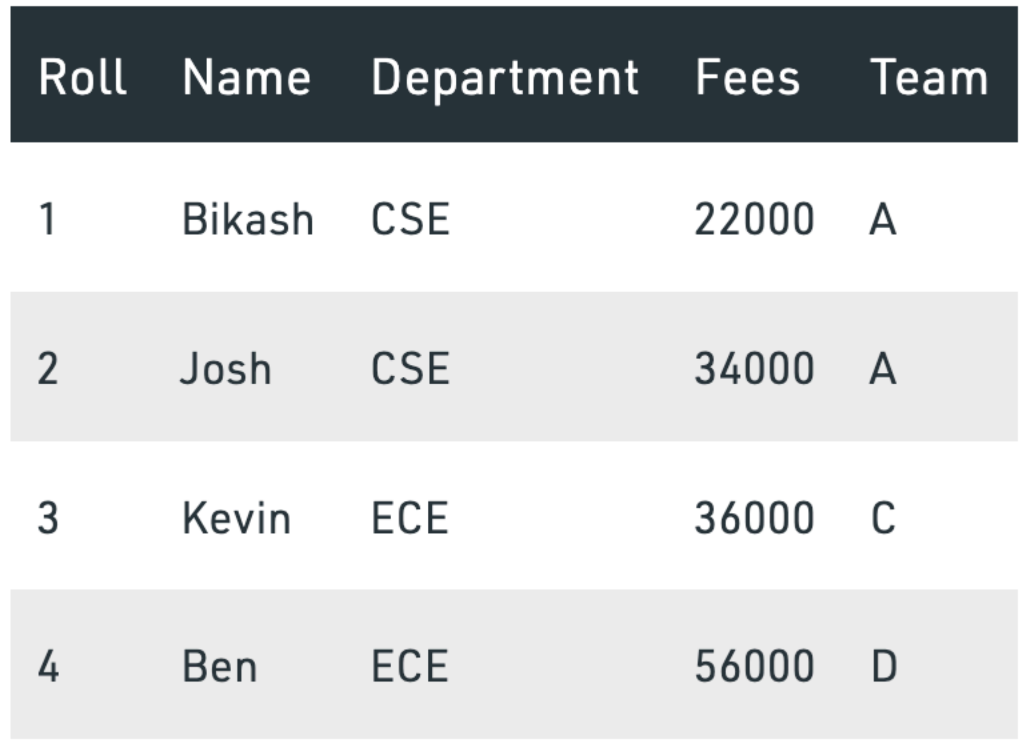
- Data type (ex. integer, float, string)
- Range (ex. A number between 35-40)
- Uniqueness (ex. User id)
- Consistent expressions (ex. Using one of St., Str, Street)
- No null values

Data Operations

- Select
- Filter
- Sort
- Group and Aggregation
- Merge
- Pivot and Unpivot
- Window

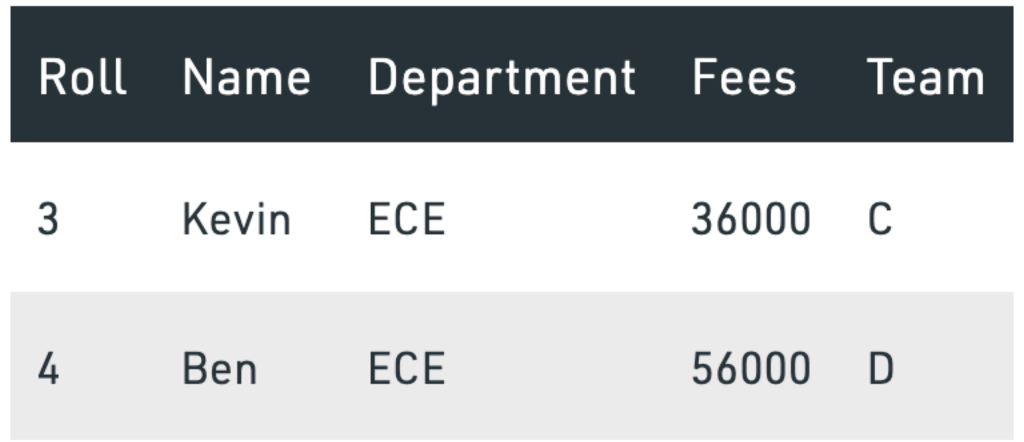
Select Operation

- **Select operation** chooses the subset of tuples from the relation that satisfies the given condition mentioned in the syntax of selection.



Roll	Name	Department	Fees	Team
1	Bikash	CSE	22000	A
2	Josh	CSE	34000	A
3	Kevin	ECE	36000	C
4	Ben	ECE	56000	D

Select all the students of department ECE whose fees is greater then equal to 10000
and belongs to Team other than A.



Roll	Name	Department	Fees	Team
3	Kevin	ECE	36000	C
4	Ben	ECE	56000	D

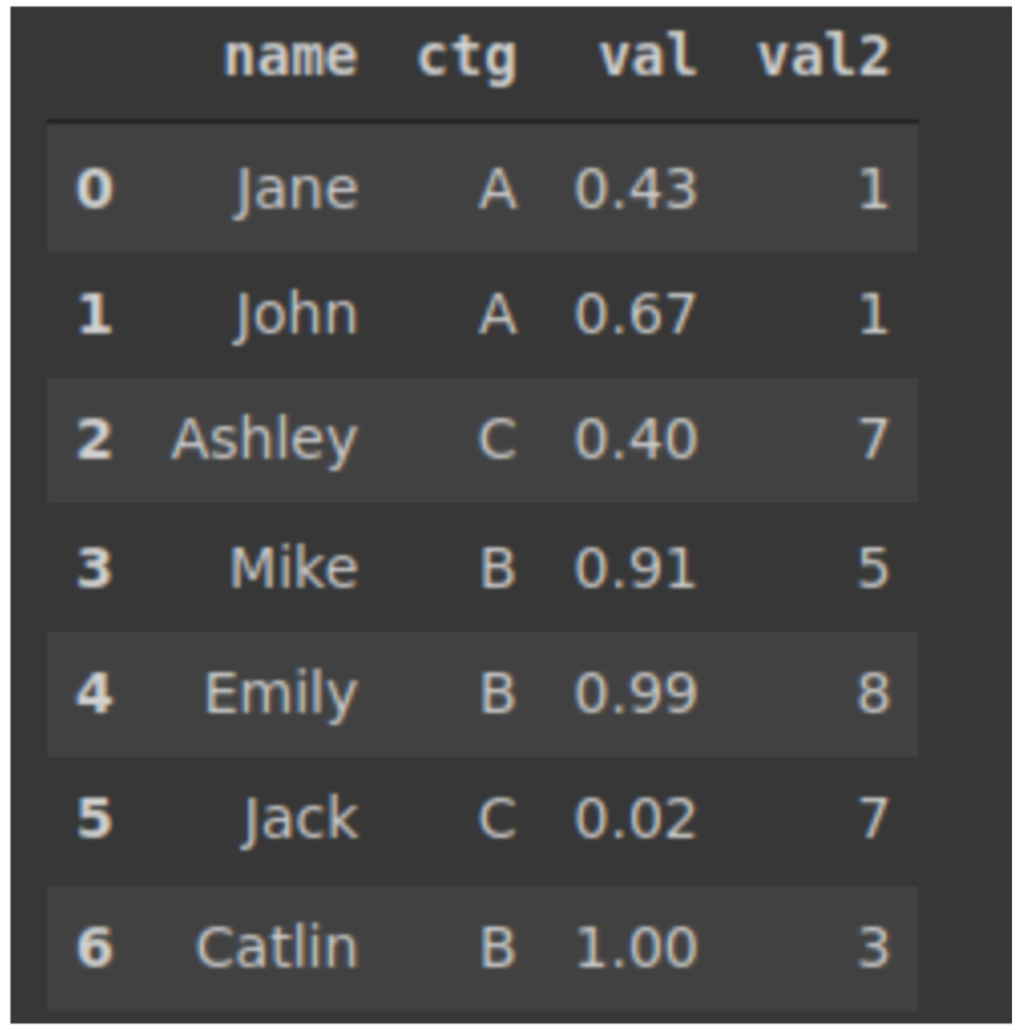
Filter Operation

- **Filter operation** filters the subset of rows,columns from the relation based on a condition or multiple conditions.

- Different filtering operations-

  Filter by rows position and column names
  Selecting multiple values of a column
  Select rows whose column value does not equal a specific value
  Select Non-Missing Data

Filter Operation



	name	ctg	val	val2
0	Jane	A	0.43	1
1	John	A	0.67	1
2	Ashley	C	0.40	7
3	Mike	B	0.91	5
4	Emily	B	0.99	8
5	Jack	C	0.02	7
6	Catlin	B	1.00	3

Rows in which the value in “val” column is greater than 0.5 and “val2” is equal to 1

```
            name        ctg       val        val2
-----------------------------------------------
1           John         A        0.67        1
```

Rows in which the value in “name” column is in the given list of names : ’John','Catlin','Mike'

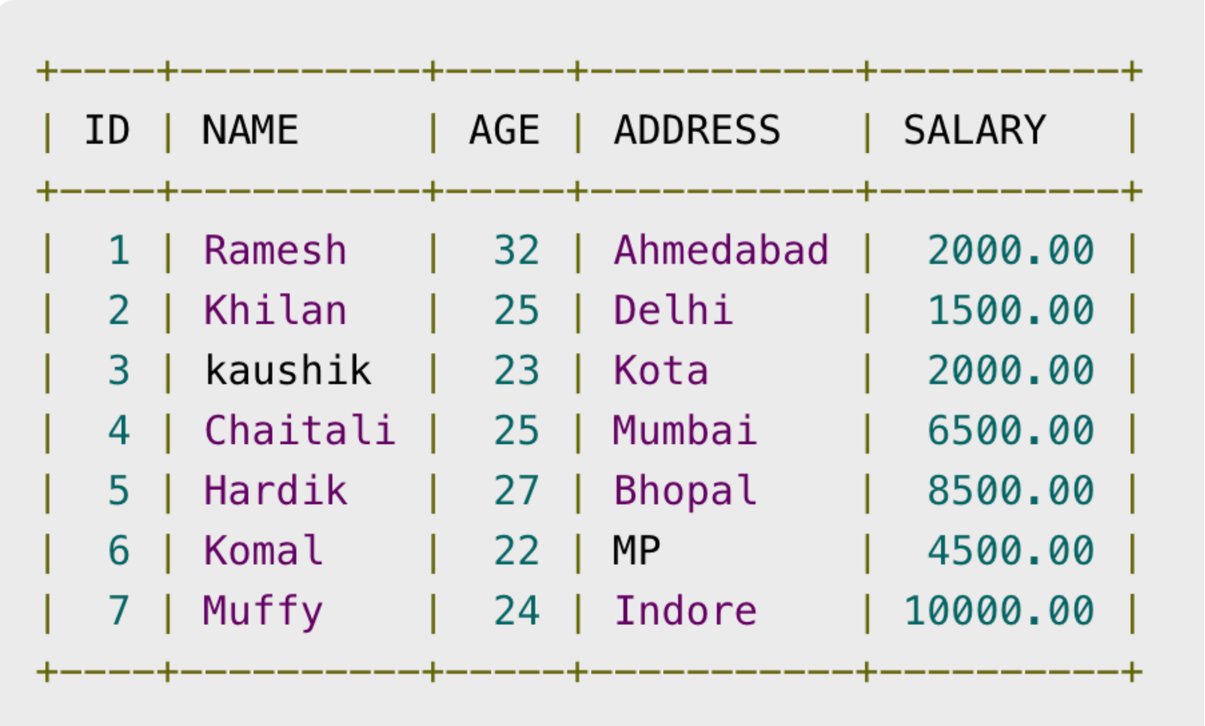
```
            name        ctg       val        val2
-----------------------------------------------
1           John         A        0.67        1
3           Mike         B        0.91        5
6         Catlin         B        1.00        3
```

Rows in which the value in “name” column where name starts with “J”

```
            name        ctg       val        val2
-----------------------------------------------
0           Jane         A        0.43        1
1           John         A        0.67        1
5           Jack         C        0.02        7
```

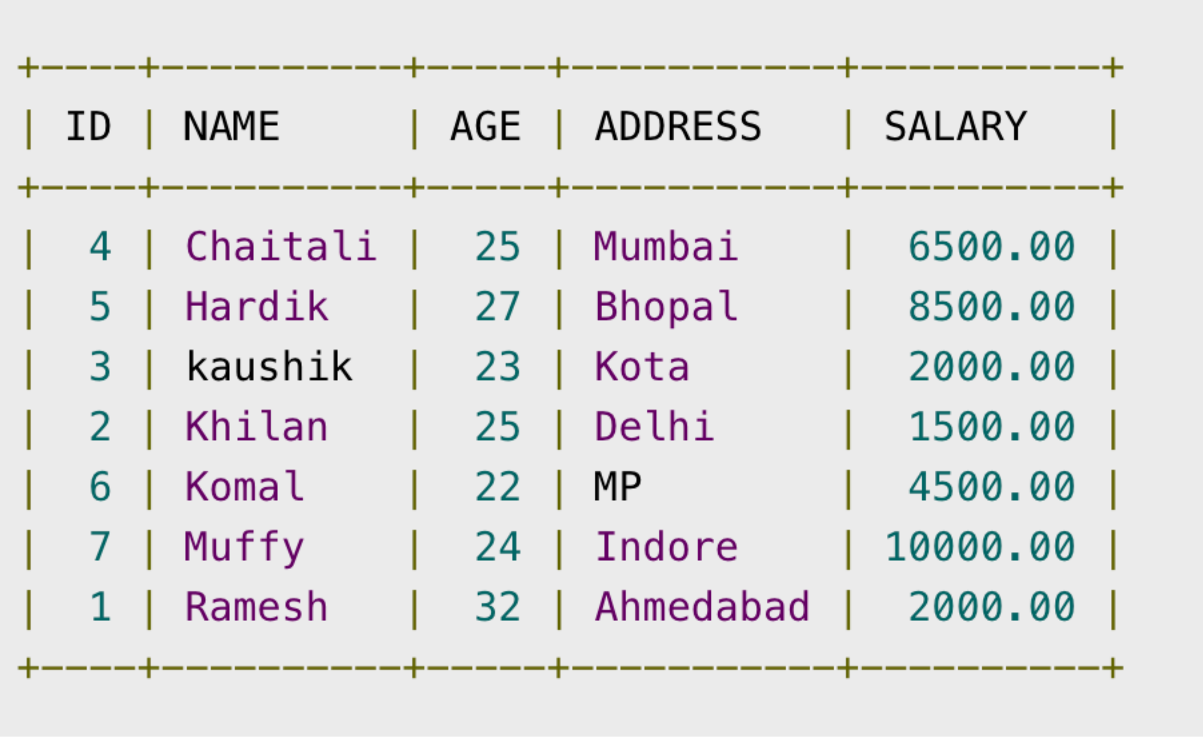
Sort Operation

- **Sort operation** sorts/orders the data in ascending or descending order on a given column(s).



```
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  1 | Ramesh   |  32 | Ahmedabad |  2000.00 |
|  2 | Khilan   |  25 | Delhi     |  1500.00 |
|  3 | kaushik  |  23 | Kota      |  2000.00 |
|  4 | Chaitali |  25 | Mumbai    |  6500.00 |
|  5 | Hardik   |  27 | Bhopal    |  8500.00 |
|  6 | Komal    |  22 | MP        |  4500.00 |
|  7 | Muffy    |  24 | Indore    | 10000.00 |
+----+----------+-----+-----------+----------+
```

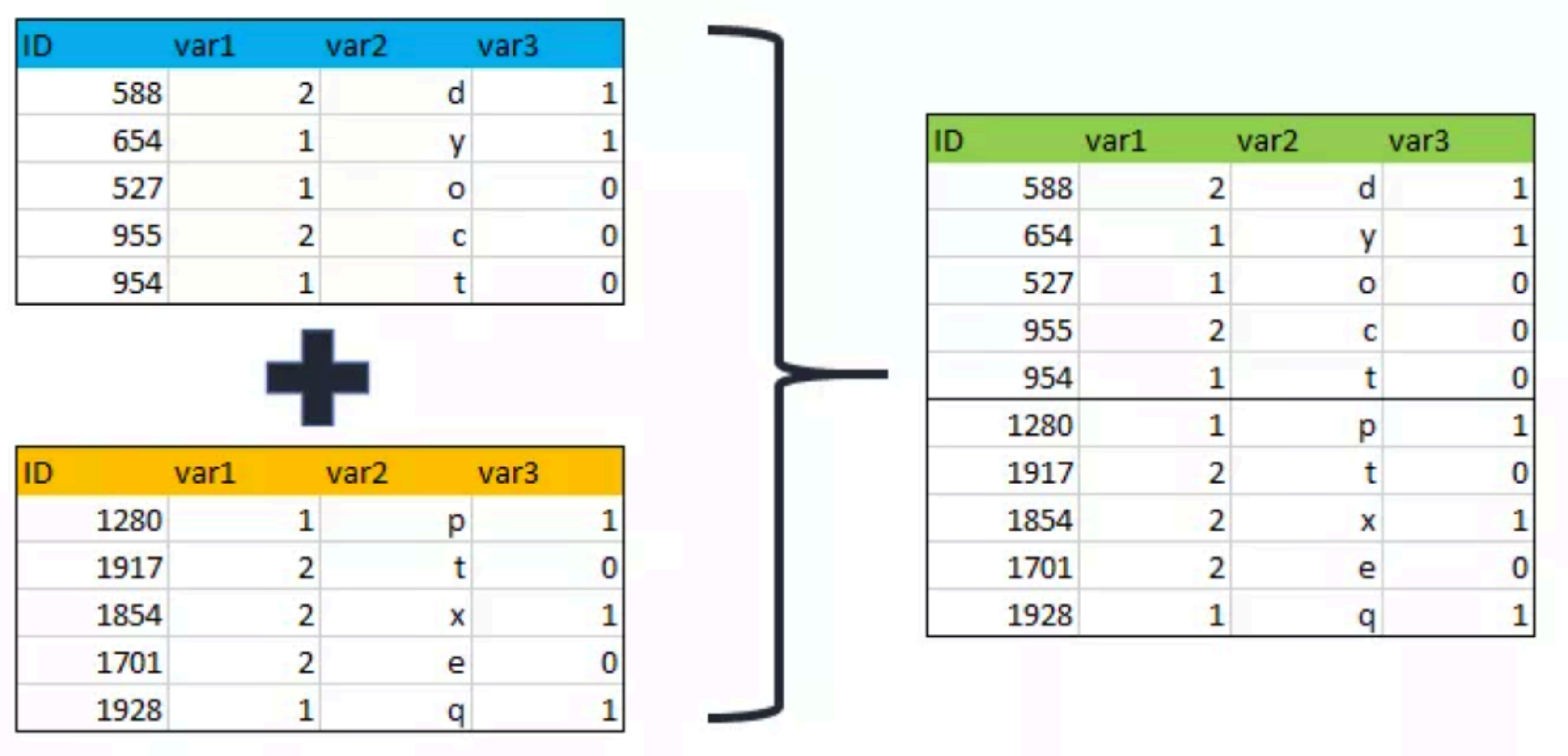
Sort the result in an ascending order by NAME and SALARY.



```
+----+----------+-----+-----------+----------+
| ID | NAME     | AGE | ADDRESS   | SALARY   |
+----+----------+-----+-----------+----------+
|  4 | Chaitali |  25 | Mumbai    |  6500.00 |
|  5 | Hardik   |  27 | Bhopal    |  8500.00 |
|  3 | kaushik  |  23 | Kota      |  2000.00 |
|  2 | Khilan   |  25 | Delhi     |  1500.00 |
|  6 | Komal    |  22 | MP        |  4500.00 |
|  7 | Muffy    |  24 | Indore    | 10000.00 |
|  1 | Ramesh   |  32 | Ahmedabad |  2000.00 |
+----+----------+-----+-----------+----------+
```

Merge Operation

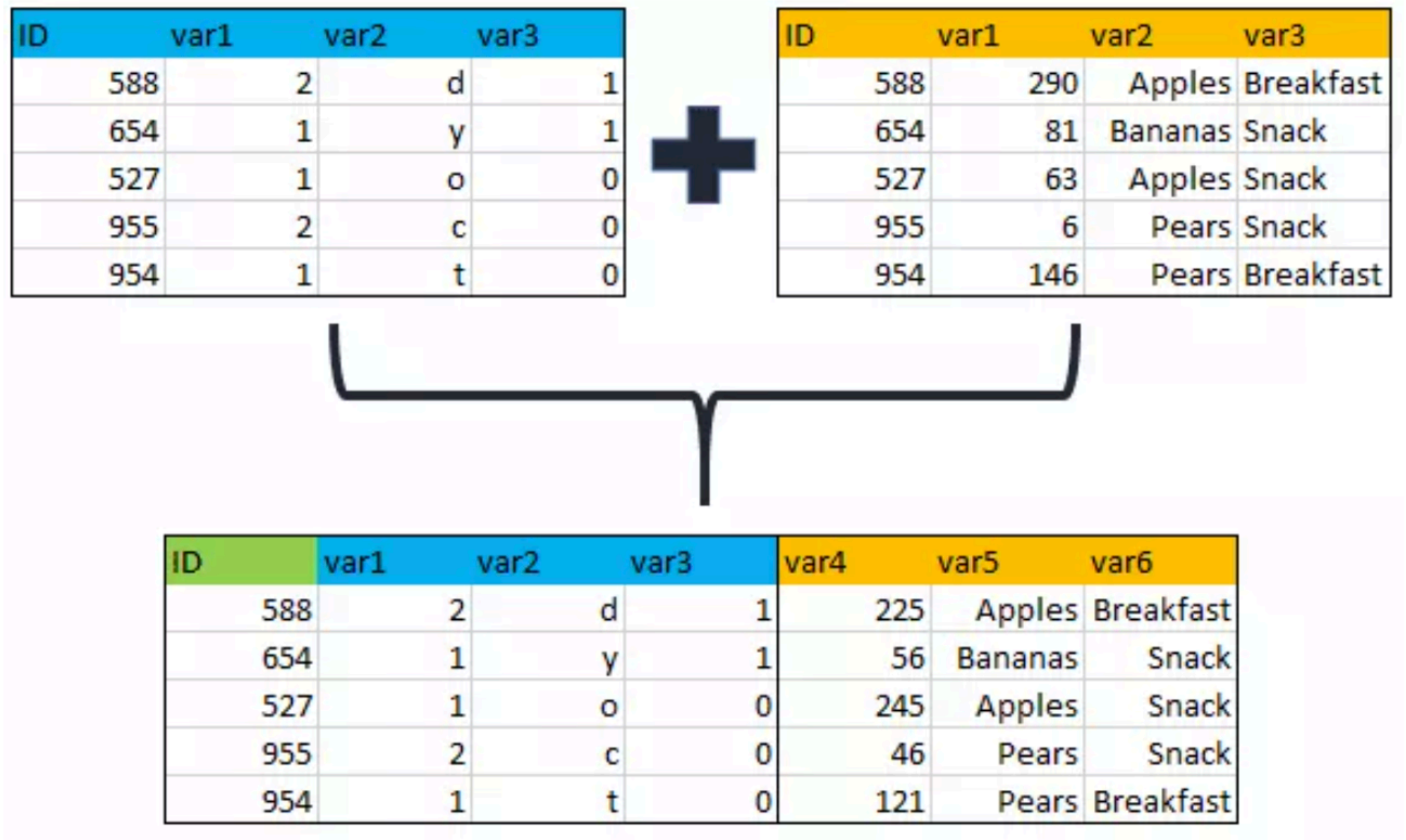
- **Merge operation** combines data from 2 or more tables into as single table based on the given column



ID	var1	var2	var3
588	2	d	1
654	1	y	1
527	1	o	0
955	2	c	0
954	1	t	0

ID	var1	var2	var3
1280	1	p	1
1917	2	t	0
1854	2	x	1
1701	2	e	0
1928	1	q	1

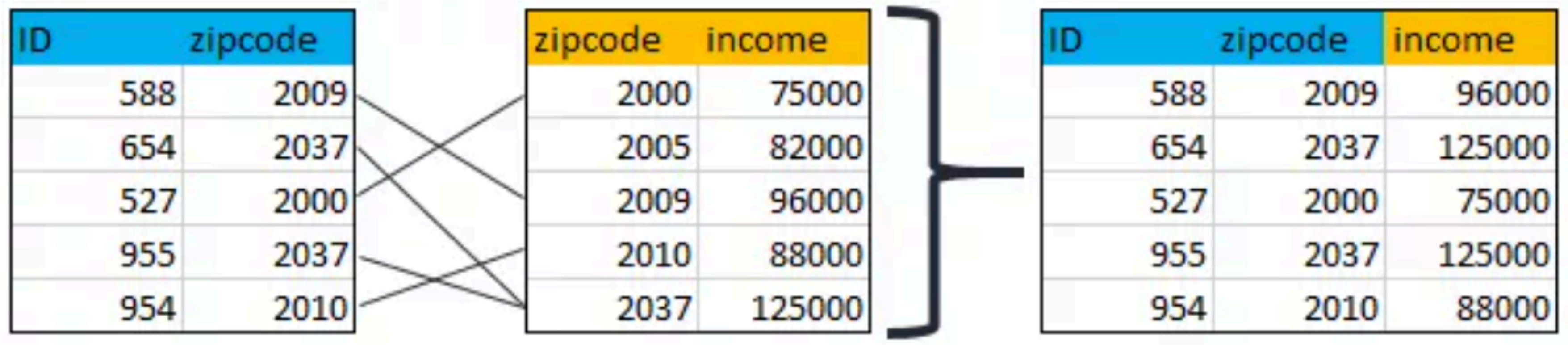
ID	var1	var2	var3
588	2	d	1
654	1	y	1
527	1	o	0
955	2	c	0
954	1	t	0
1280	1	p	1
1917	2	t	0
1854	2	x	1
1701	2	e	0
1928	1	q	1



ID	var1	var2	var3
588	2	d	1
654	1	y	1
527	1	o	0
955	2	c	0
954	1	t	0

ID	var1	var2	var3
588	290	Apples	Breakfast
654	81	Bananas	Snack
527	63	Apples	Snack
955	6	Pears	Snack
954	146	Pears	Breakfast

ID	var1	var2	var3	var4	var5	var6
588	2	d	1	225	Apples	Breakfast
654	1	y	1	56	Bananas	Snack
527	1	o	0	245	Apples	Snack
955	2	c	0	46	Pears	Snack
954	1	t	0	121	Pears	Breakfast

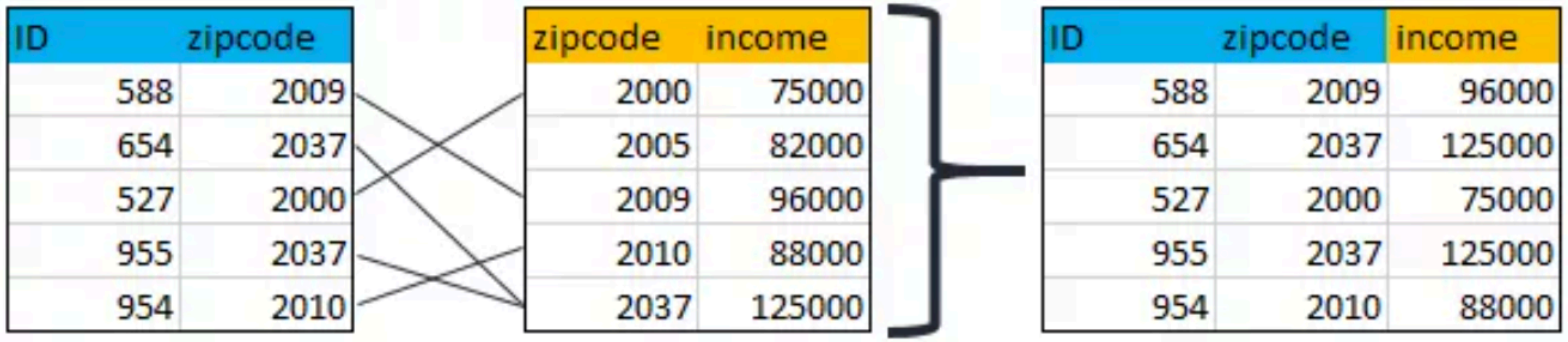


ID	zipcode
588	2009
654	2037
527	2000
955	2037
954	2010

zipcode	income
2000	75000
2005	82000
2009	96000
2010	88000
2037	125000

ID	zipcode	income
588	2009	96000
654	2037	125000
527	2000	75000
955	2037	125000
954	2010	88000

Merge Operation



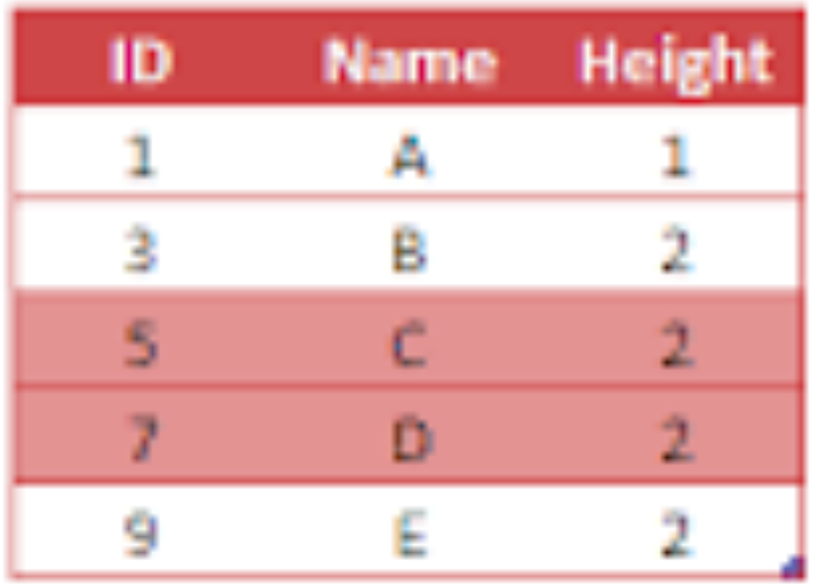
ID	zipcode
588	2009
654	2037
527	2000
955	2037
954	2010

zipcode	income
2000	75000
2005	82000
2009	96000
2010	88000
2037	125000

ID	zipcode	income
588	2009	96000
654	2037	125000
527	2000	75000
955	2037	125000
954	2010	88000

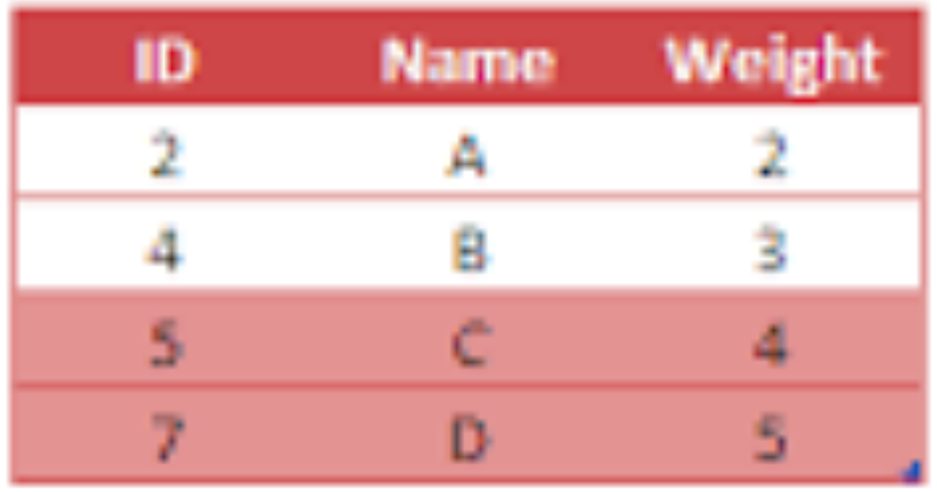
Merge Operation

Dataset - A



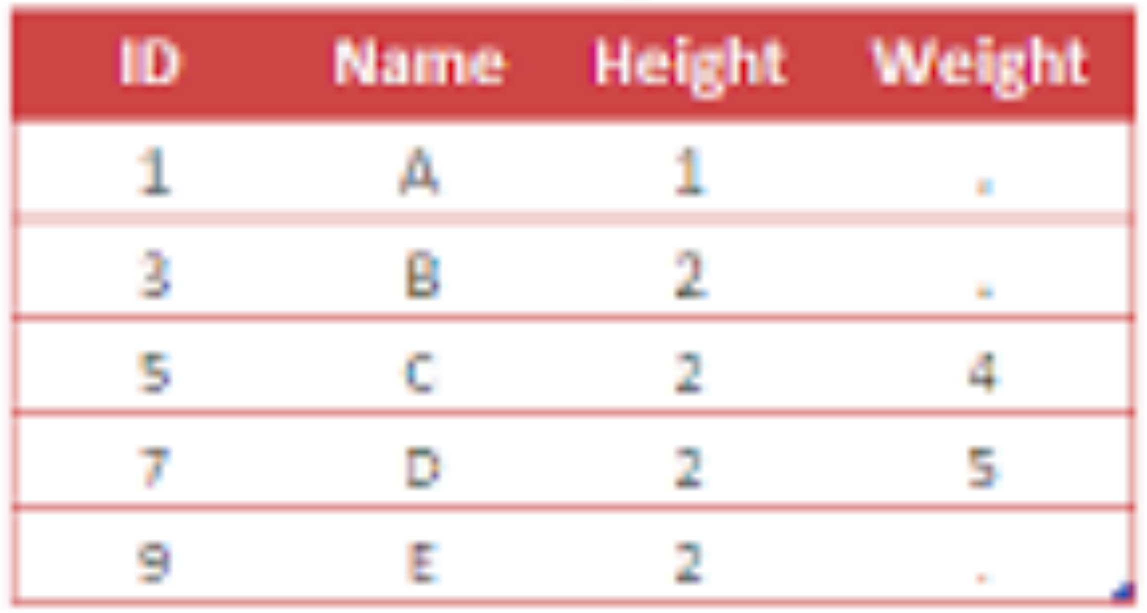
ID	Name	Height
1	A	1
3	B	2
5	C	2
7	D	2
9	E	2

Dataset - B



ID	Name	Weight
2	A	2
4	B	3
5	C	4
7	D	5

Left Join : Merged Dataset



ID	Name	Height	Weight
1	A	1	.
3	B	2	.
5	C	2	4
7	D	2	5
9	E	2	.

Pivot and Unpivot

- Pivot Table is used to summarise, sort, reorganise, group, count, total or average data stored in a table.
- It allows us to transform columns into rows and rows into columns.
- It allows grouping by any field (column), and using advanced calculations on them.

Pivot and Unpivot

Employee	Date and Time	Pizza	Total
Melissa	2019/05/26 01:17PM	Margherita	$6.03
Sylvia	2019/05/27 01:19PM	Quattro Stagioni	$6.74
Juliette	2019/05/28 02:23PM	Salami	$6.38
Melissa	2019/05/29 02:36PM	Tuna	$6.91
Sylvia	2019/06/01 02:41PM	Margherita	$6.03
Juliette	2019/06/10 02:49PM	Quattro Stagioni	$6.74
Melissa	2019/06/11 02:57PM	Salami	$6.38
Sylvia	2019/06/12 03:01PM	Tuna	$6.91
Juliette	2019/06/26 03:02PM	Margherita	$6.03
Sylvia	2019/07/16 03:11PM	Quattro Stagioni	$6.74
Juliette	2019/07/17 03:26PM	Salami	$6.38
Melissa	2019/07/18 03:28PM	Tuna	$6.91
Sylvia	2019/07/19 03:31PM	Quattro Stagioni	$6.74

Questions to answer

Do you have an idea what questions we could ask about our pizza receipts? What useful information we could get?

- Who sold how many pizzas?
- Which type of pizza was sold how many times?
- Who generated what revenue (total value of pizzas sold)?
- What pizza generated what revenue?

Answers to such questions can help us decide what pizza flavours to drop and what flavours we could try to promote more.

Or it can help us to set employee bonuses.

There are even more advanced questions to answer:

- What type of pizzas are sold most in the given month or season?
- What type of pizzas are better sold in the morning and in the afternoon?

Pivot and Unpivot

Who sold how many pizzas?

The *Row Label* is Employee. The *Summation Value* can be anything like the Pizza name.

Employee	Pizzas Count
Melissa	4
Sylvia	5
Juliette	4

Which type of pizza was sold how many times?

The *Row Label* is Pizza. The *Summation Value* can be anything like the Pizza name.

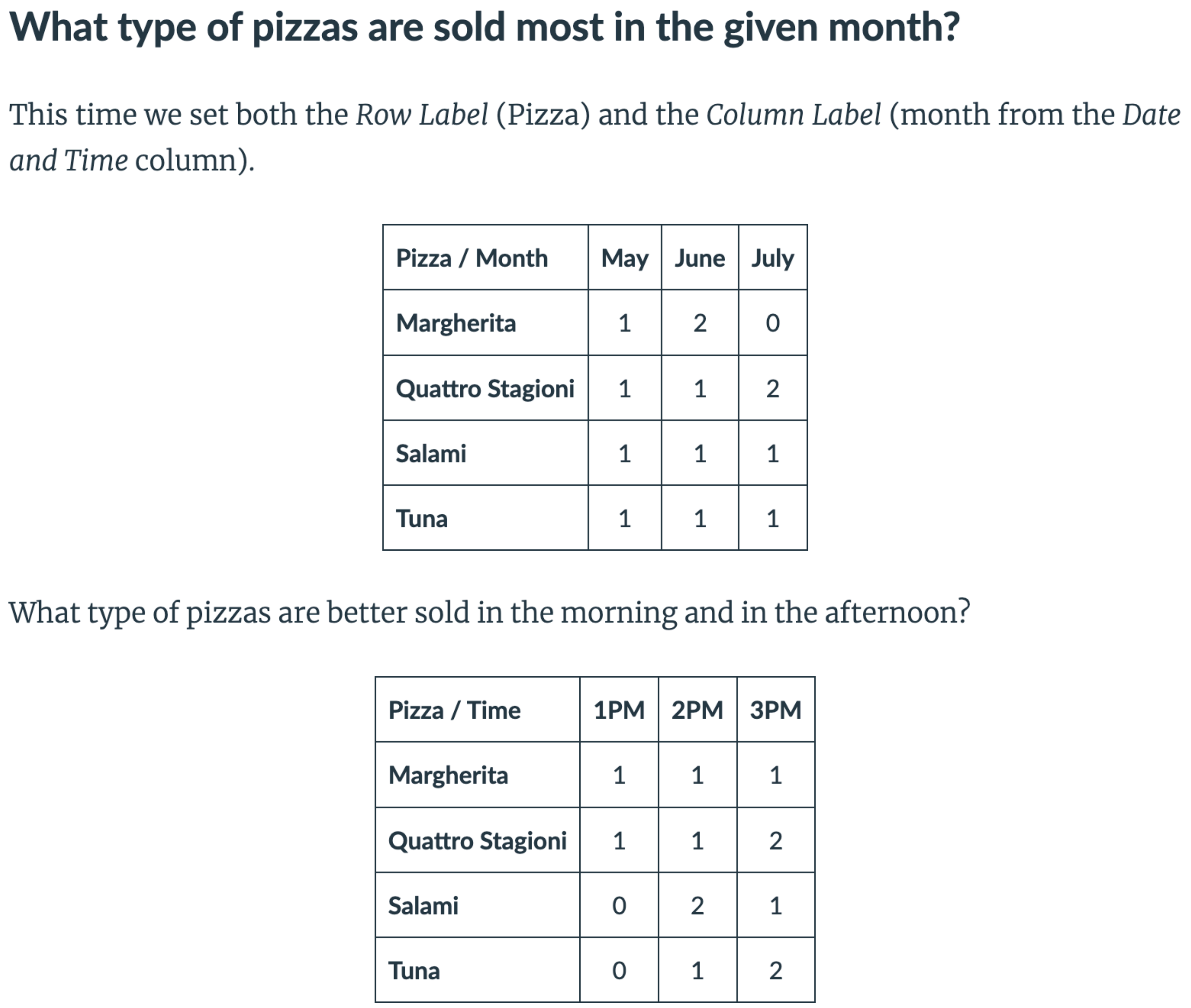
Pizza	Pizzas Count
Margherita	3
Quattro Stagioni	4
Salami	3
Tuna	3

What pizza generated what revenue?

The *Row Label* is Pizza. The *Summation Value* is still the sum of the Total column. We can also add a column summary.

Pizza	Sum of Total
Margherita	$18.09
Quattro Stagioni	$26.96
Salami	$19.14
Tuna	$20.73
Grand Total	**$84.92**

Pivot and Unpivot



What type of pizzas are sold most in the given month?

This time we set both the *Row Label* (Pizza) and the *Column Label* (month from the *Date and Time* column).

Pizza / Month	May	June	July
Margherita	1	2	0
Quattro Stagioni	1	1	2
Salami	1	1	1
Tuna	1	1	1

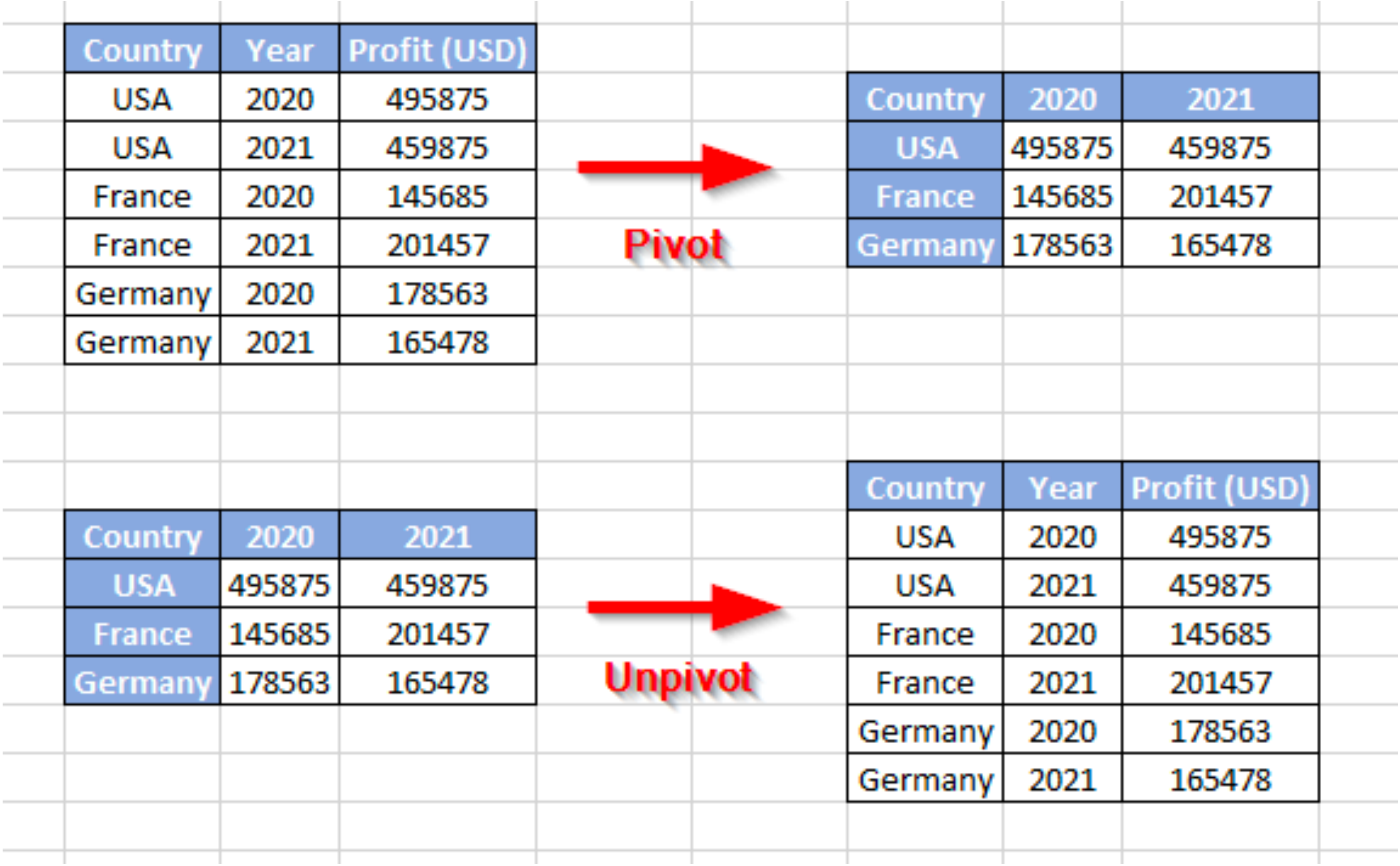
What type of pizzas are better sold in the morning and in the afternoon?

Pizza / Time	1PM	2PM	3PM
Margherita	1	1	1
Quattro Stagioni	1	1	2
Salami	0	2	1
Tuna	0	1	2

Employee	Pizza / Month	May	June	July
Melissa	Margherita	1	0	0
	Quattro Stagioni	0	0	0
	Salami	0	1	0
	Tuna	1	0	1
Sylvia	Margherita	0	1	0
	Quattro Stagioni	1	0	2
	Salami	0	0	0
	Tuna	0	1	0
Juliette	Margherita	0	1	0
	Quattro Stagioni	0	1	0
	Salami	1	0	1
	Tuna	0	0	0

Unpivot

- **Unpivot** operator does the opposite that is it transform the column based data into rows.



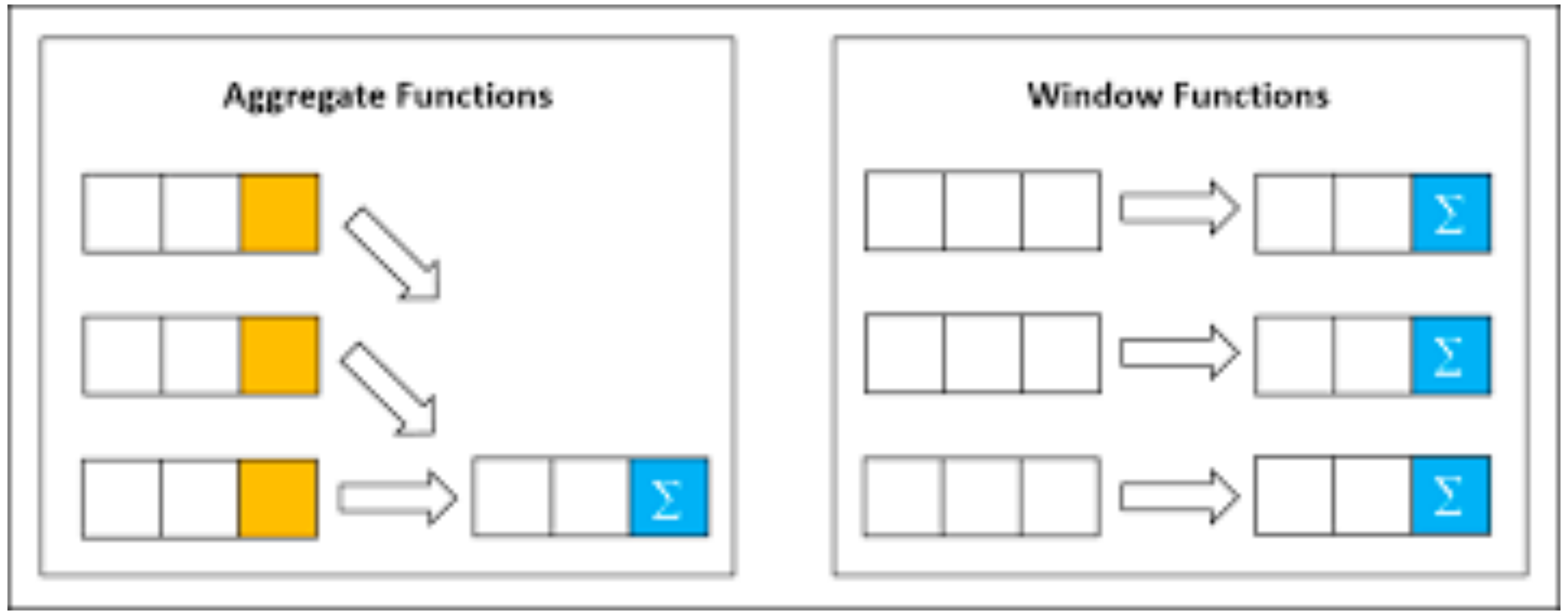
Country	Year	Profit (USD)
USA	2020	495875
USA	2021	459875
France	2020	145685
France	2021	201457
Germany	2020	178563
Germany	2021	165478

Country	2020	2021
USA	495875	459875
France	145685	201457
Germany	178563	165478

Country	2020	2021
USA	495875	459875
France	145685	201457
Germany	178563	165478

Country	Year	Profit (USD)
USA	2020	495875
USA	2021	459875
France	2020	145685
France	2021	201457
Germany	2020	178563
Germany	2021	165478

Aggregate and Window Functions



- Employee (id , Name , department , salary)
- 100 , A , Sales , 100000
- 101 , B , IT , 120000
- 102 , C , Sales , 200000
- What is the max salary in Employee table ? => Aggregate function (SUM())
- What is the max salary in each department ? => Window Function
- Ans - Sales-200000,It-120000