

# Matrices in R: Takeaways

by Dataquest Labs, Inc. - All rights reserved © 2021

## Syntax

---

### CREATING A MATRIX

- Restructure a vector into a matrix using the `matrix()` function:

```
matrix_1 <- matrix(1:12, ncol = 4, nrow = 3)
matrix_2 <- matrix(1:12, ncol = 4)
matrix_3 <- matrix(1:12, nrow = 3)
```

- Combine Vectors or Matrices by Row

```
matrix_4 <- rbind(matrix_1, matrix_2)
matrix_5 <- rbind(vector_1, vector_2)
matrix_6 <- rbind(vector_1, matrix_1)
```

- Combine Vectors or Matrices by Column

```
matrix_7 <- cbind(matrix_1, matrix_2)
matrix_8 <- cbind(vector_1, vector_2)
matrix_9 <- cbind(vector_1, matrix_1)
```

---

### INDEXING MATRICES BY ELEMENT

- Extract a single element:

```
matrix[2, 5]
matrix["Stanford", "patents"]
```

- Extract multiple elements:

```
matrix[c(1,2),c(1,3)]
matrix[c("Harvard", "Stanford"),c("world_rank", "influence")]
matrix[tuition >= 45000, "patents"]
```

---

### INDEXING MATRICES BY ROWS AND COLUMNS

- Extract a single row:

```
matrix[1,]
matrix["Harvard",]
```

- Extract a single column:

```
matrix[,2]
matrix[, "quality_of_education"]
```

- Extract multiple rows or columns:

```
matrix[,c("quality_of_education", "influence", "broad_impact")]
matrix[,c("2,3,4")]
```

---

## CALCULATE THE SUM/MEAN OF VALUES IN A VECTOR OR MATRIX

- Sum/Mean of values in a vector:

```
sum(vector)
mean(vector)
```

- Sum/Mean of values in a matrix:

```
sum(matrix[, "column"])
mean(matrix["row", ])
```

- Sum/Mean of values in a matrix by column:

```
colSums(matrix)
colMeans(matrix)
```

- Sum/Mean of values in a matrix by row:

```
rowSums(matrix)
rowMeans(matrix)
```

---

## RANK VALUES OF A VECTOR OR SUBSET OF A MATRIX

- Rank values of a vector:

```
rank(vector)
```

- Rank values of a matrix:

```
rank(matrix[, "column"])
rank(matrix["row", ])
```

---

## NAMING MATRIX ROWS AND COLUMNS

- Assign name attributes to rows of a matrix:

```
rownames(matrix) <- vector_row_names
```

- Assign name attributes to columns of a matrix:

```
colnames(matrix) <- vector_column_names
```

---

## REMOVING ROWS AND COLUMNS FROM A MATRIX

- Removing a single element:

```
matrix[-1, -2]
```

- Removing multiple elements:

```
matrix[c(-2, -5, -7), ]
```

## Concepts

- The four data structures covered in this course are:
  - Vector: one-dimensional structure for storing values of SAME TYPE.

- **Matrix: two-dimensional structure for storing values of SAME TYPE.**
- Lists: multi-dimensional structure for storing values of ANY DATA TYPE/OBJECT.
- Dataframe: two-dimensional structure for storing values of ANY DATA TYPE/OBJECT.

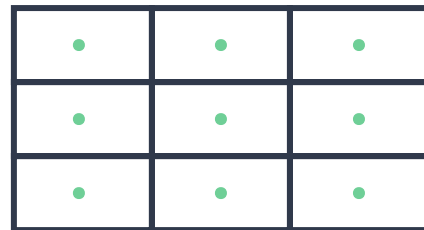
## Vector

1 Dimension | Same Data Type



## Matrix

2 Dimensions | Same Data Type



## List

Several Dimensions | Any Data Type



## Dataframe

2 Dimensions | Any Data Type



- R is a **1-indexed** programming language, which means that the first element in a matrix is assigned a position of one.

### Indexing a Row

```
matrix[row_indice,]
```

	1	2	3
1	1	2	3
2	4	5	6
3	7	8	9
4	10	11	12
5	13	14	15

### Indexing a Column

```
matrix[,column_indice]
```

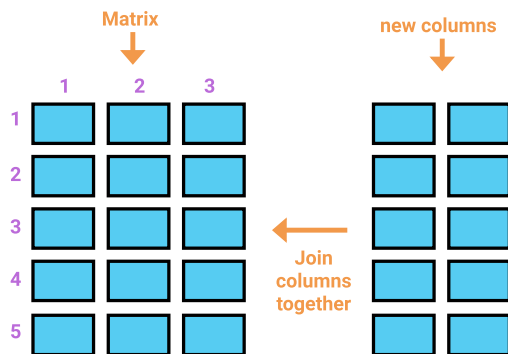
	1	2	3
1	1	2	3
2	4	5	6
3	7	8	9
4	10	11	12
5	13	14	15

### Indexing a specific value

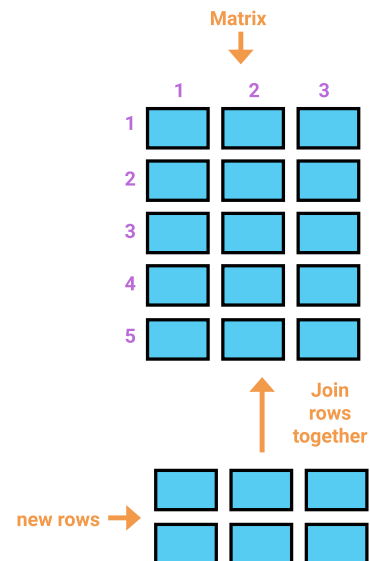
```
matrix[row, column]
```

	1	2	3
1	1	2	3
2	4	5	6
3	7	8	9
4	10	11	12
5	13	14	15

- How to combine vectors or matrices in R.

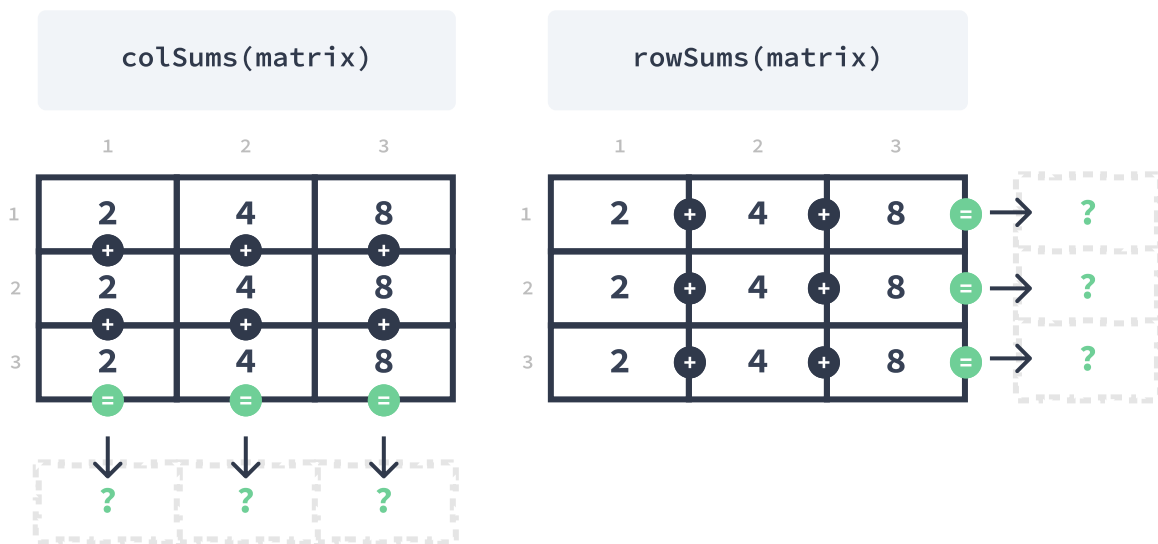


APPENDING NEW COLUMNS (VECTOR/MATRIX) TO A MATRIX  
`cbind(matrix, new_columns)`



APPENDING NEW ROWS (VECTOR/MATRIX) TO A MATRIX  
`rbind(matrix, new_rows)`

- How to aggregate elements by columns/rows in R over a two-dimensional data structure.



## Resources

- [Documentation on indexing matrices in R](#)