EXPLORE AI
ACADEMY

An introduction to using data

# Useful formulas and where to use them

“One cannot escape the feeling that these **mathematical formulas have an independent existence and an intelligence of their own,** that they are wiser than we are, wiser even than their discoverers ...”

– Heinrich Hertz

# Why should I learn this?

Trying to organise and analyse spreadsheets can feel daunting if you're unfamiliar with **formulas** and when to use them.

## 01. Increased efficiency

- Formulas allow for **more efficient** data analysis and complex calculations by simplifying the way we process data.
- **Result**: A cleaner, easier-to-read spreadsheet. Useful for presentations.

## 02. Increased productivity

- Formulas allow us to **capture and store working solutions**.
- **Result**: A reusable solution with perfect accuracy every time, and get work done faster.

## 03. Makes you stand out

- Being good at formulas can **add** to your **value**.
- **Result**: Being comfortable with formulas will make you a vital asset to companies that rely on spreadsheets to make data-driven decisions.

# Data overview

The gender pay gap dataset contains 20 rows.
We will focus on the following columns:

**A.** **Age**

The age of an employee.

**B.** **Base pay**

The annual salary of an employee excluding any other compensations.

**C.** **Bonus**

Money given to employees beyond their base pay every year.

## The dataset

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | **Gender** | **JobTitle** | **Age** | **Education** | **BasePay** | **Bonus** |
| 2 | Female | Graphic Designer | 18 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | PhD | 90208.00 | 9268.00 |
| 5 | Male | Software Engineer | 20 | Masters | 108080.00 | 10154.00 |
| 6 | Male | Graphic Designer | 26 | Masters | 99464.00 | 9319.00 |
| 7 | Female | IT | 20 | PhD | 70890.00 | 10126.00 |
| 8 | Female | Graphic Designer | 20 | College | 67585.00 | 10541.00 |
| 9 | Male | Software Engineer | 18 | PhD | 97523.00 | 10240.00 |
| 10 | Female | Graphic Designer | 33 | High School | 112976.00 | 9836.00 |
| 11 | Female | Sales Associate | 35 | College | 106524.00 | 9941.00 |
| 12 | Male | Graphic Designer | 24 | PhD | 102261.00 | 10212.00 |
| 13 | Female | Driver | 18 | College | 62759.00 | 10124.00 |
| 14 | Female | Financial Analyst | 19 | College | 84007.00 | 8990.00 |

# Relative and absolute cell references

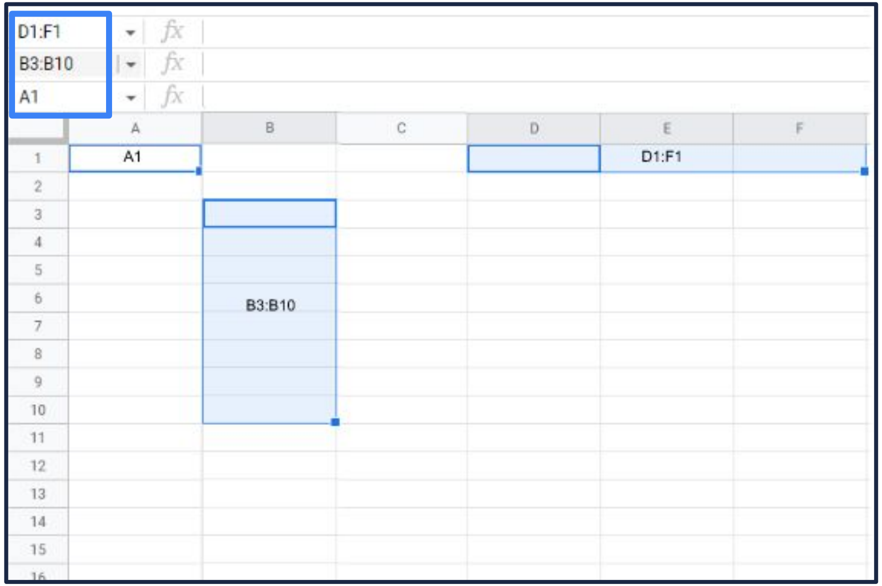
A reference is a cell address that identifies a cell or range of cells by referring to the **column letter** and **row number** of the cell or cells.

**Ranges** are referenced using **two cell references** separated by a **colon** and can span multiple rows and columns.

Cell referencing is important because it:

- Allows formulas to **automatically update** when specific **cell values change**.
- Helps **formulas update** when cells are **copied or moved**.

We can use **relative** references, **absolute** references, or a mixture of the two.



# Relative and absolute cell references

## Relative references:

- Relative references **change when copied** across multiple cells based on the relative position of rows and columns.
- For example, if the formula **=A1+B1** is copied from row 1 to row 2, the formula in row 2 will become **=A2+B2**.
- All cell references are relative references by **default**.

**01.** Enter the formula **=C2*2** on cell D2.

**02.** Replicate the formula to other rows by dragging down the fill handle.

D2 fx =C2*2

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Relative | Education | BasePay | Bonus |
| 2 | Female | Graphic Designer | 18 | 36 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | | PhD | 90208.00 | 9268.00 |

D10 fx =C10*2

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Relative | Education | BasePay | Bonus |
| 2 | Female | Graphic Designer | 18 | 36 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | 42 | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | 38 | PhD | 90208.00 | 9268.00 |
| 5 | Male | Software Engineer | 20 | 40 | Masters | 108080.00 | 10154.00 |
| 6 | Male | Graphic Designer | 26 | 52 | Masters | 99464.00 | 9319.00 |
| 7 | Female | IT | 20 | 40 | PhD | 70890.00 | 10126.00 |
| 8 | Female | Graphic Designer | 20 | 40 | College | 67585.00 | 10541.00 |
| 9 | Male | Software Engineer | 18 | 36 | PhD | 97523.00 | 10240.00 |
| 10 | Female | Graphic Designer | 33 | 66 | High School | 112976.00 | 9836.00 |
| 11 | Female | Sales Associate | 35 | 70 | College | 106524.00 | 9941.00 |
| 12 | Male | Graphic Designer | 24 | 48 | PhD | 102261.00 | 10212.00 |
| 13 | Female | Driver | 18 | 36 | College | 62759.00 | 10124.00 |
| 14 | Female | Financial Analyst | 19 | 38 | College | 84007.00 | 8990.00 |
| 15 | Female | Warehouse Associate | 30 | 60 | Masters | 86220.00 | 9583.00 |

# Relative and absolute cell references

## Absolute references:

- Absolute references **keep a row and/or column constant** by using the **dollar sign ($)** before the row number and/or column letter. For example, if the formula **=$A$1+B1** is copied from row 1 to row 2, the formula in row 2 will become **=$A$1+B2**.
- How to use:
  - **$A$1** – The column and row do not change.
  - **$A1** – The column does not change.
  - **A$1** – The row does not change.

**01.** Enter the formula **=$C$2*2** on cell D2.

**02.** Replicate the formula to other rows by dragging down the fill handle.

D2 | fx =$C$2*2

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Absolute | Education | BasePay | Bonus |
| 2 | Female | Graphic Designer | 18 | 36 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | | PhD | 90208.00 | 9268.00 |

D10 | fx =$C$2*2

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Absolute | Education | BasePay | Bonus |
| 2 | Female | Graphic Designer | 18 | 36 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | 36 | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | 36 | PhD | 90208.00 | 9268.00 |
| 5 | Male | Software Engineer | 20 | 36 | Masters | 108080.00 | 10154.00 |
| 6 | Male | Graphic Designer | 26 | 36 | Masters | 99464.00 | 9319.00 |
| 7 | Female | IT | 20 | 36 | PhD | 70890.00 | 10126.00 |
| 8 | Female | Graphic Designer | 20 | 36 | College | 67585.00 | 10541.00 |
| 9 | Male | Software Engineer | 18 | 36 | PhD | 97523.00 | 10240.00 |
| 10 | Female | Graphic Designer | 33 | 36 | High School | 112976.00 | 9836.00 |
| 11 | Female | Sales Associate | 35 | 36 | College | 106524.00 | 9941.00 |
| 12 | Male | Graphic Designer | 24 | 36 | PhD | 102261.00 | 10212.00 |
| 13 | Female | Driver | 18 | 36 | College | 62759.00 | 10124.00 |
| 14 | Female | Financial Analyst | 19 | 36 | College | 84007.00 | 8990.00 |
| 15 | Female | Warehouse Associate | 30 | 36 | Masters | 86220.00 | 9583.00 |

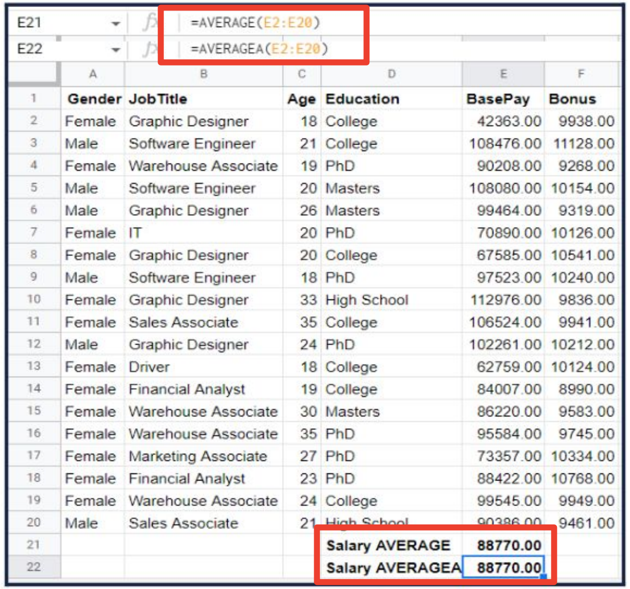
# The AVERAGE and AVERAGEA functions

Used to find the **arithmetic mean** for a list of arguments, i.e., the sum of the values in the arguments divided by the number of arguments. They differ in how they evaluate boolean and text values.

```
=AVERAGE(value1, [value2, …])
=AVERAGEA(value1, [value2, …])
```

Arguments can take the form of:

- A list of numbers with the maximum number of entries allowed being 30:
  `=AVERAGE(13,24,107)` or `=AVERAGEA(13,24,107)`
- Cell references:
  `=AVERAGE(B2,H17,G9)` or `=AVERAGEA(B2,H17,G9)`
- Range of cells:
  `=AVERAGE(B2:B24)` or `=AVERAGEA(B2:B24)`



E21 =AVERAGE(E2:E20)
E22 =AVERAGEA(E2:E20)

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Education | BasePay | Bonus |
| 2 | Female | Graphic Designer | 18 | College | 42363.00 | 9938.00 |
| 3 | Male | Software Engineer | 21 | College | 108476.00 | 11128.00 |
| 4 | Female | Warehouse Associate | 19 | PhD | 90208.00 | 9268.00 |
| 5 | Male | Software Engineer | 20 | Masters | 108080.00 | 10154.00 |
| 6 | Male | Graphic Designer | 26 | Masters | 99464.00 | 9319.00 |
| 7 | Female | IT | 20 | PhD | 70890.00 | 10126.00 |
| 8 | Female | Graphic Designer | 20 | College | 67585.00 | 10541.00 |
| 9 | Male | Software Engineer | 18 | PhD | 97523.00 | 10240.00 |
| 10 | Female | Graphic Designer | 33 | High School | 112976.00 | 9836.00 |
| 11 | Female | Sales Associate | 35 | College | 106524.00 | 9941.00 |
| 12 | Male | Graphic Designer | 24 | PhD | 102261.00 | 10212.00 |
| 13 | Female | Driver | 18 | College | 62759.00 | 10124.00 |
| 14 | Female | Financial Analyst | 19 | College | 84007.00 | 8990.00 |
| 15 | Female | Warehouse Associate | 30 | Masters | 86220.00 | 9583.00 |
| 16 | Female | Warehouse Associate | 35 | PhD | 95584.00 | 9745.00 |
| 17 | Female | Marketing Associate | 27 | PhD | 73357.00 | 10334.00 |
| 18 | Female | Financial Analyst | 23 | PhD | 88422.00 | 10768.00 |
| 19 | Female | Warehouse Associate | 24 | College | 99545.00 | 9949.00 |
| 20 | Male | Sales Associate | 21 | High School | 90386.00 | 9461.00 |
| 21 | | | | Salary AVERAGE | 88770.00 | |
| 22 | | | | Salary AVERAGEA | 88770.00 | |

# The AVERAGE and AVERAGEA functions

F2 | fx =E2-E21

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Gender | JobTitle | Age | Education | BasePay | Av. BasPay Disparity |
| 2 | Female | Graphic Designer | 18 | College | 42363.00 | -46407.00 |
| 3 | Male | Software Engineer | 21 | College | 108476.00 | 19706.00 |
| 4 | Female | Warehouse Associate | 19 | PhD | 90208.00 | 1438.00 |
| 5 | Male | Software Engineer | 20 | Masters | 108080.00 | 19310.00 |
| 6 | Male | Graphic Designer | 26 | Masters | 99464.00 | 10694.00 |
| 7 | Female | IT | 20 | PhD | 70890.00 | -17880.00 |
| 8 | Female | Graphic Designer | 20 | College | 67585.00 | -21185.00 |
| 9 | Male | Software Engineer | 18 | PhD | 97523.00 | -21185.00 |
| 10 | Female | Graphic Designer | 33 | High School | 112976.00 | 24206.00 |
| 11 | Female | Sales Associate | 35 | College | 106524.00 | 17754.00 |
| 12 | Male | Graphic Designer | 24 | PhD | 102261.00 | 13491.00 |
| 13 | Female | Driver | 18 | College | 62759.00 | -26011.00 |
| 14 | Female | Financial Analyst | 19 | College | 84007.00 | -4763.00 |
| 15 | Female | Warehouse Associate | 30 | Masters | 86220.00 | -2550.00 |
| 16 | Female | Warehouse Associate | 35 | PhD | 95584.00 | 6814.00 |
| 17 | Female | Marketing Associate | 27 | PhD | 73357.00 | -15413.00 |
| 18 | Female | Financial Analyst | 23 | PhD | 88422.00 | -348.00 |
| 19 | Female | Warehouse Associate | 24 | College | 99545.00 | 10775.00 |
| 20 | Male | Sales Associate | 21 | High School | 90386.00 | 1616.00 |
| 21 | | | | Salary AVERAGE | 88770.00 | |
| 22 | | | | Salary AVERAGEA | 88770.00 | |

## Example use:

Determine if there is a **disparity** between employee annual **base pay** based on **gender**.

**01.** Determine the average of **BasePay** for all employees.

**02.** Subtract the calculated average salary from **BasePay** for each employee.

## Insight

Our dataset reveals that **only female employees** are paid a salary that is **below the average rate**. This is irrespective of their education level, age, or job title.

# The AVERAGE and AVERAGEA functions

01. **Floating-point** numbers will always return a **floating-point** result.

02. **Integer** numbers can produce either a **floating-point** or **integer** number.

03. A **mix** of floating-point and integer data types will result in a **floating-point**.

| | | | | | |
|---|---|---|---|---|---|
| emale | Driver | 18 | College | 62759.00 | -2 |
| 02. | cial Analyst | 19 | College | 84007.00 | - |
| emale | Warehouse Associate | 30 | Masters | 86220.00 | - |
| emale | Warehouse Associate | 35 | PhD | 95584.00 | |
| emale | Marketing Associate | 27 | PhD | 73357.00 | -1 |
| emale | Financial Analyst | 23 | PhD | 88422.00 | |
| emale | Warehouse Associate | 24 | College | 99545.00 | 1 |
| ale | Sales Associate | 21 | High School | 90386.00 | |
| | | | **Salary AVERAGE** | **88770.00** | |
| | | | **Salary AVERAGEA** | **88770.00** | |

| | | | | |
|---|---|---|---|---|
| emale | Driver | 18 | College | 62759.50 |
| 01. | ncial Analyst | 19 | College | 84007.50 |
| emale | Warehouse Associate | 30 | Masters | 86220.50 |
| emale | Warehouse Associate | 35 | PhD | 95584.50 |
| emale | Marketing Associate | 27 | PhD | 73357.50 |
| emale | Financial Analyst | 23 | PhD | 88422.50 |
| emale | Warehouse Associate | 24 | College | 99545.50 |
| lale | Sales Associate | 21 | High School | 90386.50 |
| | | | **Salary AVERAGE** | **88770.50** |
| | | | **Salary AVERAGEA** | **88770.50** |

| | | | | | |
|---|---|---|---|---|---|
| emale | Driver | 18 | College | 62759.00 | - |
| 03. | cial Analyst | 19 | College | 84007.00 | |
| emale | Warehouse Associate | 30 | Masters | 86220.00 | |
| emale | Warehouse Associate | 35 | PhD | 95584.33 | |
| emale | Marketing Associate | 27 | PhD | 73357.40 | - |
| emale | Financial Analyst | 23 | PhD | 88422.50 | |
| emale | Warehouse Associate | 24 | College | 99545.00 | |
| lale | Sales Associate | 21 | High School | 90386.33 | |
| | | | **Salary AVERAGE** | **88770.08** | |
| | | | **Salary AVERAGEA** | **88770.08** | |

# The AVERAGE and AVERAGEA functions

**04.** **Empty** cells will be ignored.

AVERAGE will ignore cells with **text**, including cells with **boolean** values (True and False).

AVERAGEA will translate all **text to zero**, **True** to **1**, and **False** to **0**.

**05.** An **error** will be returned if an argument contains an error.

| | | | | | |
|---|---|---|---|---|---|
| 13 | Female | Driver | 18 | College | 62759.00 |
| 04. | le | Financial Analyst | 19 | College | TRUE |
| 15 | Female | Warehouse Associate | 30 | Masters | 86220.00 |
| 16 | Female | Warehouse Associate | 35 | PhD | |
| 17 | Female | Marketing Associate | 27 | PhD | 73357.00 |
| 18 | Female | Financial Analyst | 23 | PhD | 88422.00 |
| 19 | Female | Warehouse Associate | 24 | College | Analysis |
| 20 | Male | Sales Associate | 21 | High School | 90386.00 |
| 21 | | | | **Salary AVERAGE** | **87968.38** |
| 22 | | | | **Salary AVERAGEA** | **78194.17** |

| | | | | | |
|---|---|---|---|---|---|
| 13 | Female | Driver | 18 | College | 62759.00 |
| 05. | le | Financial Analyst | 19 | College | 84007.00 |
| 15 | Female | Warehouse Associate | 30 | Masters | 86220.00 |
| 16 | Female | Warehouse Associate | 35 | PhD | 95584.00 |
| 17 | Female | Marketing Associate | 27 | PhD | 73357.00 |
| 18 | Female | Financial Analyst | 23 | PhD | #NAME? |
| 19 | Female | Warehouse Associate | 24 | College | 99545.00 |
| 20 | Male | Sales Associate | 21 | High School | 90386.00 |
| 21 | | | | **Salary AVERAGE** | **#NAME?** |
| 22 | | | | **Salary AVERAGEA** | **#NAME?** |

# The CEILING and ROUNDUP functions

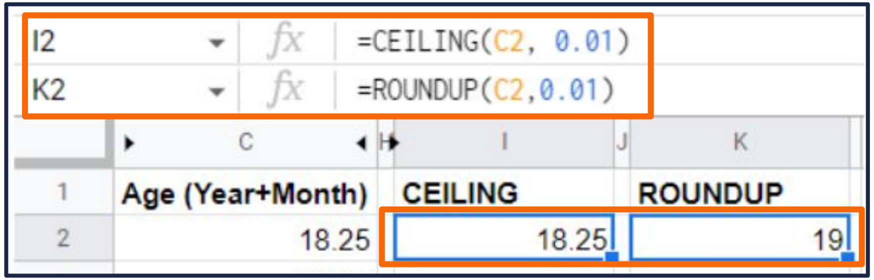
Used to **round** numbers **up**. CEILING rounds up to the nearest **integer** ***multiple** of the specified ****factor** while ROUNDUP rounds up to a certain **number of decimal places**.

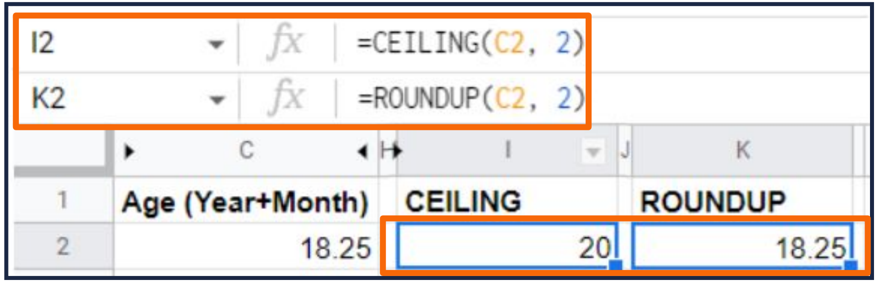
```
=CEILING(value, [factor])
=ROUNDUP(value, [places])
```

- `value` – The value to be rounded off. Can contain the actual value or a cell reference.
- `factor/places` – The precision (number of decimal places) of the result.

* A **multiple** is a **product** of a given number with other numbers. For example, 6 is a multiple of 1 and 6 or 2 and 3.

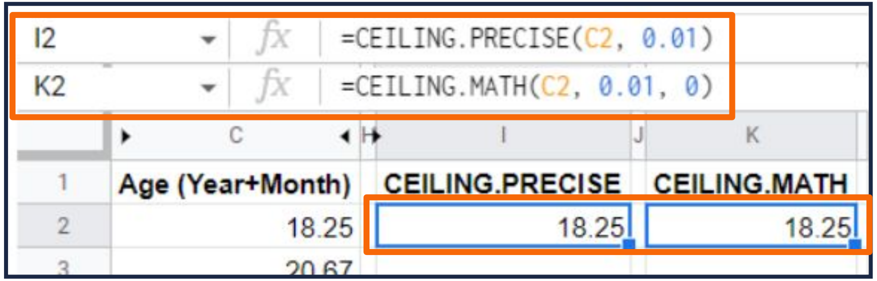
** A **factor divides** a given number **without a remainder**. For example, 1, 2, 3, and 6 are factors of 6.

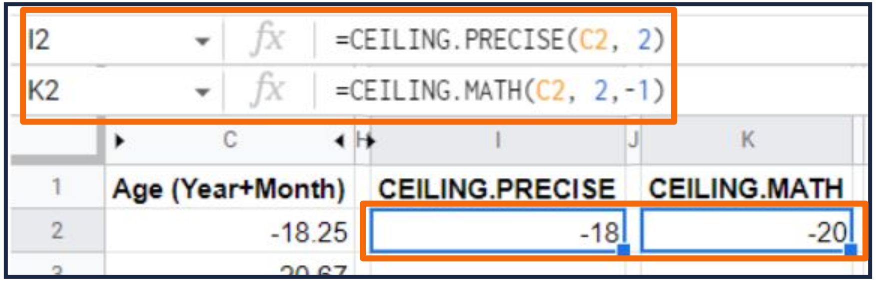


# The alternative CEILING functions

> **CEILING.PRECISE** and **CEILING.MATH** functions round up to the nearest **integer** or **multiple** of the specified factor. CEILING.MATH also specifies if the number is rounded toward or away from zero.




```
=CEILING.PRECISE(value, [factor])
=CEILING.MATH(value,[factor],[mode])
```

- **`value`** – The value to be rounded off. Can contain the actual value or a cell reference.
- **`factor`** – The precision (number of decimal places) of the result.
- **`mode`** – If the value is *negative*, it specifies the rounding direction. Zero or blank will round up towards zero. Otherwise, it's rounded away from zero.

# The CEILING and ROUNDUP functions

## Example use:

**Round up** the employee's **age** to the nearest **year**.

**01.** Round up the value of the first employee.

**02.** Replicate the formula to other employees by dragging down the fill handle.

D2 | fx =CEILING(C2,1)

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Gender** | **JobTitle** | **Age (Year+Month)** | **Age (Years)** | **Education** |
| 2 | Female | Graphic Designer | 18.25 | 19 | College |
| 3 | Male | Software Engineer | 20.67 | | College |
| 4 | Female | Warehouse Associate | 19.00 | | PhD |

D20 | fx =CEILING(C20, 1)

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Gender** | **JobTitle** | **Age (Year+Month)** | **Age (Years)** | **Education** |
| 2 | Female | Graphic Designer | 18.25 | 19 | College |
| 3 | Male | Software Engineer | 20.67 | 21 | College |
| 4 | Female | Warehouse Associate | 19.00 | 19 | PhD |
| 5 | Male | Software Engineer | 19.33 | 20 | Masters |
| 6 | Male | Graphic Designer | 25.50 | 26 | Masters |
| 7 | Female | IT | 20.00 | 20 | PhD |
| 8 | Female | Graphic Designer | 19.92 | 20 | College |
| 9 | Male | Software Engineer | 18.00 | 18 | PhD |
| 10 | Female | Graphic Designer | 32.58 | 33 | High School |
| 11 | Female | Sales Associate | 34.08 | 35 | College |
| 12 | Male | Graphic Designer | 24.00 | 24 | PhD |
| 13 | Female | Driver | 18.33 | 19 | College |

### Insight

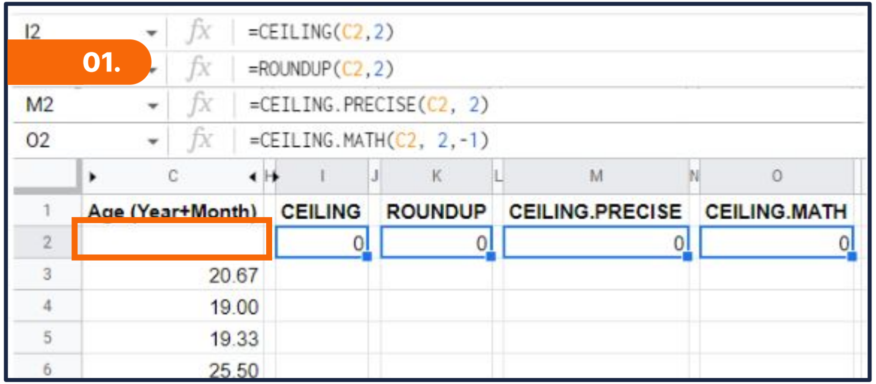
This can be used to **determine** if any employees are expected to **retire within that year**.

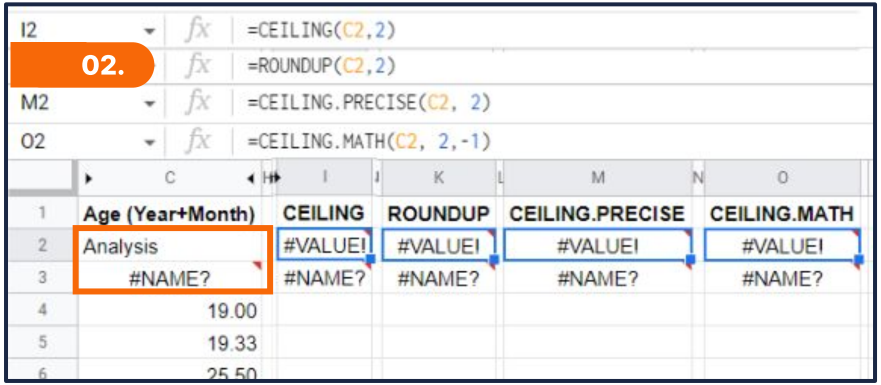
Our **dataset shows** that all employees are between 18-36 years old so **none will be retiring** any time soon.

# The CEILING and ROUNDUP functions

**01.** Will result in zero if the cell is **empty.**

**02.** Will return an **error** if the argument contains text or an error.



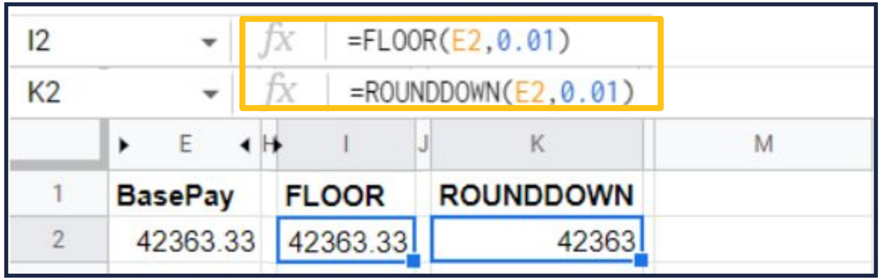



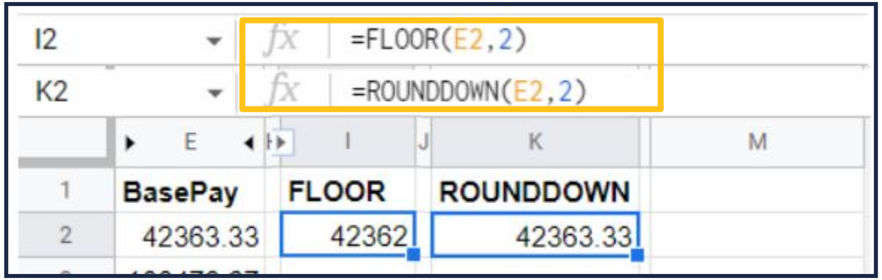
# The FLOOR and ROUNDDOWN functions

FLOOR and ROUNDDOWN are used to **round** numbers **down**. FLOOR rounds a number down to the nearest **multiple** of the specified **factor** while ROUNDDOWN rounds a number to a certain number of **decimal places.**

```
=FLOOR(value, [factor])
=ROUNDDOWN(value,[places])
```

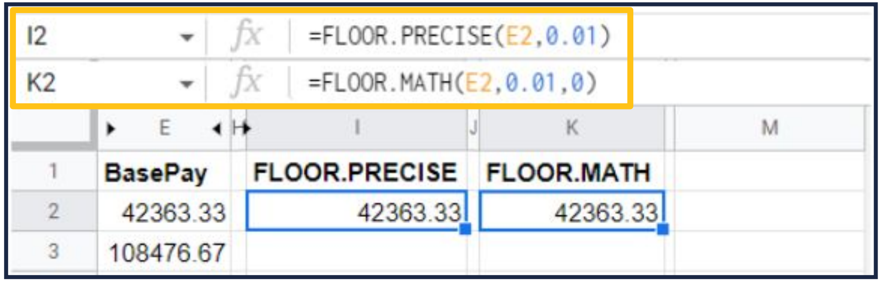
- `value` – The value to round down.
- `factor/places` – The precision (number of decimal places) of the result. This argument is optional and is set to one by default (if not specified).

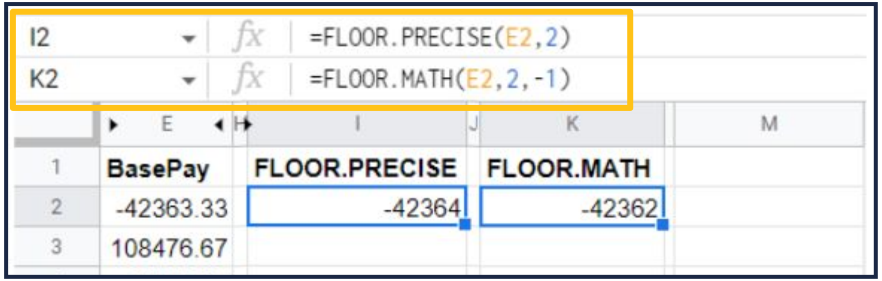


# The alternative FLOOR functions

**FLOOR.PRECISE** and **FLOOR.MATH** round a number down to the nearest **integer** or **multiple** of the specified factor. FLOOR.MATH rounds down toward or away from zero.




```
=FLOOR.PRECISE(value, [factor])
=FLOOR.MATH(value, [factor], [mode])
```

- **`value`** – The value to round down.
- **`factor`** – The precision (number of decimal places) of the result. This argument is optional and is set to one by default (if not specified). It may not be equal to zero or be a cell reference to an empty cell.
- **`mode`** – If value is negative, it specifies the rounding direction. Zero or blank will round away from zero. Otherwise, it's rounded towards zero.

# The FLOOR and ROUNDDOWN functions

## Example use:

**Round down** employee salaries to the **nearest zero cent**.

**01.** Round down the value of the first employee.

**02.** Replicate the formula to other employees by dragging down the fill handle.

`=FLOOR(E2,1)`

| B | C | D | E | F |
|---|---|---|---|---|
| **JobTitle** | **Age (Year+Month)** | **Education** | **BasePay** | **BasePay (Adjusted)** |
| Graphic Designer | 18.25 | College | 42363.33 | 42363.00 |
| Software Engineer | 20.67 | College | 108476.67 | |
| Warehouse Associate | 19.00 | PhD | 90208.25 | |

`=FLOOR(E20,1)`

| B | C | D | E | F |
|---|---|---|---|---|
| **JobTitle** | **Age (Year+Month)** | **Education** | **BasePay** | **BasePay (Adjusted)** |
| Graphic Designer | 18.25 | College | 42363.33 | 42363.00 |
| Software Engineer | 20.67 | College | 108476.67 | 108476.00 |
| Warehouse Associate | 19.00 | PhD | 90208.25 | 90208.00 |
| Software Engineer | 19.33 | Masters | 108080.00 | 108080.00 |
| Graphic Designer | 25.50 | Masters | 99464.80 | 99464.00 |
| IT | 20.00 | PhD | 70890.10 | 70890.00 |
| Graphic Designer | 19.92 | College | 67585.75 | 67585.00 |
| Software Engineer | 18.00 | PhD | 97523.33 | 97523.00 |
| Graphic Designer | 32.58 | High School | 112976.57 | 112976.00 |
| Sales Associate | 34.08 | College | 106524.40 | 106524.00 |
| Graphic Designer | 24.00 | PhD | 102261.01 | 102261.00 |
| Driver | 18.33 | College | 62759.99 | 62759.00 |
| Financial Analyst | 18.25 | College | 84007.80 | 84007.00 |

## Insight

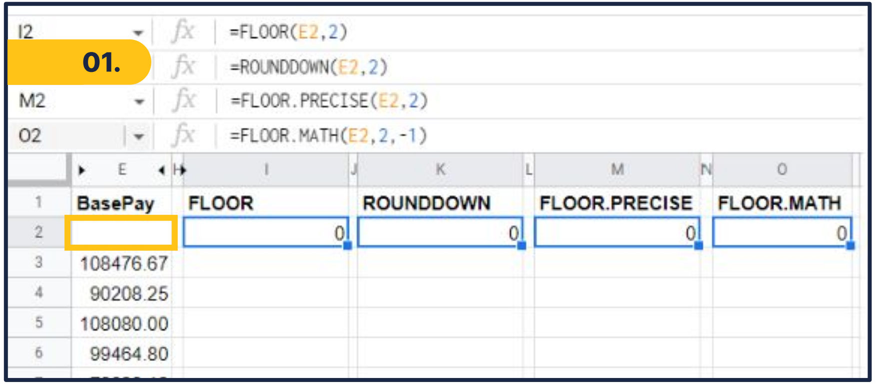
It is a **common** practice to **round down** employees' **after-tax payment** to the nearest cent.

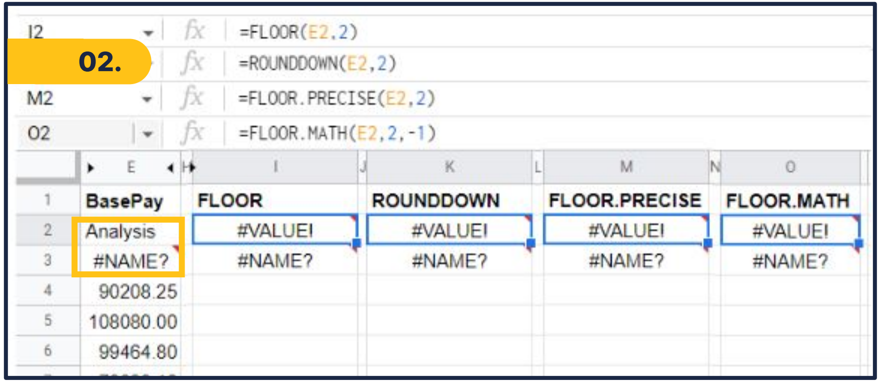
You could use this knowledge to investigate if the **BasePay** was captured **before or after taxation** to hypothesise why it was rounded down.

# The FLOOR and ROUNDDOWN functions

**01.** Will result in zero if the cell is **empty.**

**02.** Will return an **error** if the cell contains text or an error.



An introduction to using data

# The MOD function

> MOD is short for modulo or **modulus**. It divides a number and then gives the **remainder** as an answer.

```
=MOD(dividend, divisor)
```

- **dividend** – The number being divided.
- **divisor** – The number by which you want to divide the dividend argument. The divisor may not be equal to zero or be a cell reference to an empty cell.

For example, **`5 mod 3 = 2`**
**5** is the dividend, **3** is the divisor, and **2** is the modulo.

fx =MOD(E2,100)

| B | C | D | E | F |
|---|---|---|---|---|
| obTitle | Age (Year+Month) | Education | BasePay | BasePay Modulus |
| raphic Designer | 18.25 | College | 42363.33 | 63.33 |
| oftware Engineer | 20.67 | College | 108476.67 | |
| Varehouse Associate | 19.00 | PhD | 90208.25 | |
| oftware Engineer | 19.33 | Masters | 108080.00 | |
| raphic Designer | 25.50 | Masters | 99464.80 | |
| T | 20.00 | PhD | 70890.10 | |
| raphic Designer | 19.92 | College | 67585.75 | |
| oftware Engineer | 18.00 | PhD | 97523.33 | |
| raphic Designer | 32.58 | High School | 112976.57 | |
| ales Associate | 34.08 | College | 106524.40 | |
| raphic Designer | 24.00 | PhD | 102261.01 | |
| river | 18.33 | College | 62759.99 | |
| inancial Analyst | 18.25 | College | 84007.80 | |
| Varehouse Associate | 29.67 | Masters | 86220.33 | |

# The MOD function

fx =MOD(E2,100)

| | B | C | D | E | F |
|---|---|---|---|---|---|
| | obTitle | Age (Year+Month) | Education | BasePay | BasePay Modulus |
| | Graphic Designer | 18.25 | College | 42363.33 | 63.33 |
| | oftware Engineer | 20.67 | College | 108476.67 | |
| | Varehouse Associate | 19.00 | PhD | 90208.25 | |

fx =MOD(E20,100)

| B | C | D | E | F |
|---|---|---|---|---|
| JobTitle | Age (Year+Month) | Education | BasePay | BasePay Modulus |
| Graphic Designer | 18.25 | College | 42363.33 | 63.33 |
| Software Engineer | 20.67 | College | 108476.67 | 76.67 |
| Warehouse Associate | 19.00 | PhD | 90208.25 | 8.25 |
| Software Engineer | 19.33 | Masters | 108080.00 | 80.00 |
| Graphic Designer | 25.50 | Masters | 99464.80 | 64.80 |
| IT | 20.00 | PhD | 70890.10 | 90.10 |
| Graphic Designer | 19.92 | College | 67585.75 | 85.75 |
| Software Engineer | 18.00 | PhD | 97523.33 | 23.33 |
| Graphic Designer | 32.58 | High School | 112976.57 | 76.57 |
| Sales Associate | 34.08 | College | 106524.40 | 24.40 |
| Graphic Designer | 24.00 | PhD | 102261.01 | 61.01 |
| Driver | 18.33 | College | 62759.99 | 59.99 |
| Financial Analyst | 18.25 | College | 84007.80 | 7.80 |

## Example use:

Identify **salaries** that are **not multiples of 100.**

**01.** Calculate the remainder of the first employee's salary when divided by 100.

**02.** Replicate the formula to other employees by dragging down the fill handle.

## Insight

Our dataset fully consists of data that are **not multiples of 100**.

This could suggest that employees are **paid on an hourly basis** – a hypothesis worth checking out!

# The MOD function

**01.** Will result in zero if the cell is **empty.**

**02.** Will return an **error** if the argument contains text or an error.

**01.**

=MOD(E2,100)

| JobTitle | Age (Year+Month) | Education | BasePay | BasePay Modulus |
|---|---|---|---|---|
| Graphic Designer | 18.25 | College | | 0.00 |
| Software Engineer | 20.67 | College | 108476.67 | 76.67 |
| Warehouse Associate | 19.00 | PhD | 90208.25 | 8.25 |
| Software Engineer | 19.33 | Masters | 108080.00 | 80.00 |
| Graphic Designer | 25.50 | Masters | 99464.80 | 64.80 |
| IT | 20.00 | PhD | 70890.10 | 90.10 |
| Graphic Designer | 19.92 | College | 67585.75 | 85.75 |
| Software Engineer | 18.00 | PhD | 97523.33 | 23.33 |

**02.**

=MOD(E3,100)

| JobTitle | Age (Year+Month) | Education | BasePay | BasePay Modulus |
|---|---|---|---|---|
| Graphic Designer | 18.25 | College | Analysis | #VALUE! |
| Software Engineer | 20.67 | College | #NAME? | #NAME? |
| Warehouse Associate | 19.00 | PhD | 90208.25 | 8.25 |
| Software Engineer | 19.33 | Masters | 108080.00 | 80.00 |
| Graphic Designer | 25.50 | Masters | 99464.80 | 64.80 |
| IT | 20.00 | PhD | 70890.10 | 90.10 |
| Graphic Designer | 19.92 | College | 67585.75 | 85.75 |
| Software Engineer | 18.00 | PhD | 97523.33 | 23.33 |

# The SQRT and POWER/POW functions

> SQRT is used to calculate the square root of a given number, while POW, which is equivalent to POWER, is used to calculate the value of a number (base) raised to a certain exponent.

```
=SQRT(value)
```

- **`value`** – The number whose square root we want to calculate.

For example, $\sqrt{4}$ will be entered as **`=SQRT(4)`**

```
=POW(base, exponent)
=POWER(base, exponent)
```

- **`base`** – Any real number.
- **`exponent`** – The number of times the base will be multiplied by itself.

For example, $2^3$ will be entered as **`=POW/POWER(2,3)`**

# The SQRT and POWER/POW functions

C2 =POW(A2,3)
E2 =SQRT(A2)

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Number | | Power | | Square Root | |
| 2 | 2 | | 8 | | 1 | |
| 3 | 4 | | | | | |
| 4 | 9 | | | | | |

I22

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Number | | Power | | Square Root | |
| 2 | 2 | | 8 | | 1 | |
| 3 | 4 | | 64 | | 2 | |
| 4 | 9 | | 729 | | 3 | |
| 5 | 16 | | 4096 | | 4 | |
| 6 | 25 | | 15625 | | 5 | |
| 7 | 36 | | 46656 | | 6 | |
| 8 | 49 | | 117649 | | 7 | |
| 9 | 64 | | 262144 | | 8 | |
| 10 | 81 | | 531441 | | 9 | |
| 11 | 100 | | 1000000 | | 10 | |

## Example use:

When working with **mathematical equations**.

**01.** Calculate the square root and power of the first number.

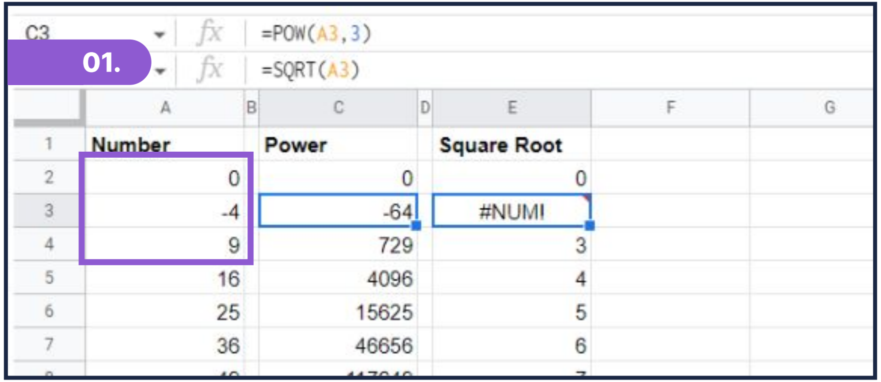
**02.** Replicate the formula to other rows by dragging down the fill handle.

## An introduction to using data

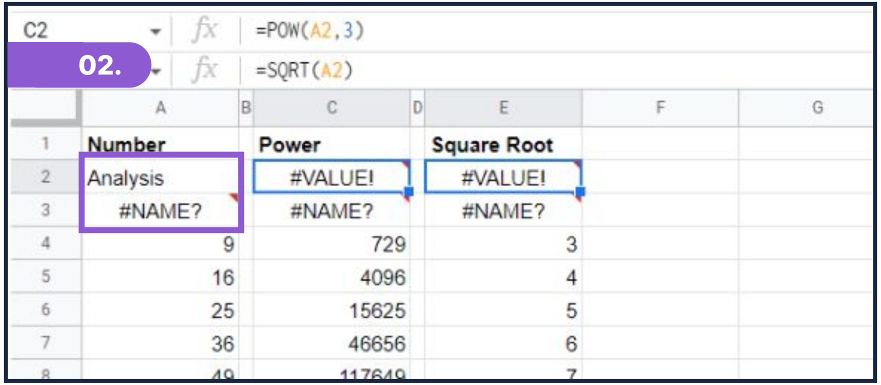
# The SQRT and POWER/POW functions

**01.** SQRT only works with **positive numbers** or **zero, while** POW (or POWER) works with **positive** and **negative** numbers.

**02.** SQRT and POW (or POWER) will result in an **error** if the argument is **text** or an **error**.




| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Number | | Power | | Square Root | | |
| 2 | 0 | | 0 | | 0 | | |
| 3 | -4 | | -64 | | #NUM! | | |
| 4 | 9 | | 729 | | 3 | | |
| 5 | 16 | | 4096 | | 4 | | |
| 6 | 25 | | 15625 | | 5 | | |
| 7 | 36 | | 46656 | | 6 | | |




| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Number | | Power | | Square Root | | |
| 2 | Analysis | | #VALUE! | | #VALUE! | | |
| 3 | #NAME? | | #NAME? | | #NAME? | | |
| 4 | 9 | | 729 | | 3 | | |
| 5 | 16 | | 4096 | | 4 | | |
| 6 | 25 | | 15625 | | 5 | | |
| 7 | 36 | | 46656 | | 6 | | |