Here’s the result of both expressions:

```python
import math print(math.trunc(2.8)) # Output: 2 print(math.trunc(-2.8)) #
Output: -2
```

Explanation:

- `math.trunc(x)` removes the decimal part of the number and **truncates toward zero**.
- So:
  - `math.trunc(2.8) → 2`
  - `math.trunc(-2.8) → -2`

This is different from `math.floor()` and `math.ceil()`:

- `math.floor(-2.8) → -3` (rounds down)
- `math.ceil(-2.8) → -2` (rounds up toward zero)

towarf zero means

"Toward zero" means the value is **truncated in the direction of zero**, regardless of whether it's positive or negative.

In simpler terms:

- For **positive numbers**, it removes the decimal and keeps the **lower integer**.
  - Example: `2.8 → 2`
- For **negative numbers**, it removes the decimal and keeps the **higher integer** (closer to zero).

- Example: `-2.8` → `-2`

Contrast with Other Behaviors:

Function	`2.8`	`-2.8`
`math.trunc()`	2	-2
`math.floor()`	2	-3
`math.ceil()`	3	-2

So, **"truncating toward zero"** just means dropping the decimal part and keeping the integer part that is closest to zero.

closest to zero

Yes, exactly — “**closest to zero**” means the integer part of a number that has the **smallest absolute value** but still doesn't exceed the original number in magnitude.

Here’s what that looks like:

Number	Closest Integer Toward Zero	Why
2.8	2	2 is closer to 0 than 3
-2.8	-2	-2 is closer to 0 than -3
0.9	0	0 is closer to 0 than 1
-0.9	0	0 is closer to 0 than -1

So when we say `math.trunc()` **truncates toward zero**, it means:

- It **removes the decimal** part of the number.
- The result is the **integer that is closest to zero**, without going past it.