6. Integer

absolute

`to_user = (abs(user_floor - elevator_floor))`

random

first we need to import the library:

`import random`

`bingo = random.randint(1,100)`

7. String

String Operations

For some of these, we need to import the library:

`import string`

Search

- **var.find() / var.rfind():** Searches the string for a specified value and returns the position of where it was found
- **var.index():** Searches the string for a specified value and returns the position of where it was found
- **var.count():** Returns the number of times a specified value occurs in a string

Format / Split / Replace

- **var.rjust():** `syntax | num = num.rjust(width, 'fillchar')` `example | num = num.rjust(2, '0')`
- **var.join():** `var += ''.join('Enter text here')`
- **var.partition(): / var.rpartition()** Returns a tuple where the string is parted into three parts
- **var.split() / var.rsplit():** Splits the string at the specified separator, and returns a list
- **var.splitlines():** Splits the string at line breaks and returns a list
- **var.rstrip() / var.lstrip():** Returns a right/left trim version of the string
- **var.replace:** Returns a string where a specified value is replaced with a specified value

Lowercase / Uppercase Conversion

- **var.upper():** Convert a string to uppercase
- **var.lower():** Convert a string to lowercase
- **var.capitalize():** Capitalizes the string. First letter is CAPITAL, rest are small letters
- **var.swapcase():** Swaps cases, lower case becomes upper case and vice versa
- **var.title():** Converts the first character of each word to upper case
- **var.casefold():** Converts string into lower case

Boolean Checks

- **var.startswith():** Returns true if the string starts with the specified value
- **var.endswith():** Returns true if the string ends with the specified value
- **var.istitle():** Returns True if the string follows the rules of a title
- **var.isalnum():** Returns True if all characters in the string are alphanumeric
- **var.isalpha():** Returns True if all characters in the string are in the alphabet
- **var.isascii():** Returns True if all characters in the string are ascii characters
- **var.isdigit():** Returns True if all characters in the string are digits
- **var.isnumeric():** Returns True if all characters in the string are numeric
- **var.isspace():** Returns True if all characters in the string are whitespaces
- **'text' in var:** Check if a letter/symbol exists in a string. returns True/False `check_sym = '@' in address`

Slicing

Slicing

```python
b = "Hello, World!"

# Get the characters from position 2 to position 5 (not included):
print(b[2:5]) # Output: "ell"

# Get the characters from position -5 to position -2 (not included):
print(b[-5:-2]) # Output: "orl"
```