+ - / * // **

SIMPLE PYTHON CALCULATOR

Do some math with two inputs you entered

Structure, how I made folders and files

I wanted to make it a simple python package

Pyproject.toml – app specifications

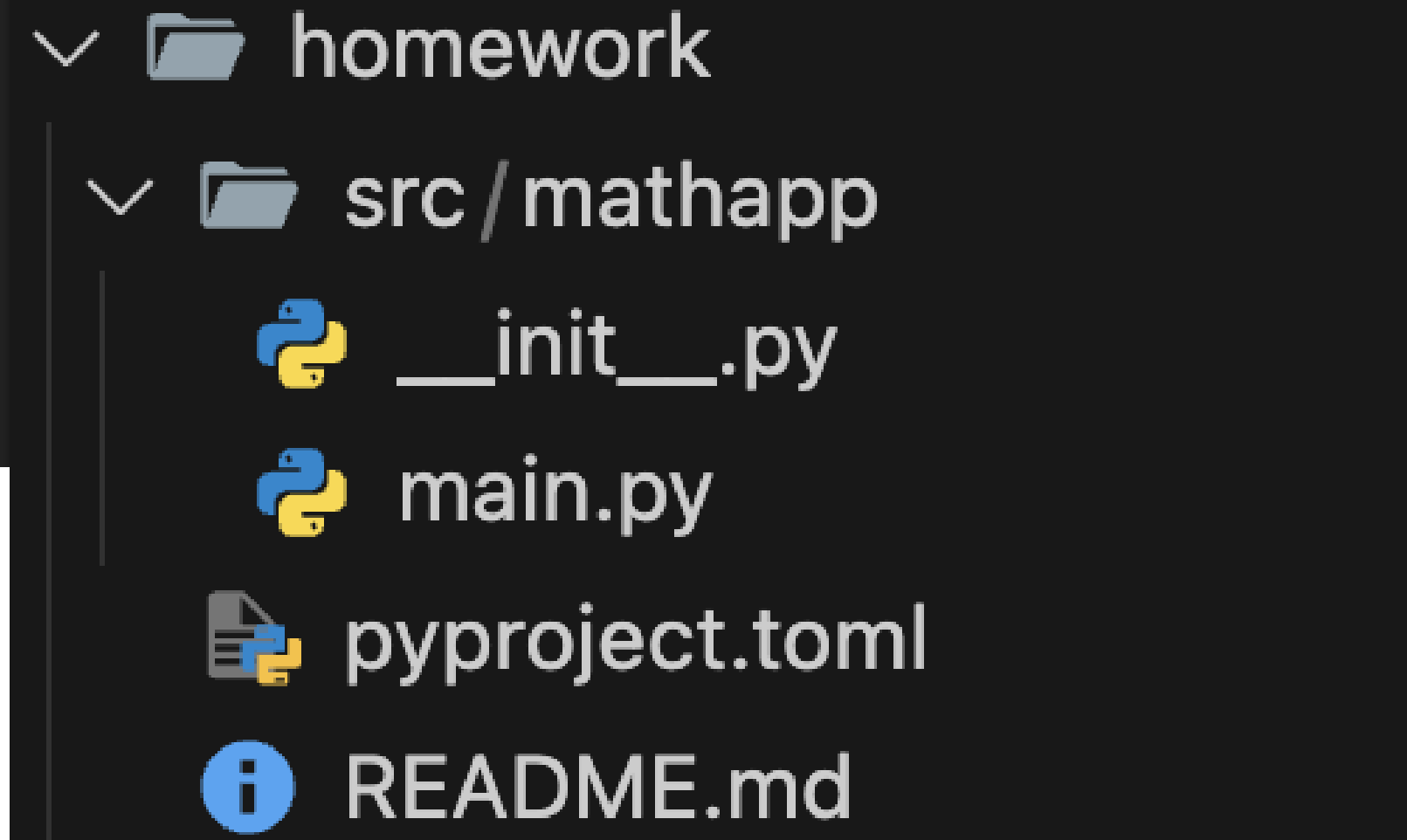
README.md – docs & manuals

Setup the application

```
pip install -e .
```

Run the application and see the result

```
run
```



Constants

```
# Defining the math operators we can use
OPERATORS = ['+', '-', '/', '*', '//', '**']
```

- Clean and understable code
- Prevent from accidental changes leading bugs
- Change at once
- …

```
# A simple calculation program

# Defining the math operators we can use
OPERATORS = ['+', '-', '/', '*', '//', '**']

# Checking if the input is number
def validate_numeric(input):
    if not input.isnumeric():
        print("The input is not a number!")
        do_math()

def do_math():
    print("-------------------------")
    # Defining first input variable
    a = input('Enter the first number: \n')
    validate_numeric(a)

    # define second input variable
    b = input("Enter the second number: \n")
    validate_numeric(b)

    # ask for math operator
    o = input("Enter one of following math operators (+, -, /, *, //, **): \n")
    if o not in OPERATORS:
        print("The operator is not valid!")
        do_math()

    # check if division by zero
    if o == '/' and b == '0':
        print("Division by zero!")
        do_math()

    result = eval(f"{a} {o} {b}")

    # calculate the result
    print(f"The result is, \n {result}")
```

Input Validation

- Check first and second input is number or not
- Use built-in String.isnumeric() method

```
# Checking if the input is number
def validate_numeric(input):
    if not input.isnumeric():
        print("The input is not a number!")
        do_math()

def do_math():
    print("-------------------------")
    # Defining first input variable
    a = input('Enter the first number: \n')
    validate_numeric(a)

    # define second input variable
    b = input("Enter the second number: \n")
    validate_numeric(b)
```

Logic ☺

- built-in eval() function in python
- print suitable message to user
- restart app if validation fails
- result at the end

```
# ask for math operator
o = input("Enter one of following math operators (+, -, /, *, //, **): \n")
if o not in OPERATORS:
    print("The operator is not valid!")
    do_math()

# check if division by zero
if o == '/' and b == '0':
    print("Division by zero!")
    do_math()

result = eval(f"{a} {o} {b}")

# calculate the result
print(f"The result is, \n {result}")
```