Nama = Helmy Luqmanulhakim

NIM = 22051204014

Kelas = Sains Komputasi 2022A
****

Task [1] : Penjumlahan Matriks A dan B

In [ ]:
```
import numpy as np

matrix_A : np.ndarray = np.array([
    [8., 5., 6.],
    [4., 5., 2.],
    [9., 7, 10.]
])

print(f"Data type member of matrix A : {matrix_A[0, 0].dtype}", end='\n\n')
print(f"Matrix A : \n{matrix_A}")
```

```
Data type member of matrix A : float64

Matrix A :
[[ 8.  5.  6.]
 [ 4.  5.  2.]
 [ 9.  7. 10.]]
```

In [ ]:
```
matrix_B : np.ndarray = np.array([
    [5., 5., 6.],
    [4., 5., 2.],
    [9., 7, 6.]
])

print(f"Data type member of matrix B : {matrix_B[0, 0].dtype}", end='\n\n')
print(f"Matrix B : \n{matrix_B}")
```

```
Data type member of matrix B : float64

Matrix B :
[[5. 5. 6.]
 [4. 5. 2.]
 [9. 7. 6.]]
```

In [ ]:
```
print(f"Matrix A + Matrix B : \n{matrix_A + matrix_B}")
```

```
Matrix A + Matrix B :
[[13. 10. 12.]
 [ 8. 10.  4.]
 [18. 14. 16.]]
```

Task [2] : Pengurangan Matriks A dan B

In [ ]:
```
print(f"Matrix A - Matrix B : \n{matrix_A - matrix_B}")
```

```
Matrix A - Matrix B :
[[3. 0. 0.]
 [0. 0. 0.]
 [0. 0. 4.]]
```

Task [3] : Zero Matrix

In [ ]:
```python
import numpy as np

matrix_A : np.ndarray = np.array([
    [3., 8., 5.],
    [6., 4., 7.]
])

matrix_C : np.ndarray = np.array([
    [9., 5., 3.],
    [7., 2., 1.]
])

dim_n : int = 2
dim_m : int = 3
matrix_D : np.ndarray = np.zeros((dim_n, dim_m))

for i in range(dim_n):
    for j in range(dim_m):
        matrix_D[i][j] = matrix_A[i][j] + matrix_C[i][j]

print(f"matrix_D : \n{matrix_D}")
```

```
matrix_D :
[[12. 13.  8.]
 [13.  6.  8.]]
```

Task [4] : Analisis Operasi Matriks

In [ ]:
```python
import numpy as np

matrix_A : np.ndarray = np.array([
    [3., 8., 5.],
    [6., 4., 7.]
])
matrix_B : np.ndarray = np.array([
    [1., 3.],
    [5., 9.],
    [2., 4.]
])

dim_n : int = 2
dim_m : int = 3
dim_p : int = 2

matrix_E : np.ndarray = np.zeros((dim_n, dim_p))

for i in range(dim_n):
    for j in range(dim_p):
        for k in range(dim_m):
```

```
            matrix_E[i][j] += matrix_A[i][k] * matrix_B[k][j]

print(f"matrix_E : \n{matrix_E}")
```

```
matrix_E :
[[ 53. 101.]
 [ 40.  82.]]
```

Pada operasi di atas dilakukan proses inisialisasi matriks A berordo 2x3 dan matriks B berordo 3x2. Kemudian dilakukan inisialisasi matrix nol sebagai matriks yang akan menampung hasil operasi.

Operasi di atas merupakan operasi `Dot Product` untuk Matriks A dan B.

Task [5] : Analisis Operasi Matriks

In [ ]:
```
import numpy as np

matrix_A : np.ndarray = np.array([
    [3., 8., 5.],
    [6., 4., 7.]
])

matrix_X : np.ndarray = np.array([
    [2.],
    [3.],
    [4.]
])

dim_n : int = 2
dim_m : int = 3

matrix_E : np.ndarray = np.zeros((dim_n, 1))

for i in range(0, dim_n):
    for j in range(0, dim_m):
        matrix_E[i][0] += matrix_A[i][j] * matrix_X[j][0]

print(f"matrix_E : \n{matrix_E}")
```

```
matrix_E :
[[50.]
 [52.]]
```

Pada operasi di atas dilakukan proses inisialisasi matriks A berordo 3x2 dan matriks X berordo 1x3. Kemudian dilakukan inisialisasi matrix nol sebagai matriks yang akan menampung hasil operasi.

Operasi di atas merupakan operasi `Dot Product` untuk Matriks A dan B.

Task [6] : Analisis Operasi List

In [ ]:
```
list_X : list = [
    [20, 9],
    [8, 5],
    [9, 8]
]

list_result : list = [
    [0, 0, 0],
    [0, 0, 0]
]

for i in range(len(list_X)):
    for j in range(len(list_X[0])):
        list_result[j][i] = list_X[i][j]

print("Transpose of matrix X : ")
for r in list_result:
    print(r)
```

```
Transpose of matrix X :
[20, 8, 9]
[9, 5, 8]
```

Pada kode di atas, dilakukan proses transponse matrix untuk List X.

Task [7] : Analisis Operasi

In [ ]:
```
a: int = 5
b: int = 6
c: int = 7

y: int = a*2 + b*3 + c*4

print(f"y = {y}")
```

```
y = 56
```

Pada operasi di atas, melakukan proses perkalian dan pertambahan variabel a dan b.

Task [8] : Analisis Operasi Fungsi

In [ ]:
```
def equation(a: int, b: int) -> int:
    return 25*a + 60*b

print(f"f (2,3) = {equation(2, 3)}")
```

```
f (2,3) = 230
```

Pada fungsi di atas sama seperti operasi pada task 7, hanya saja dilakukan proses dengan menggunakan fungsi.

Task [9] : Analisis Operasi dalam Class

In [ ]:
```python
class MathFunc():
    def eq1(a: int, b: int, c: int) -> int:
        return 2*a + 3*b + 4*c

print(f"f (5,6,7) = {MathFunc.eq1(5, 6, 7)}")
```

```
f (5,6,7) = 56
```

Pada class di atas sama seperti operasi pada task 7 dan 8, hanya saja dilakukan proses dengan menggunakan class.

Task [10] : Analisis Operasi dalam Class

In [ ]:
```python
class MathFunc():
    def eq1(a: int, b: int, c: int) -> int:
        return 2*a + 3*b + 4*c

    def eq2(a: int, b: int, c: int) -> tuple:
        x = MathFunc.eq1(a, b, c)
        y = a * b * c
        return x, y

print(f"f (5,6,7) = {MathFunc.eq2(5, 6, 7)}")
```

```
f (5,6,7) = (56, 210)
```

Dilakukan penambahan fungsi pada class di atas.

Task [11] : Analisis Operasi dalam Class

In [ ]:
```python
class MathFunc():
    def eq1(a: int, b: int, c: int) -> int:
        return 2*a + 3*b + 4*c

    def eq2(a: int, b: int, c: int) -> tuple:
        X = MathFunc.eq1(a, b, c)
        y = a * b * c
        return X, y

    def eq3(a: int, b: int, c: int) -> int:
        X = a*6 + b*8 + c*8
        print(f"X = {X}")
        return X

    def eq4(a: int, b: int, c: int) -> tuple:
        X, y = MathFunc.eq2(a, b, c)
        print(f"X = {X}, y = {y}")
        return X, y

print(f"f (1,1,1) = {MathFunc.eq3(1, 1, 1)}", end='\n\n')
print(f"f (1,1,1) = {MathFunc.eq4(2, 2, 2)}")
```

```
X = 22
f (1,1,1) = 22

X = 18, y = 8
f (1,1,1) = (18, 8)
```

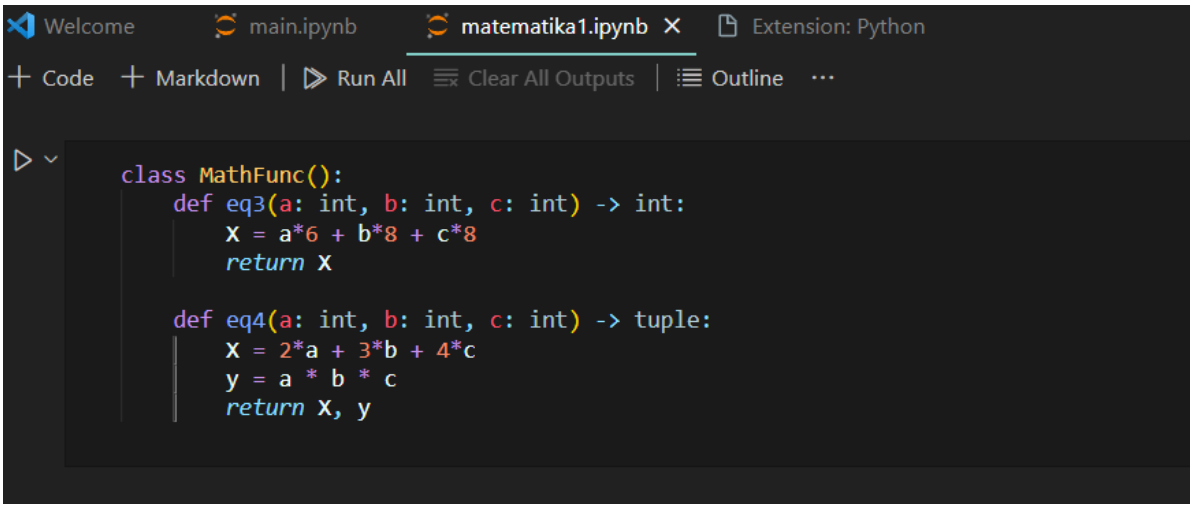
Dilakukan penambahan fungsi pada class di atas.

Task [12]

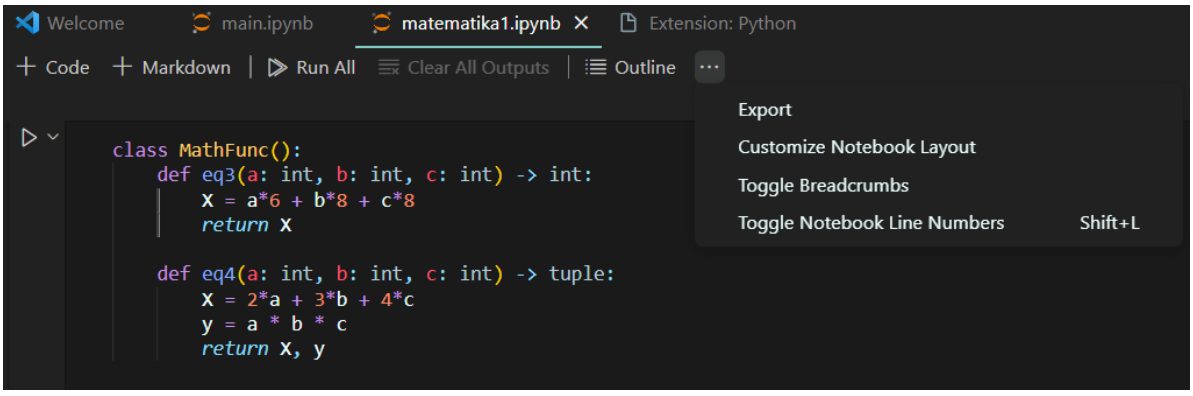
In [ ]:
```python
def eq3(a: int, b: int, c: int) -> int:
    X = a*6 + b*8 + c*8
    return X

print(f"f (1,1,1) = {eq3(1, 1, 1)}")
```

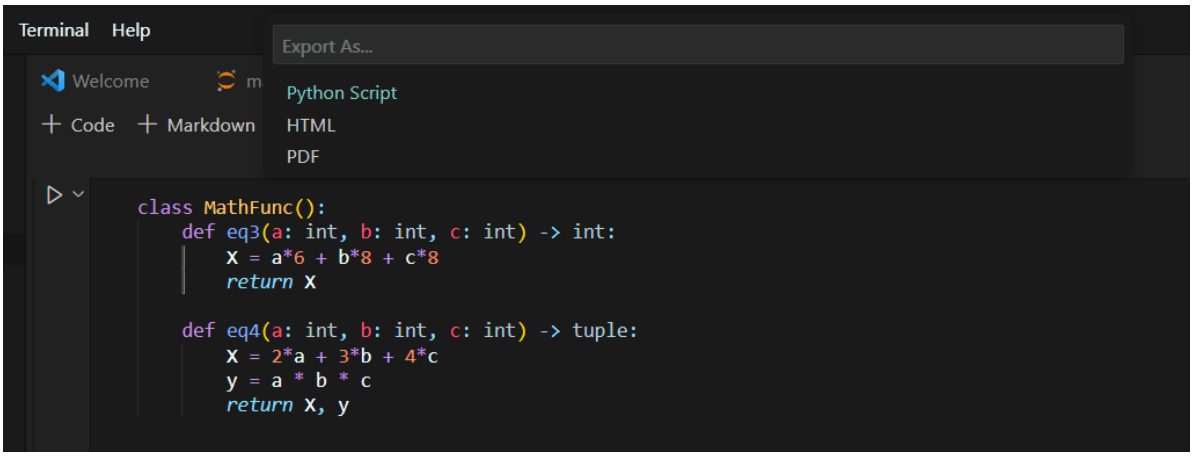
```
f (1,1,1) = 22
```

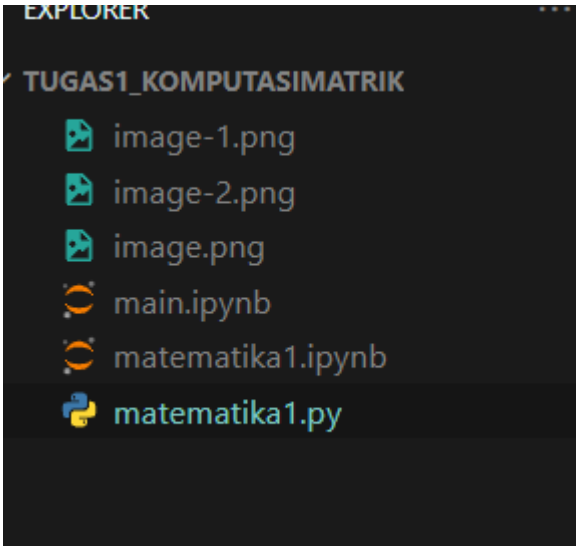
Task [13] : Simpan file ipynb sebagai matematika1.ipynb



Task [14] : Konversi ipynb menjadi pyscript







Task [15] : Konversi ipynb menjadi pyscript

In [ ]:
```
from matematika1 import MathFunc

print(f"f (1,1,1) = {MathFunc.eq4(1, 1, 1)}")
print(f"f (3,3,3) = {MathFunc.eq3(3, 3, 3)}")
```

```
f (1,1,1) = (9, 1)
f (3,3,3) = 66
```

Task [15] : Konversi ipynb menjadi pyscript

```
from matematika1 import MathFunc

print(f"f (1,1,1) = {MathFunc.eq4(1, 1, 1)}")
print(f"f (3,3,3) = {MathFunc.eq3(3, 3, 3)}")
```

[62] ✓ 0.0s

```
f (1,1,1) = (9, 1)
f (3,3,3) = 66
```

PROBLEMS OUTPUT DEBUG CONSOLE PORTS JUPYTER TERMINAL

```
Dev\SainsKomputasi\Tugas1_KomputasiMatrik via v3.11.3
❯ python
Python 3.11.3 | packaged by Anaconda, Inc. | (main, Apr 19 2023, 23:46:34) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from matematika1 import MathFunc
>>> MathFunc.eq4(1, 1, 1)}
  File "<stdin>", line 1
    MathFunc.eq4(1, 1, 1)}
                         ^
SyntaxError: unmatched '}'
>>> MathFunc.eq4(1, 1, 1)
(9, 1)
>>> MathFunc.eq3(3, 3, 3)
66
>>> _
```