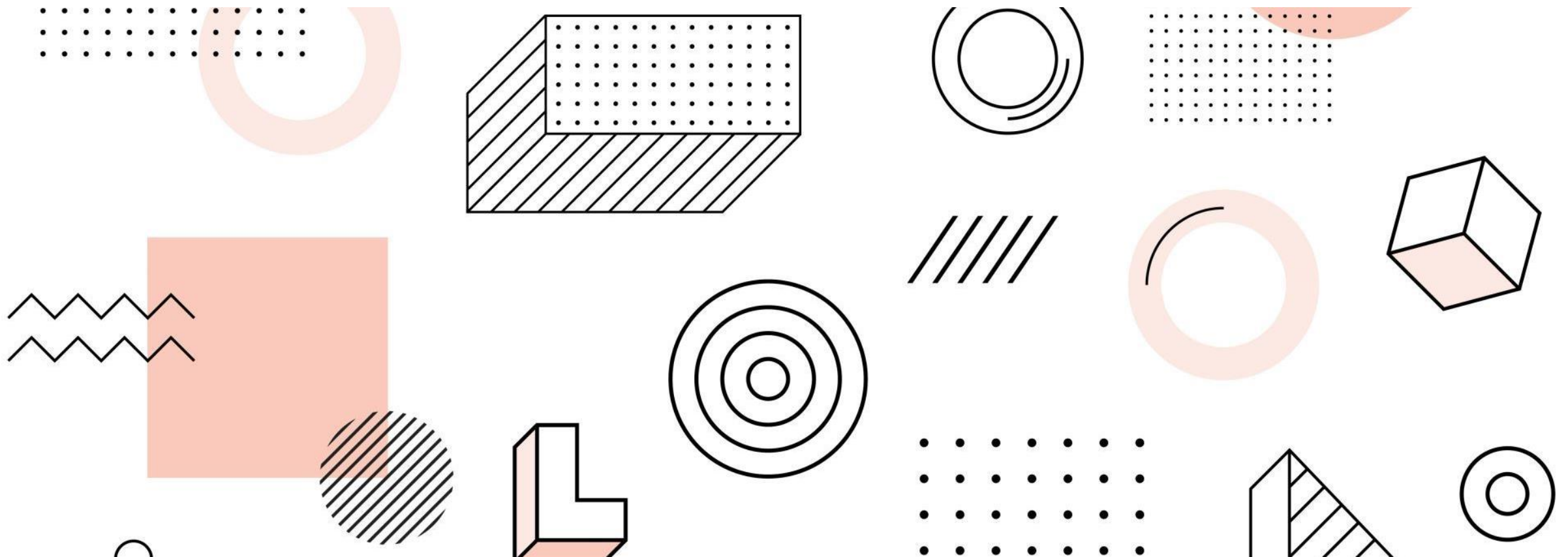


\_\_\_\_\_

## Týden 11. Matematické výrazy v balíčku Numpy



# Tak začneme!

Dneska se seznámíme s velice populárním balíčkem Numpy:

- **Numpy** je základní balíček pro numerické výpočty v jazyce Python. Poskytuje výkonné datové struktury, jako jsou pole a matice, spolu s kolekcí matematických funkcí, které umožňují efektivně provádět numerické operace.

# VÍCEROZMĚRNÁ POLE

Hlavní datovou strukturou v numpy je **ndarray** (n-rozměrné pole), které umožňuje ukládat a manipulovat s homogenními daty pevné velikosti. Umožňuje efektivní a vektorové výpočty na velkých souborech dat, takže je vhodná pro úlohy, jako je analýza dat, vědecké výpočty a strojové učení.

```
import numpy as np
a = np.array([1, 2, 3])
b = np.array([(1.5, 2, 3), (4, 5, 6)])
```

## VYTVÁŘENÍ POLÍ

Numpy nabízí funkce pro snadné vytváření polí, včetně generování posloupností, přetváření a skládání polí. S poli lze provádět různé operace, například indexování a slicing, aritmetické operace s prvky a agregace (např. součet, průměr, max).

```
c = np.zeros((3, 4))
d = np.ones((2, 3, 4))
np.empty((2, 3))
np.arange(10, 30, 5)
np.linspace(0, 2 * np.pi, 6)
```

# INDEXACE

Jednorozměrná pole lze indexovat, slicovat a iterovat nad nimi podobně jako nad seznamy a jinými Python posloupnostmi.

```
b = np.array([[ 0,  1,  2,  3],
               [10, 11, 12, 13],
               [20, 21, 22, 23],
               [30, 31, 32, 33]])

b[2, 3]
b[0:5, 2:4]
```

# VYSÍLÁNÍ

Numpy podporuje vysílání (broadcasting), což je výkonný mechanismus pro provádění operací s prvky na polích různých tvarů. Automaticky zarovnává a rozšiřuje pole tak, aby odpovídala kompatibilním tvarům, čímž eliminuje potřebu explicitních smyček.

```
a = np.array([[ 0,  0,  0],  
              [10, 10, 10],  
              [20, 20, 20],  
              [30, 30, 30]])
```

```
b = np.array([1, 2, 3])  
a + b
```

## MATEMATICKÉ FUNKCE

Numpy poskytuje rozsáhlou sadu matematických funkcí pro úlohy jako trigonometrie, logaritmy, exponenciály, statistické výpočty a lineární algebra. Tyto funkce jsou optimalizovány pro výkon a dokáží efektivně pracovat s velkými poli.

```
A = np.arange(3)
np.exp(A)
np.sqrt(A)
np.sin(A)
np.cos(A)
```