

NumPy Kutubxonasi To'liq Imkoniyatlari (Kommentariyalar bilan)

```
import numpy as np

# =====
# 1. Asosiy massiv (array) operatsiyalari
# =====

# Oddiy massiv yaratish
a = np.array([1, 2, 3])

# 2x3 o'lchamda nol to'ldirilgan massiv
b = np.zeros((2, 3))

# 3x3 o'lchamda 1-lardan iborat massiv
c = np.ones((3, 3))

# 2x2 shaklda 7 bilan to'ldirilgan massiv
d = np.full((2, 2), 7)

# 0 dan 10 gacha, 2 qadam bilan massiv
e = np.arange(0, 10, 2)

# 0 dan 1 gacha 5 ta teng oraliqli qiymatlar
f = np.linspace(0, 1, 5)

# 3x3 birlik matritsa
g = np.eye(3)

print(a, "\n", b, "\n", c, "\n", d, "\n", e, "\n", f, "\n", g)

# =====
# 2. Arifmetik va matematik amallar
# =====
x = np.array([1, 2, 3])
y = np.array([10, 20, 30])

# Elementma-element qo'shish
print(x + y)

# Elementma-element ko'paytirish
print(x * y)

# Har bir elementni kvadratga ko'tarish
print(x ** 2)

# Kvadrat ildiz
print(np.sqrt(y))

# Eksponenta  $e^x$ 
print(np.exp(x))

# Logarifm
print(np.log(x))

# Sinus
print(np.sin(x))

# =====
# 3. Shakl bilan ishlash
# =====
arr = np.arange(6)

# 2x3 shaklga o'zgartirish
print(arr.reshape(2, 3))

# 1D shaklga tekislash
print(arr.flatten())

# Transpozitsiya (qator va ustunni almashtirish)
```

```

print(arr.T)

# =====
# 4. Indeks slash va slicing
# =====
m = np.array([[1, 2, 3],
              [4, 5, 6],
              [7, 8, 9]])

# Bitta element
print(m[0, 2])

# Butun ustun
print(m[:, 1])

# Butun qator
print(m[1, :])

# Kichik qism (sub-massiv)
print(m[0:2, 0:2])

# Shart bo'yicha filtr
print(m[m > 5])

# Shart asosida almashtirish
print(np.where(m > 5, 1, 0))

# =====
# 5. Statistik funksiyalar
# =====
data = np.array([1, 2, 3, 4, 5])

print(np.sum(data))      # Yig'indi
print(np.mean(data))     # O'rtacha qiymat
print(np.std(data))      # Standart og'ish
print(np.min(data))      # Eng kichik qiymat
print(np.max(data))      # Eng katta qiymat
print(np.percentile(data, 50)) # Median (50%)

# =====
# 6. Chiziqli algebra (numpy.linalg)
# =====
A = np.array([[1, 2],
              [3, 4]])
B = np.array([[5, 6],
              [7, 8]])

print(np.dot(A, B))      # Matritsa ko'paytmasi
print(np.linalg.inv(A))  # Teskari matritsa
print(np.linalg.det(A))  # Determinant
print(np.linalg.eig(A))  # O'z qiymatlar va vektorlar
print(np.trace(A))       # Diagonal yig'indisi

# =====
# 7. Tasodifiy sonlar (numpy.random)
# =====
np.random.seed(42) # Takrorlanuvchanlik

print(np.random.rand(2, 3)) # 0-1 oralig'idagi tasodifiy sonlar
print(np.random.randn(3, 3)) # Normal taqsimot
print(np.random.randint(0, 10, 5)) # 0-9 oralig'ida 5 ta butun son
print(np.random.choice([1, 2, 3, 4])) # Tasodifiy tanlov

# =====
# 8. Ma'lumot turlari (dtype)
# =====
arr = np.array([1.5, 2.3, 3.1])
print(arr.dtype) # float64
print(arr.astype(int)) # butun songa aylantirish

# =====

```

```

# 9. Massivlarni birlashtirish va bo'lish
# =====
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])

print(np.concatenate((a, b))) # [1 2 3 4 5 6]
A = np.array([[1, 2], [3, 4]])
B = np.array([[5, 6], [7, 8]])

print(np.vstack((A, B)))      # Vertikal qo'shish
print(np.hstack((A, B)))      # Gorizontal qo'shish
print(np.split(a, 3))          # 3 qismga bo'lish

# =====
# 10. Tartiblash va unikal qiymatlar
# =====
arr = np.array([5, 2, 9, 1, 5, 6])
print(np.sort(arr))           # Tartiblangan massiv
print(np.argsort(arr))         # Indeksalar bo'yicha tartiblash
print(np.unique(arr))          # Takrorlanmas qiymatlar
print(np.flip(arr))            # Teskari tartib
print(np.roll(arr, 2))          # Siljitish

# =====
# 11. Fayllar bilan ishlash
# =====
arr = np.array([[1, 2, 3], [4, 5, 6]])

np.save('data.npy', arr)      # Saqlash (binary format)
np.savetxt('data.txt', arr)   # Saqlash (matnli format)

a1 = np.load('data.npy')      # Yuklash
a2 = np.loadtxt('data.txt')    # Yuklash

print(a1)
print(a2)

# =====
# 12. Statistika tahlil
# =====
x = np.array([1, 2, 3, 4, 5])
y = np.array([2, 4, 6, 8, 10])

print(np.corrcoef(x, y))      # Korrelyatsiya
print(np.cov(x, y))           # Kovariatsiya
print(np.clip(x, 2, 4))        # Chegaralash
print(np.sign(y - 5))          # Is'hora funksiyasi

# =====
# 13. Broadcasting va vektorlashtirish
# =====
arr = np.array([1, 2, 3])
print(arr + 5)                 # [6 7 8]
print(arr * 10)                 # [10 20 30]

import time

n = 1_000_000
data = range(n)

start = time.time()
result = [x**2 for x in data]
print("Python:", time.time() - start)

arr = np.arange(n)
start = time.time()
result = arr ** 2
print("NumPy:", time.time() - start)

# =====
# 14. Boshqa kutubxonalar bilan integratsiya
# =====

```

```
import pandas as pd
import matplotlib.pyplot as plt

# NumPy ma'lumotidan Pandas DataFrame yaratish
data = np.random.randn(5, 3)
df = pd.DataFrame(data, columns=['A', 'B', 'C'])
print(df)

# NumPy + Matplotlib vizualizatsiyasi
x = np.linspace(0, 10, 100)
y = np.sin(x)

plt.plot(x, y)
plt.title("NumPy + Matplotlib misoli")
plt.show()
```