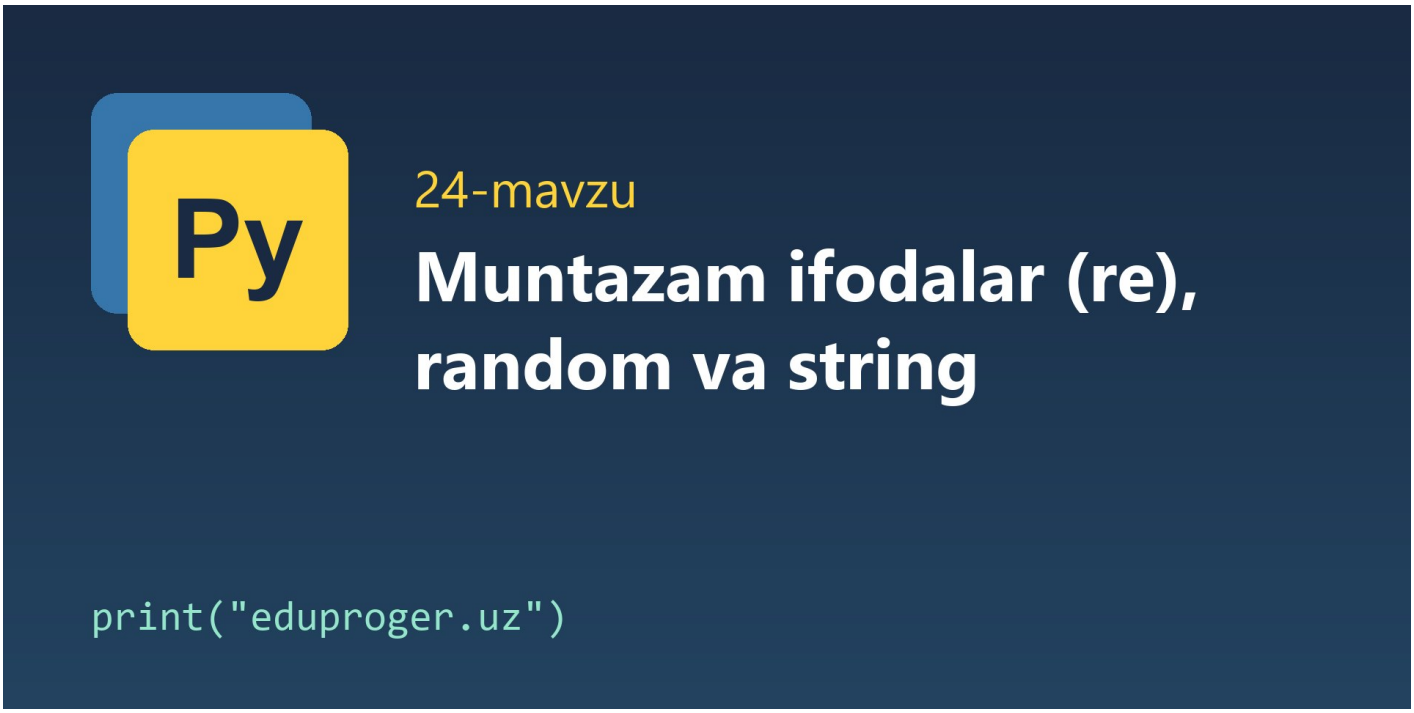
24. Muntazam ifodalar (re), random va string



Matndan telefon raqamlarini ajratish, emailni tekshirish, sanalarni topish — bularni oddiy satr metodlari bilan qilish qiyin. Buning uchun **muntazam ifodalar** (regular expressions, qisqacha regex) bor: ular matnda «naqsh» qidiradi. Pythonda — `re` moduli.

1. Birinchi naqsh

Naqsh oldiga `r` qo'yib yoziladi (r-satr: `\` belgisi o'zgarmasin). `\d` — bitta raqam, `+` — «bir yoki ko'p»:

```
import re
matn = "Buyurtma 125 ta, narxi 48000 so'm, 2026-yil"
print(re.findall(r"\d+", matn))
print(re.search(r"\d+", matn).group())
```

Natija:

```
['125', '48000', '2026']
125
```

2. Asosiy belgilar

Naqsh	Ma'nosi	Misol

`\d`	raqam (0–9)	`\d\d` → 42
`\w`	harf, raqam yoki _	`\w+` → so'z
`\s`	bo'sh joy (bo'shliq, tab, \n)	
`.`	istalgan bitta belgi (\n dan tashqari)	`a.c` → abc, a5c
`[abc]`	qavsdagi belgilardan biri	`[aeiou]` — unli
`[^0-9]`	qavsdagilardan BOSHQA	raqam emas
`^` / `$`	satr boshi / oxiri	`^\d+$` — faqat raqamlar
`\b`	so'z chegarasi	`\bcat\b`

Miqdor	Ma'nosi
`*`	0 yoki ko'p
`+`	1 yoki ko'p
`?`	0 yoki 1 (ixtiyoriy)
`{n}`	aynan n ta
`{n,m}`	n dan m gacha

3. re funksiyalari

Funksiya	Vazifasi
`re.search(n, s)`	Birinchi moslikni topadi (bo'lmasa None)
`re.match(n, s)`	Faqat satr BOSHIDAN tekshiradi
`re.fullmatch(n, s)`	Butun satr naqshga to'liq mosmi
`re.findall(n, s)`	Barcha mosliklar ro'yxati
`re.sub(n, y, s)`	Mosliklarni y bilan almashtiradi
`re.split(n, s)`	Naqsh bo'yicha bo'ladi

```
import re
print(re.sub(r"\s+", " ", "bu   matnda    ortiqcha   bo'shliq"))
print(re.split(r"[,;\s]+", "olma, nok;anor  uzum"))
print(re.findall(r"\b[A-Z]\w*", "Ali va Vali Toshkentga ketdi"))
print(bool(re.fullmatch(r"\d{4}", "2026")), bool(re.fullmatch(r"\d{4}", "26")))
```

Natija:

```
bu matnda ortiqcha bo'shliq
['olma', 'nok', 'anor', 'uzum']
['Ali', 'Vali', 'Toshkentga']
True False
```

4. Guruhlar: ma'lumotni ajratib olish

Qavs `( )` naqshning qismini **guruh** qiladi — keyin uni alohida olamiz:

```
import re
m = re.search(r"(\d{4})-(\d{2})-(\d{2})", "Tug'ilgan sana: 2010-05-17")
print(m.group(0), m.group(1), m.group(2), m.group(3))
for ism, ball in re.findall(r"(\w+):(\d+)", "Ali:78, Malika:95, Sardor:88"):
    print(ism, int(ball))
```

Natija:

```
2010-05-17 2010 05 17
Ali 78
Malika 95
Sardor 88
```

5. Amaliy tekshiruvlar

O'zbekiston telefon raqami

```
import re
naqsh = r"\+998[ -]?\d{2}[ -]?\d{3}[ -]?\d{2}[ -]?\d{2}"
for t in ["+998 90 123 45 67", "+998901234567", "998901234567", "+998 90 12"]:
    print(t, "->", bool(re.fullmatch(naqsh, t)))
```

Natija:

```
+998 90 123 45 67 -> True
+998901234567 -> True
998901234567 -> False
+998 90 12 -> False
```

Email

```
import re
naqsh = r"[\w.+-]+@[\w-]+\.[\w.]+"
matn = "Bog'lanish: info@eduproger.uz yoki ali.valiyev@gmail.com, xato@ manzil
print(re.findall(naqsh, matn))
```

Natija:

```
['info@eduproger.uz', 'ali.valiyev@gmail.com']
```

> **Eslatma**
>
> Murakkab naqshlarni bosqichma-bosqich quring va sinab ko'ring. Email va telefonning «mukammal» naqshi juda uzun bo'ladi — amaliyotda soddaroq tekshiruv + tasdiqlash xati/SMS ishlatiladi.

6. random moduli

Tasodifiy sonlar o'yinlar, test savollarini aralashtirish va sinov ma'lumotlari uchun kerak. `random.seed(n)` bir xil «tasodifiy» ketma-ketlikni qayta beradi — dasturni

tekshirishda qulay:

```
import random
random.seed(2026)
print(random.randint(1, 6))
print(random.choice(["tosh", "qaychi", "qog'oz"]))
print(random.sample(range(1, 50), 6))
kartalar = list("ABCDE")
random.shuffle(kartalar)
print(kartalar)
print(round(random.uniform(1, 10), 2))
```

Natija:

```
1
qaychi
[33, 49, 42, 7, 15, 39]
['A', 'B', 'C', 'D', 'E']
7.76
```

Funksiya	Natija
`randint(a, b)`	a..b butun son (b kiradi)
`random()`	0 ≤ x < 1 kasr son
`uniform(a, b)`	a..b kasr son
`choice(s)`	ketma-ketlikdan bitta element
`sample(s, k)`	k ta TAKRORSIZ element
`shuffle(s)`	ro'yxatni joyida aralashtiradi
`seed(n)`	takrorlanadigan ketma-ketlik

> **Eslatma**
>
> random parol va maxfiy kalitlar uchun YARAMAYDI — buning uchun `secrets` moduli bor: `secrets.token_hex(8)`.

7. string moduli

```
import string
print(string.ascii_lowercase)
print(string.digits)
print(string.punctuation)
matn = "Salom, dunyo! 2026."
print("".join(c for c in matn if c not in string.punctuation))
```

Natija:

```
abcdefghijklmnopqrstuvwxyz
0123456789
!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
Salom dunyo 2026
```

8. Xulosa

- re: `findall`, `search`, `fullmatch`, `sub`, `split`; `\d \w \s . [ ] ^ $` va `* + ? {n,m}`; guruhlar `( )`.
- random: `randint`, `choice`, `sample`, `shuffle`, `seed`; maxfiy narsalar uchun — `secrets`.
- string: tayyor belgilar to'plamlari (`ascii_lowercase`, `digits`, `punctuation`).