

Ev Tapşırığı — Fayl əsaslı Mini Mağaza Sistemi

Səviyyə: 1-ci kurs

İcra mühiti: 2 weeks

Saxlama: Yalnız fayllar (DB YOXDUR) — .json və ya sadə .txt istifadə oluna bilər

1) Məqsəd

Python dilində, fayl əsaslı (DB-siz) kiçik **mağaza sistemi** hazırlamaq. Sistem aşağıdakıları təmin etməlidir:

- İstifadəçi hesabı (username, password, balance) və **login cəhd limiti**: 3 səhv cəhd → **10 saniyə** cooldown, sonra yenidən 3 cəhd.
 - Kateqoriya → Məhsul → Miqdar seçimi; **seçildikdən sonra balans dərhal çıxılmaz**. İstifadəçi məhsulu **Səbətə** (gözləyən sifarişlər) əlavə edə və ya **Favoritlərə** ata bilər.
 - Səbət** bölməsindən **Checkout** zamanı ümumi məbləğ hesablanır; balans kifayət edərsə, onda məbləğ **Checkout-da** balansdan çıxılır və alış **tarixçəyə** qeyd olunur. Kifayət etmirsə, Checkout rədd edilir.
 - Favoritlər, Settings (şifrə dəyişmə), Tarixçə (log)** funksiyaları.
-

2) Funksional tələblər

2.1. İstifadəçi hesabı və giriş

- Hər istifadəçinin **username, password, balance** (AZN) sahələri var.
- Login zamanı **parol 3 dəfə səhv daxil edilərsə**, sistem **10 saniyəlik gözləmə** (cooldown) tətbiq edir; bu müddət bitdikdən sonra yenidən cəhd etmək mümkündür.
 - Göstəriş: Konsolda **timer/geri sayım** göstərmək tövsiyə olunur.
 - Nümunə axın: wrong → wrong → wrong → (10s gözlə) → yenidən 3 cəhd limiti.
- Uğurlu giriş → əsas menyu açılır.

2.2. Kateqoriyalar və məhsullar

- Ən azı 3–4 kateqoriya**, hər kateqoriyada **ən azı 3 məhsul**.
- Kateqoriya seçildikdə məhsullar **ad + qiymət** ilə siyahılanır.

- İstifadəçi məhsulu seçdikdə sistem aşağıdakı ardıcılığı göstərməlidir:
 1. **Məhsul adı və vahid qiymət**
 2. **Miqdar (ədəd)** soruşulur (müsbət tam ədəd)
 3. **Əməliyyat seçimi:**
 - B → **Səbətə əlavə et** (balans çıxılmaz)
 - F → **Favoritlərə əlavə et**
 - X → Ləğv et/geri dön
- **Qeyd:** Məhsul seçildikdən sonra **heç vaxt dərhal balansdan çıxma** aparılmaz; yalnız Səbətdə **Checkout** ediləndə çıxılır.

2.3. Səbət (gözləyənlər) və Checkout

- Səbətə əlavə olunan elementlər category, product, unit_price, qty, line_total = unit_price × qty sahələri ilə saxlanılır.
- Səbətdə əməliyyatlar:
 - list → səbəti göstər (hər sətir və **ümumi məbləğ**)
 - qty <id> <new_qty> → miqdarı dəyiş
 - remove <id> → sətiri sil
 - clear → səbəti təmizlə
 - checkout → **ümumi məbləği** hesabla və balansla müqayisə et
 - **Kifayət edirsə:** balansdan çıx, **purchases**-ə yaz, səbəti təmizlə, tarixçəyə **CHECKOUT_SUCCESS** qeyd et
 - **Kifayət etmir:** əməliyyatı rədd et, tarixçəyə **CHECKOUT_FAIL (insufficient balance)** yaz

2.4. Favoritlər

- İstənilən məhsul **Favoritlərə** əlavə/silinə bilər.
- Favoritlər siyahısından məhsulu bir kliklə **Səbətə** əlavə etmək mümkün olmalıdır (qty soruşularaq).

2.5. Settings (Şifrə dəyişmə)

Settings (Şifrə dəyişmə)

- **Köhnə şifrə** təsdiqlənmədən **yeni şifrə** təyin edilə bilməz.
- Minimum tələblər: ən azı 4 simvol (istəsən daha sərt qayda qoya bilərsən).

2.6. Tarixçə (log)

- Sistem aşağıdakı hadisələri tarixçə faylına yazır:
 - Uğurlu/uyğursuz **login** cəhdləri (timestamp ilə)
 - **Alış** əməliyyatları (məhsul, ədəd, məbləğ, balansdan çıxılan cəm)
 - **Şifrə dəyişmə** hadisəsi
- Tarixçə bölməsində ən son hərəkətlər göstərilir (məs: son 20 sətir).

3) İstifadəçi interfeysi (konsol menyu)

Login-dən sonra əsas menyu:

- 1) Kateqoriyalar
- 2) Səbətım (gözləyənlər)
- 3) Favoritlərim
- 4) Tarixçə
- 5) Settings (şifrəni dəyiş)
- 6) Balansımı göstər
- 0) Çıxış

Kateqoriyalar → kateqoriya seç → məhsullar siyahısı:

[Məhsul ID]	Ad	Qiymət (AZN)
-------------	----	--------------

1	T-Shirt	12.50
2	Hoodie	45.00
3	Jeans	60.00

...

Seçim:

ID daxil et → Məhsul detallı göstərilsin → Miqdar soruşulsun →

"B" (Səbətə əlavə) | "F" (Favorite əlavə) | "X" (Geri)

B → Geri qayıt

Səbət menyusu:

list | qty <id> <new_qty> | remove <id> | clear | checkout | back

4) Fayl əsaslı saxlanma (DB yoxdur)

Bütün fayllar proqram tərəfindən avtomatik yaranmalıdır (yoxdursa, default dəyərlərlə).

Təvsiyə olunan fayllar:

- users.json — istifadəçi siyahısı (və ya users.txt)
- products.json — kateqoriya və məhsullar
- basket_<username>.json — alınanlar (sifarişlər)
- favorites_<username>.json — favorit məhsullar
- history_<username>.log — hadisə jurnalı (sətir-sətir, mətn)
- session_<username>.json — login cəhd sayğacı və lock məlumatı (istəsən bunu users.json daxilində də saxlaya bilərsən)

4.1. JSON formatına nümunələr

users.json

```
[  
  {  
    "username": "student1",  
    "password": "1234",  
    "balance": 100.0,  
    "failed_attempts": 0,  
    "lock_until": null  
  }  
]
```

```
]
```

products.json

```
{
```

```
  "Geyimlər": [
```

```
    { "id": 1, "name": "T-Shirt", "price": 12.50 },
```

```
    { "id": 2, "name": "Hoodie", "price": 45.00 },
```

```
    { "id": 3, "name": "Jeans", "price": 60.00 }
```

```
  ],
```

```
  "Elektronika": [
```

```
    { "id": 1, "name": "Qulaqlıq", "price": 35.00 },
```

```
    { "id": 2, "name": "Powerbank", "price": 25.00 },
```

```
    { "id": 3, "name": "Sıçan", "price": 15.00 }
```

```
  ],
```

```
  "Kitablar": [
```

```
    { "id": 1, "name": "Algorithms 101", "price": 20.00 },
```

```
    { "id": 2, "name": "Clean Code", "price": 55.00 },
```

```
    { "id": 3, "name": "Python Basics", "price": 30.00 }
```

```
  ],
```

```
  "Aksesuarlar": [
```

```
    { "id": 1, "name": "Saat", "price": 80.00 },
```

```
    { "id": 2, "name": "Kəmər", "price": 18.00 },
```

```
    { "id": 3, "name": "Cüzdan", "price": 25.00 }
```

```
  ]
```

```
}
```

basket_student1.json (gözləyənlər)

```
[
```

```
{ "category": "Geyimlər", "product": "T-Shirt", "unit": 12.50, "qty": 2, "line_total": 25.00 },
{ "category": "Elektronika", "product": "Qulaqlıq", "unit": 35.00, "qty": 1, "line_total": 35.00 }
]
```

purchases_student1.json (tamamlanmış alışlar)

```
[
  { "ts": "2025-10-25T13:25:40", "items": [
    { "category": "Geyimlər", "product": "T-Shirt", "unit": 12.50, "qty": 2, "line_total": 25.00 },
    { "category": "Elektronika", "product": "Qulaqlıq", "unit": 35.00, "qty": 1, "line_total": 35.00 }
  ], "total": 60.00 }
]
```

favorites_student1.json

```
[ { "category": "Geyimlər", "id": 2, "name": "Hoodie", "price": 45.00 } ]
```

history_student1.log

[2025-10-25 13:18:01] LOGIN_FAIL (wrong password)

[2025-10-25 13:18:09] LOGIN_FAIL (wrong password) → COOLDOWN 10s

[2025-10-25 13:18:20] LOGIN_SUCCESS

[2025-10-25 13:20:12] BASKET_ADD (Geyimlər/T-Shirt x2)

[2025-10-25 13:22:33] FAVORITE_ADD (Elektronika/Powerbank)

[2025-10-25 13:24:10] CHECKOUT_SUCCESS total=60.00 | balance: 100.00 → 40.00

Qeyd: purchases_*.json tamamlanmış alışları saxlayır; səbət isə yalnız **gözləyən** elementlər üçündür.

5) Qaydalar və yoxlamalar

- **Miqdar** yalnız **müsbət tam ədəd** olmalıdır (0 və ya mənfi → rədd).
- Məbləğ hesablaması: $\text{total} = \text{price} \times \text{qty}$ → balansdan **dərhal** çıxılır.
- Balans **kifayət etmirsə**, alış **baş vermir** (tarixçəyə "FAILED_PURCHASE" kimi düşə bilər).

- **Favoritə əlavə** balansı dəyişmir.
- **Şifrə dəyişmə** üçün köhnə şifrə düzgün verilməlidir.
- Fayllar oxunmur/format səhvdirsə, **təhlükəsiz default** dəyərlərlə yenidən yaradılmalıdır.

6) Test ssenariləri (minimum)

1. **Login limiti:** 3 səhv → 10s cooldown → sonra doğru şifrə ilə giriş.
2. **Səbət axını:** Geyimlərdən T-Shirt ×2 **Səbətə əlavə et** → balans **dəyişməməlidir**.
3. **Checkout uğurlu:** Səbətdəki cəmi hesabla → balans kifayət edirsə **Checkout et** → balansdan çıx → purchases-a yaz.
4. **Checkout uğursuz:** Səbətə balansdan çox məbləğlik elementlər yığ → checkout → **rədd** (səbət dəyişməz qalır).
5. **Favorit → Səbət:** Favoritə əlavə et → Favoritdən Səbətə at → Checkout et.
6. **Şifrə dəyişmə:** Köhnə şifrəni təsdiq et → yenisini təyin et → çıxış → yeni şifrə ilə giriş.
7. **Tarixçə:** Yuxarıdakı bütün hadisələr logda ardıcılıqla görünsün.
8. **Persistensiya:** Proqram bağlanıb-açıldıqdan sonra səbət, favoritlər, purchases, balans **qorunsun**.

7) Qiymətləndirmə (100 bal)

- Funksionallıq (login/cooldown, alış, favorit, tarixçə, settings) — **40**
- Fayl I/O və məlumatların düzgün saxlanması — **20**
- Məlumat yoxlamaları və xəta idarəsi — **15**
- İstifadəçi təcrübəsi (aydın menyu, mesajlar, taymer) — **10**
- Sənədləşmə (README, nümunə axınlar, fayl formatları) — **10**
- Kodun səliqəsi (funksiyalara bölünmə, adlandırma) — **5**

8) Təhvil formatı

- README.md — işə salma, menyu nümunələri, fayl formatı izahı
 - default_data/ — **hazır users.json, products.json**
 - sample_runs.txt — **ən az 8 addımlı 2 tam ssenari** (giriş → alış → favorit → tarixçə → çıxış)
 - Proqram faylları (məs: main.py və ya app.js)
-

9) Bonus (istəyə bağlı)

- **Balans artırma** (top-up) funksiyası
 - **Stok** izləmə: stok qurtarsa almaq mümkün olmasın
 - **Axtarış/Filtr**: məhsul adına görə filtr
 - **CSV export**: Səbəti .csv-ə ixrac
 - **Sadə “undo”**: son alışdan 30 saniyə içində geri qaytarma
-

İstəyə görə fayl adlarını data/ qovluğunda saxla, fayl mövcud deyilsə **default** dəyərlərlə avtomatik yarat.