

Python Harçlık ve Bütçe Takip Uygulaması

ÖĞRETMEN KILAVUZU

Kategori	Python	Süre	2–3 ders saati
Seviye	Başlangıç	Hedef	Başlangıç seviye lise öğrencileri

Amaç

- Liste, fonksiyon ve toplama/çıkarma işlemlerini günlük yaşam problemiyle uygulamak.

Problem durumu

- Gelir ve giderlerin ayrı kaydedilip kalan bütçenin hatasız hesaplanması gerekir

Kazanımlar

- Gelir/gideri sınıflandırır.
- Fonksiyonla kayıt ekler.
- Bakiye hesaplar.

Hazırlık

- Python Harçlık ve Bütçe Takip Uygulaması için örnek girdileri, güvenlik sınırlarını ve test ölçütlerini dersten önce kontrol etmek.

Araç / malzemeler

- Python 3
- Sahte bütçe senaryoları

Uygulama

- Veri yapısını seç.
- Kayıt fonksiyonu yaz.
- Bakiye fonksiyonu ekle.
- Geçersiz tutarı kontrol et.
- Özet üret.

Test aşaması

- Yalnız gelir
- Gelir+gider
- Negatif tutar
- Sıfır kayıt

Olası hatalar ve çözümler

- Gider işareti yanlış: Girdiyi veya bağlantıyı tek tek doğrula; değişiklikten sonra aynı testi yeniden çalıştır.
- Negatif değer kabulü: Girdiyi veya bağlantıyı tek tek doğrula; değişiklikten sonra aynı testi yeniden çalıştır.
- Gerçek finans verisi kullanımı: Girdiyi veya bağlantıyı tek tek doğrula; değişiklikten sonra aynı testi yeniden çalıştır.

Güvenlik

- Gerçek hesap, kart veya ödeme bilgisi kullanmayın.

Öğretmen yönergeleri

- Hazır çözümü vermeden önce öğrenciden tahminini ve denediği adımı açıklamasını isteyin.
- Sonucu yalnız çalışıyor/çalışmıyor olarak değil; test kanıtı ve açıklamayla değerlendirin.

Geliştirme önerileri

- Kategori özeti
- Aylık hedef
- Metin tabanlı grafik

Bütçe takip iskeleti

Tüm değerler eğitim amaçlı sahte örneklerdir; gerçek finansal bilgi kullanılmaz.

```
hareketler = []
def kaydet(tur, tutar):
    if tur in ("gelir", "gider") and tutar > 0: hareketler.append((tur, tutar))
def bakiye():
    return sum(t if tur == "gelir" else -t for tur, t in hareketler)
kaydet("gelir", 500); kaydet("gider", 120)
print("Kalan bütçe:", bakiye())
```