

استفاده از JSON: زبان مشترک داده‌ها

آیا تا به حال به این فکر کرده‌اید که چگونه اطلاعات را به گونه‌ای ذخیره و به اشتراک بگذارید که هم ساده، هم قابل فهم و هم سریع باشد؟ JSON (JavaScript Object Notation) دقیقاً برای همین منظور طراحی شده است. JSON یک فرمت سبک و قابل خواندن برای تبادل داده است که در سراسر دنیای برنامه‌نویسی، از توسعه نرم‌افزار تا تبادل اطلاعات در وب، استفاده می‌شود.

فرض کنید می‌خواهیم اطلاعات یک کاربر را در یک فایل ذخیره کنیم. این اطلاعات شامل نام، ایمیل، علایق و سایر جزئیات است. با استفاده از JSON، می‌توانیم این اطلاعات را به راحتی در یک فایل با پسوند `.json` ذخیره کنیم.

user_data.json:

```
1  {  
2    "username": "john_doe",  
3    "email": "john.doe@example.com",  
4    "interests": ["programming", "reading", "traveling"],  
5    "is_active": True,  
6    "profile": {  
7      "city": "London",  
8      "country": "UK"  
9    }  
10 }
```

ساختار JSON

- **کلید-مقدار (Key-Value):** JSON از ساختار کلید-مقدار برای ذخیره داده‌ها استفاده می‌کند، مشابه دیکشنری‌ها در پایتون.
- **انواع داده‌ها:** JSON از انواع داده‌های مختلفی پشتیبانی می‌کند، از جمله:
 - **رشته (String):** متن بین دو علامت نقل قول (") قرار می‌گیرد.
 - **عدد (Number):** اعداد صحیح و اعشاری.

- **بولی (Boolean):** مقادیر true یا false .
- **آرایه (Array):** لیستی از مقادیر بین دو براکت ([]) .
- **دیکشنری (Dictionary/Object):** مجموعه‌ای از کلید-مقدارها بین دو آکولاد ({ }) .
- **Null:** مقدار null .

استفاده از JSON در پایتون

پایتون کتابخانه استاندارد json را برای کار با داده‌های JSON ارائه می‌دهد.

- `json.load()` : برای خواندن داده‌های JSON از یک فایل و تبدیل آن‌ها به دیکشنری پایتون.

```

1 import json
2
3 with open('user_data.json', 'r') as file:
4     data = json.load(file)
5
6 print(data['username']) # Output: john_doe

```

`json.dump()` : برای تبدیل دیکشنری پایتون به داده‌های JSON و نوشتن آن‌ها در یک فایل.

```

1 import json
2
3 user = {
4     "username": "jane_smith",
5     "email": "jane.smith@example.com"
6 }
7
8 with open('new_user.json', 'w') as file:
9     json.dump(user, file)

```

`json.loads()` : برای تبدیل رشته JSON به دیکشنری پایتون.

```

1 import json
2
3

```

```
4 | json_string = '{"city": "Paris", "country": "France"}'  
5 | data = json.loads(json_string)  
6 |  
   | print(data['city']) # Output: Paris
```

• `json.dumps()`: برای تبدیل دیکشنری پایتون به رشته JSON.

```
1 | import json  
2 |  
3 | user = {"age": 30, "city": "Berlin"}  
4 | json_string = json.dumps(user)  
5 |  
6 | print(json_string) # Output: {"age": 30, "city": "Berlin"}
```

نکات

- داده‌های JSON بسیار شبیه به دیکشنری‌های پایتون هستند، اما تفاوت‌های کوچکی در قواعد نحوی دارند.
- از `json.load()` و `json.dump()` برای کار با فایل‌های JSON و از `json.loads()` و `json.dumps()` برای کار با رشته‌های JSON استفاده کنید.
- در صورت بروز خطا در تجزیه JSON، خطای `json.JSONDecodeError` رخ می‌دهد.