

## 例外処理・指南編

---



できた。

```
# -*- coding: cp932 -*-
import datetime
import sys

day=["月曜日", "火曜日", "水曜日", "木曜日", "金曜日", "土曜日", "日曜日"]
date_form=raw_input("Date:")

while date_form!="bye":
    date_form

    try:
        d=date_form.split("/")
    #
    #正しい入力時の処理
    #
        if len(d)==3:
            date_output=datetime.date(int(d[0]), int(d[1]), int(d[2]))
            print date_output, "は", day[date_output.weekday()]
            date_form=raw_input("Date:")
    #
    #年月しか入力しないなど形式がおかしい場合の処理
    #
        else:
            print "誤った入力です"
            date_form=raw_input("Date:")
    #
    #範囲外もしくは存在しない年月日を入力した場合の例外処理
    #
    except (ValueError, OverflowError):
        print "誤った入力です"
        date_form=raw_input("Date:")

sys.exit()
```

お疲れ様でした。色々と入力して試してみましたが、かなり無理なデータを入力しても異常終了しませんでしたので大丈夫と思います。

まともな入力...

- 2010/11/20 → 土曜日
- 2011/9/2 → 金曜日
- 1750/01/02 → 金曜日

変な入力...

- 2010/9/31
- 2010/9/30/3
- 2010/9
- 2010.9.30
- 2010/9/30 ←全角数字

スクリプトを順に見てまいります。

まずはあらかじめdayというリストを作って、あとで曜日を日本語に直すための準備をしました。

で、入力をループで何回も行う処理を、こんな書き方で表現してくれました。

```
-----  
date_form=raw_input("Date:")  
while date_form!="bye":  
    #  
    # ... ここで何か意味のあることをする ...  
    #  
    date_form=raw_input("Date:")  
sys.exit()  
-----
```

ループに入るときに、まず一回目の入力。ループはwhileで制御されており、これの終了条件は、キーボードからの入力内容を格納する date\_form 変数に 'bye' が入っているとき。終了条件を満たさない間は永久ループである、と。いいですね。

このやり方で充分ですが、ちょっとループの書き方を工夫すると、raw\_input を行う場所が一回だけで済むようにもなりますので、どうぞご参考に。下みたいな感じで：

```
-----  
while True  
    date_form=raw_input("Date:")  
    if date_form == "bye":  
        break  
    #  
    # ... ここで何か意味のあることをする ...  
    #  
-----
```

breakを導入したのがミソです。

あと、スクリプトの最後に sys.exit() を書いてくれましたが、まあ、最終行まで到達したらもうpythonの実行は終わるんですから、今回のような例ではここは省略しちゃってもよかったですね。じゃあなんで sys.exit なんか教えたんだ！と言われそうです。すいません。スクリプトの途中でいきなりpythonを終了させちゃうような書き方もありますんで、そういうときは有用ですよ。

