

## ثبت نام گروهی

دنیا استارت‌آپی تاسیس کرده و قبل از اینکه وبسایت استارت‌آپ آماده شود، صفحه دیگری برای پیش ثبت‌نام کاربران در نظر گرفته‌است. در صفحه‌ی پیش ثبت‌نام از کاربران ایمیل و رمز عبور آنها پرسیده شده‌است ولی در حال حاضر که وبسایت آماده است، ۲ کار باید انجام شود. یکی اینکه درستی تک-تک ایمیل‌ها قبل از وارد شدن به پایگاه داده بررسی شود.

- ایمیل‌ها باید در قالب `username@websiteName.extension` باشند.
- قسمت `username` باید تنها شامل حروف، اعداد، خط فاصله ( - )، و زیرخط ( \_ ) باشد.
- `websiteName` باید تنها شامل حروف و اعداد باشد.
- حداکثر طول `extension` برابر ۳ است.

دیگر اینکه، در وبسایت به خاطر امنیت کاربران رمز عبور آنها به صورت خام نگهداری نمی‌شود بلکه ابتدا با الگوریتم MD5 هاش شده و مقدار هاش شده‌ی آن نگهداری می‌شود. الگوریتم MD5 به گونه‌ای عمل می‌کند که اگر کاربر هنگام ورود دوباره رمز عبور خود را وارد کند، مقدار هاش شده آن نیز با مقدار ذخیره شده یکسان خواهد بود پس می‌توانیم به جای مقایسه رمز عبورهای خام مقدار هاش شده‌ی آنها را ذخیره کنیم.

اگر متن شما در متغیر `text` باشد، می‌توانید به شکل زیر از الگوریتم MD5 استفاده کنید.

```
1 import hashlib
2 hashlib.md5(text.encode()).hexdigest()
```

فایل سوال به شکل زیر است.

```
1 import hashlib
2 import re
3 def is_email_valid(email):
4     #your code
5
6 def hash_password(password):
7     #your code
8
9 def process_data(data):
10     data = filter(lambda pair: is_email_valid(pair[0]), data)
11     data = list(map(lambda t: (t[0], hash_password(t[1])), data))
12     return data
```

حال شما باید دو تابع بالا را تکمیل کنید (تابع `process_data` را تغییر ندهید).

۱. `is_email_valid` این تابع در صورتی که ایمیل مورد قبول باشد، `True` و در غیر این صورت `False` برمی‌گرداند. در نهایت این تابع به `filter` داده می‌شود تا فقط ایمیل‌های مورد قبول بمانند.

۲. `hash_password` مقدار هش شده رمز عبور بر اساس الگوریتم MD5 را برمی‌گرداند. در نهایت به تابع `map` داده می‌شود تا همه رمزهای عبور هش شوند.

## مثال

کد زیر را اجرا می‌کنیم :

```
1 | users = [  
2 |     ["email", "2A-^N#Hc"],  
3 |     ["mohamadreza33@gmailcom", "tQ$jk2?v"],  
4 |     ["alish007@gmail.com", "BBWyV#bh"],  
5 |     ["poooyan12/5@gmail.com", "lWuw??xB"]  
6 | ]
```

انتظار می‌رود در ادامه کد بالا تابع `process_data` به صورت زیر کار کنند:

```
1 | >>> process_data(users)  
2 | [( 'alish007@gmail.com', 'd07260035ab152ade1a4aa5c87f8d89f' )]
```