به نام خدا

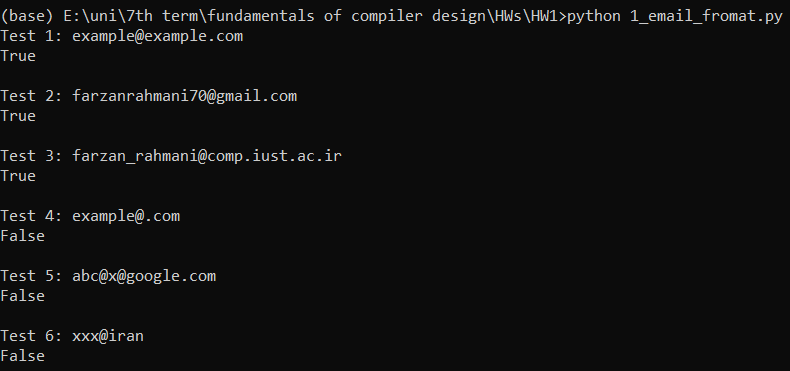
# تمرین سری اول درس اصول طراحی کامپایلر دکتر سعید پارسا

# فرزان رحمانی 99521271

## سوال 1

### الف

نتایج:



توضیح pattern:  
 • ^ : شروع خط.

• [a-zA-Z0-9\_.+-]+ : یک یا چند (+) از نویسه های داخل پرانتز ([]). این کاراکترها می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9)، زیرخط (\_)، نقطه (.)، به علاوه (+)، یا خط فاصله (-) باشند.

• @ : با نماد @ مطابقت دارد.

• [a-zA-Z0-9-]+ : یک یا چند (+) از نویسه های داخل پرانتز ([]). این کاراکترها می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9) یا خط فاصله (-) باشند.

• \. : با علامت نقطه (.) مطابقت دارد.

• [a-zA-Z0-9-.]+ : یک یا چند (+) از نویسه های داخل پرانتز ([]). این کاراکترها می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9)، نقطه (.)، یا خط فاصله (-) باشند.

• $ : انتهای خط.

توضیح کد:  
کد ضمیمه شده یک اسکریپت پایتون است که یک تابع "test\_email(email)" را برای اعتبارسنجی آدرس ایمیل با استفاده از عبارات منظم (regex) تعریف می کند. در اینجا خلاصه ای از آنچه که کد انجام می دهد آورده شده است:

1. اسکریپت ماژول «re» را وارد می کند که از عبارات منظم در پایتون پشتیبانی می کند.

2. تابع «تست\_ایمیل(ایمیل)» یک آدرس ایمیل را به عنوان ورودی می گیرد و اگر ایمیل معتبر باشد، «درست» و در غیر این صورت «نادرست» را برمی گرداند.

3. تابع یک الگوی عبارت منظم را برای مطابقت با آدرس ایمیل تعریف می کند. الگو در متغیر "الگو" ذخیره می شود.

4. الگوی عبارت منظم به شرح زیر است:

```

^[a-zA-Z0-9\_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-.]+$

```

- `^` شروع خط را نشان می دهد.

- «[a-zA-Z0-9\_.+-]+» با یک یا چند نویسه مطابقت دارد که می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9)، زیرخط (\_)، نقطه ( .)، به علاوه (+)، یا خط فاصله (-).

- `@` با نماد @ مطابقت دارد.

- «[a-zA-Z0-9-]+» با یک یا چند نویسه مطابقت دارد که می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9)، یا خط فاصله (-) باشند.

- `\.`، با علامت معکوس فرار کرد، با نماد نقطه (.) مطابقت دارد.

- «[a-zA-Z0-9-.]+» با یک یا چند نویسه مطابقت دارد که می توانند حروف کوچک (a-z)، حروف بزرگ (A-Z)، اعداد (0-9)، نقطه (.)، یا خط فاصله (. -).

- `$` پایان خط را نشان می دهد.

5. این تابع از «re.match(الگو، ایمیل)» استفاده می کند تا بررسی کند که «ایمیل» ارائه شده با الگوی تعریف شده مطابقت دارد یا خیر. اگر مطابقت وجود داشته باشد، «درست» را برمی‌گرداند. در غیر این صورت، "False" را برمی گرداند.

6. سپس اسکریپت تابع «test\_email()» را با آدرس های ایمیل مختلف آزمایش می کند.

- سه تست اول ("Test 1"، "Test 2" و "Test 3") آدرس های ایمیل معتبر هستند و انتظار می رود که تابع برای همه آنها "True" را برگرداند.

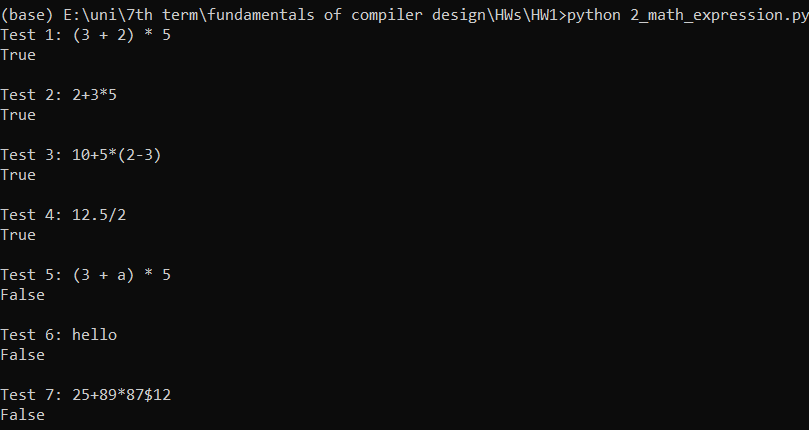
- سه تست آخر ("تست 4"، "تست 5" و "آزمون 6") آدرس های ایمیل نامعتبر هستند و انتظار می رود این تابع برای همه آنها "False" را برگرداند.

7. اسکریپت توضیحات تست و نتایج مربوطه را چاپ می کند.

توجه داشته باشید که عبارت معمولی استفاده شده در این قطعه کد یک نسخه ساده شده است و ممکن است همه قالب‌های آدرس ایمیل معتبر ممکن را پوشش ندهد. اعتبارسنجی ایمیل می‌تواند پیچیده باشد و توصیه می‌شود از یک کتابخانه یا سرویس جامع و آزمایش‌شده برای اعتبارسنجی ایمیل در محیط تولید استفاده کنید.

### ب

نتایج:



توضیح pattern:

* ^: شروع یک خط را بیان می کند.
* [0-9+\-\*/.()]: با هر کاراکتری که یک رقم است، «+»، «-»، «\*»، «/»، «.»، «(»، یا «)» مطابقت دارد.
* +: کمیتی است که یک یا چند بار با عنصر قبلی مطابقت دارد.
* $: پایان یک خط را نشان می دهد.

توضیح کد:

کد ضمیمه شده یک اسکریپت پایتون است که تابع «test\_math\_expr(expr)» را برای تأیید یک عبارت ریاضی با استفاده از عبارات منظم (regex) تعریف می کند. در اینجا توضیحی درباره کاری که کد انجام می دهد آورده شده است:

1. اسکریپت ماژول «re» را وارد می کند که از عبارات منظم در پایتون پشتیبانی می کند.

2. تابع «test\_math\_expr(expr)» یک عبارت ریاضی را به عنوان ورودی می گیرد و اگر عبارت معتبر باشد، «True» و در غیر این صورت «False» را برمی گرداند.

3. تابع یک الگوی عبارت منظم را برای مطابقت با عبارت ریاضی تعریف می کند. الگو در متغیر "الگو" ذخیره می شود.

4. الگوی عبارت منظم به شرح زیر است:

```

^[0-9+\-\*/.() ]+$

```

- `^` شروع یک خط را بیان می کند.

- «[0-9+\-\*/.()]» با هر کاراکتری که رقمی است (0-9)، «+»، «-»، «\*»، «/»، «.»، «منطبق است. ('، یا ')'.

- `+` کمیتی است که یک یا چند بار با عنصر قبلی مطابقت دارد.

- `$` پایان یک خط را بیان می کند.

5. تابع از «re.match(pattern, expr)» استفاده می کند تا بررسی کند که «expr» ارائه شده با الگوی تعریف شده مطابقت دارد یا خیر. اگر مطابقت وجود داشته باشد، «درست» را برمی‌گرداند. در غیر این صورت، "False" را برمی گرداند.

6. سپس اسکریپت تابع «test\_math\_expr()» را با عبارات مختلف ریاضی آزمایش می کند.

- چهار تست اول ("Test 1"، "Test 2"، "Test 3" و "Test 4") عبارات ریاضی معتبر هستند و انتظار می رود که تابع برای همه آنها "True" را برگرداند.

- تست های پنجم و ششم ("Test 5" و "Test 6") عبارات ریاضی نامعتبر هستند و انتظار می رود که تابع برای هر دوی آنها "False" را برگرداند.

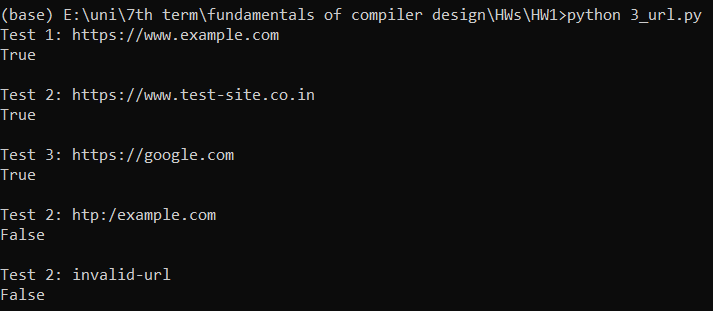
- تست هفتم ('Test 7') مورد جالبی است که در آن عبارت حاوی یک کاراکتر نامعتبر است ('$') اما همچنان انتظار می رود که تابع 'True' را برگرداند. این به این دلیل است که الگوی regex با کاراکترهای '+' و '\*' و همچنین سایر کاراکترهای مجاز مطابقت دارد و اجازه می دهد تا عبارت معتبر در نظر گرفته شود.

7. اسکریپت توضیحات تست و نتایج مربوطه را چاپ می کند.

توجه داشته باشید که عبارت منظم استفاده شده در این قطعه کد یک نسخه ساده شده است و ممکن است همه قالب‌های بیان ریاضی معتبر را پوشش ندهد. اعتبار سنجی عبارت ریاضی می تواند پیچیده و وابسته به زمینه باشد، با در نظر گرفتن عواملی مانند اولویت عملگر و وجود متغیرها. اگر به اعتبارسنجی عبارت ریاضی قوی‌تری نیاز دارید، توصیه می‌شود از یک کتابخانه تخصصی استفاده کنید یا تجزیه‌کننده‌ای را ارزیابی کنید که بتواند نیازهای خاص مورد استفاده شما را برطرف کند.

### ج

نتایج:



توضیح pattern:

* ^ شروع یک خط را بیان می کند.
* (http|https) با "http" یا "https" مطابقت دارد.
* :\/\/ به معنای واقعی کلمه با '://' مطابقت دارد.
* [a-zA-Z0-9-\.]+ با یک یا چند نویسه که حروف (بزرگ یا کوچک)، رقم، «-»، یا «.» هستند مطابقت دارد.
* \. با کاراکتر تحت اللفظی «.» مطابقت دارد.
* [a-zA-Z]{2,} با دو یا چند نویسه که حروف هستند (یا بزرگ یا کوچک) مطابقت دارد.
* (\/.\*)؟ یک گروه اختیاری است که با "/" به دنبال صفر یا بیشتر از هر کاراکتری مطابقت دارد.
* $ پایان یک خط را نشان می دهد.

توضیح کد:

کد ضمیمه شده یک اسکریپت پایتون است که تابع «test\_url(url)» را برای اعتبارسنجی URL با استفاده از عبارات منظم (regex) تعریف می کند. در اینجا توضیحی درباره کاری که کد انجام می دهد آورده شده است:

1. اسکریپت ماژول «re» را وارد می کند که از عبارات منظم در پایتون پشتیبانی می کند.

2. تابع «test\_url(url)» یک URL را به عنوان ورودی می گیرد و اگر URL معتبر باشد، «True» و در غیر این صورت «False» را برمی گرداند.

3. تابع یک الگوی عبارت منظم را برای مطابقت با URL تعریف می کند. الگو در متغیر "الگو" ذخیره می شود.

4. الگوی عبارت منظم به شرح زیر است:

````

^(http|https):\/\/[a-zA-Z0-9-\.]+\.[a-zA-Z]{2,}(\/.\*)؟$

```

- `^` شروع یک خط را بیان می کند.

- «(http|https)» با «http» یا «https» مطابقت دارد.

- `:\/\/` به معنای واقعی کلمه با '://' مطابقت دارد.

- «[a-zA-Z0-9-\.]+» با یک یا چند نویسه که حروف (بزرگ یا کوچک)، رقم، «-» یا «.» هستند مطابقت دارد.

- «\.» با کاراکتر تحت اللفظی «.» مطابقت دارد.

- `[a-zA-Z]{2,}` با دو یا چند نویسه که حروف هستند (یا بزرگ یا کوچک) مطابقت دارد.

- "(\/.\*)؟" یک گروه اختیاری است که با "/" به دنبال صفر یا بیشتر از هر کاراکتری مطابقت دارد.

- `$` پایان یک خط را بیان می کند.

5. این تابع از «re.match(الگو، url)» استفاده می کند تا بررسی کند که «url» ارائه شده با الگوی تعریف شده مطابقت دارد یا خیر. اگر مطابقت وجود داشته باشد، «درست» را برمی‌گرداند. در غیر این صورت، "False" را برمی گرداند.

6. سپس اسکریپت تابع «test\_url()» را با URL های مختلف آزمایش می کند.

- سه تست اول ("Test 1"، "Test 2" و "Test 3") URL های معتبری هستند و انتظار می رود که تابع برای همه آنها "True" را برگرداند.

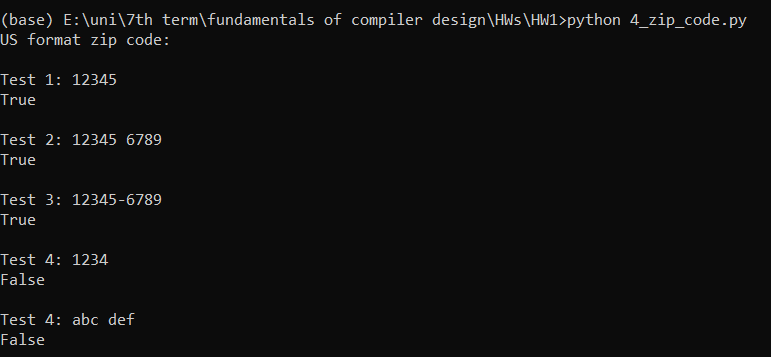
- دو آزمایش آخر ('Test 4' و 'Test 5') URL های نامعتبر هستند و انتظار می رود که تابع برای هر دوی آنها 'False' را برگرداند.

7. اسکریپت توضیحات تست و نتایج مربوطه را چاپ می کند.

توجه داشته باشید که عبارت منظم استفاده شده در این قطعه کد یک نسخه ساده شده است و ممکن است همه قالب‌های URL معتبر ممکن را پوشش ندهد. اعتبار سنجی URL با در نظر گرفتن طرح های مختلف، زیر دامنه ها، پارامترهای پرس و جو و سایر مؤلفه ها می تواند پیچیده باشد. اگر به اعتبارسنجی URL قوی‌تری نیاز دارید، توصیه می‌شود از یک کتابخانه تخصصی استفاده کنید یا یک راه‌حل تجزیه و اعتبار URL را ارزیابی کنید که بتواند نیازهای خاص مورد استفاده شما را برطرف کند.

### د

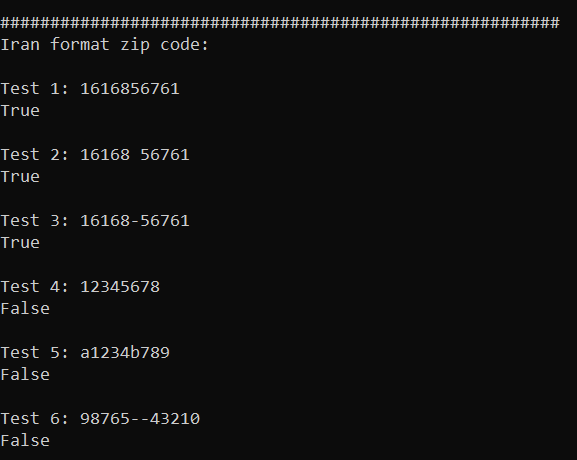
نتایج:



توضیح pattern آمریکا:

* ^ : شروع خط.
* \d{5} : دقیقاً پنج رقم.
* (?:[-\s]\d{4})؟ : یک گروه اختیاری (؟) که شامل خط فاصله (-) یا فاصله (\s) و دقیقاً چهار رقم (\d{4}) است.
* $ : انتهای خط.

نتایج:



توضیحات pattern ایران:

* ^ : شروع خط.
* \d{5} : دقیقاً پنج رقم، نشان دهنده کد حرکت است.
* [-\s]؟ : یک گروه اختیاری (?) که شامل خط فاصله (-) یا فاصله (\s) است.
* \d{5}: دقیقاً پنج رقم، نشان دهنده کد توزیع است.
* $ : انتهای خط.

توضیح کد:

کد ضمیمه شده یک اسکریپت پایتون است که دو تابع را تعریف می کند: "test\_zip(zip\_code)" و "test\_iranian\_zip(zip\_code)". این توابع برای اعتبارسنجی کدهای پستی در قالب های مختلف با استفاده از عبارات منظم (regex) استفاده می شود. در اینجا توضیحی درباره کاری که کد انجام می دهد آورده شده است:

1. اسکریپت ماژول «re» را وارد می کند که از عبارات منظم در پایتون پشتیبانی می کند.

2. تابع «test\_zip(zip\_code)» یک کد پستی را به عنوان ورودی می گیرد و اگر کد پستی در قالب ایالات متحده معتبر باشد، «درست» و در غیر این صورت «نادرست» را برمی گرداند.

3. تابع یک الگوی عبارت منظم را برای مطابقت با کد پستی تعریف می کند. الگو در متغیر "الگو" ذخیره می شود.

4. الگوی عبارت منظم برای قالب ایالات متحده به شرح زیر است:

```

^\d{5}(?:[-\s]\d{4})؟$

```

- `^` شروع یک خط را بیان می کند.

- `\d{5}` دقیقاً با پنج رقم مطابقت دارد.

- «(?:[-\s]\d{4})؟» یک گروه اختیاری است که شامل خط فاصله (-) یا فاصله (\s) و دقیقاً چهار رقم (\d{4}) است.

- `$` پایان یک خط را بیان می کند.

5. این تابع از «re.match(الگو، کد پستی)» استفاده می کند تا بررسی کند که «کد\_زیپ» ارائه شده با الگوی تعریف شده مطابقت دارد یا خیر. اگر مطابقت وجود داشته باشد، «درست» را برمی‌گرداند. در غیر این صورت، "False" را برمی گرداند.

6. اسکریپت تابع «test\_zip()» را با کدهای پستی مختلف در قالب ایالات متحده آزمایش می کند.

- سه تست اول ("Test 1"، "Test 2" و "Test 3") کدهای پستی معتبر در قالب ایالات متحده هستند و انتظار می رود که عملکرد برای همه آنها "True" را برگرداند.

- دو تست آخر ("تست 4" و "آزمون 5") کدهای پستی نامعتبر هستند و انتظار می رود این تابع برای هر دوی آنها "False" را برگرداند.

7. سپس اسکریپت تابع «test\_iranian\_zip(zip\_code)» را تعریف می‌کند، که از ساختاری مشابه تابع قبلی پیروی می‌کند اما مخصوصاً برای اعتبارسنجی کدهای پستی در قالب ایران است.

8. الگوی بیان منظم برای قالب ایران به شرح زیر است:

```

^\d{5}[-\s]؟\d{5}$

```

- `^` شروع یک خط را بیان می کند.

- `\d{5}` دقیقاً با پنج رقم مطابقت دارد که نشان دهنده کد عزیمت است.

- `[-\s]؟` یک گروه اختیاری است که شامل خط فاصله (-) یا فاصله (\s) است.

- `\d{5}` دقیقاً با پنج رقم مطابقت دارد که نشان دهنده کد توزیع است.

- `$` پایان یک خط را بیان می کند.

9. تابع `test\_iranian\_zip()` با کدهای پستی مختلف در قالب ایران تست شده است.

- سه تست اول ("تست 1"، "تست 2" و "آزمون 3") کدهای پستی معتبر در قالب ایران هستند و انتظار می رود تابع برای همه آنها "True" را برگرداند.

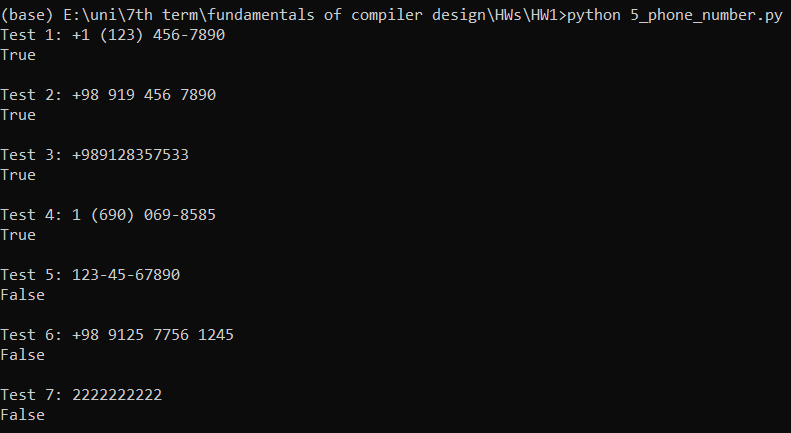
- سه تست آخر ("تست 4"، "تست 5" و "آزمون 6") کدهای پستی نامعتبر هستند و انتظار می رود این تابع برای همه آنها "False" را برگرداند.

10. اسکریپت توضیحات تست و نتایج مربوطه را چاپ می کند.

توجه داشته باشید که عبارات منظم استفاده شده در این قطعه کد، نسخه های ساده شده هستند و ممکن است همه قالب های کد پستی معتبر ممکن برای ایالات متحده و ایران را پوشش ندهند. اعتبار سنجی کد پستی می تواند بر اساس الزامات خاص کشور متفاوت باشد. اگر به اعتبارسنجی کد پستی قوی‌تری نیاز دارید، توصیه می‌شود از یک کتابخانه تخصصی استفاده کنید یا با مشخصات قالب کد پستی برای کشور خاصی که هدف آن هستید مشورت کنید.

### ه

نتایج:



توضیح pattern:

* ^ : شروع خط.
* ^\+?\d{1,3}\s? : کد کشور که شامل یک علامت مثبت اختیاری (+?) و به دنبال آن یک فضای اختیاری (\s?) است. بقیه الگوها با یک شماره تلفن در قالب‌های مختلف، مانند "1234567890"، "123 456 7890"، "(123)4567890"، "(123) 456 7890" و غیره مطابقت دارد.

توضیح کد:

کد ضمیمه شده یک اسکریپت پایتون است که تابع «تست\_فون(تلفن)» را برای اعتبارسنجی شماره تلفن در قالب‌های مختلف با استفاده از عبارات منظم (regex) تعریف می‌کند. در اینجا توضیحی درباره کاری که کد انجام می دهد آورده شده است:

1. اسکریپت ماژول «re» را وارد می کند که از عبارات منظم در پایتون پشتیبانی می کند.

2. تابع «تست\_تلفن(تلفن)» یک شماره تلفن را به عنوان ورودی می گیرد و اگر شماره تلفن معتبر باشد، «درست» و در غیر این صورت «نادرست» را برمی گرداند.

3. تابع یک الگوی عبارت منظم را برای مطابقت با شماره تلفن تعریف می کند. الگو در متغیر "الگو" ذخیره می شود.

4. الگوی عبارت منظم به شرح زیر است:

````

^\+?\d{1,3}\s?((?:[0-9]{3})|(?:\([0-9]{3}\)))[-\s] ?(\d{3})[-\s]?(\d{4})$

```

- `^` شروع یک خط را بیان می کند.

- `^\+?\d{1,3}\s?` با کد کشوری مطابقت دارد که شامل یک علامت مثبت اختیاری (+?) و به دنبال آن یک فضای اختیاری (\s?) است. این بخش از الگو، کدها و فرمت های مختلف کشور را امکان پذیر می کند.

- `((?:[0-9]{3})|(?:\([0-9]{3}\)))` یک گروه عکسبرداری است که با سه رقم ([0-9]{) مطابقت دارد 3}) یا سه رقم محصور در پرانتز (\([0-9]{3}\)). این قسمت از الگو با کد منطقه مطابقت دارد.

- `[-\s]؟` یک گروه اختیاری است که شامل خط فاصله (-) یا فاصله (\s) است.

- «(\d{3})» دقیقاً با سه رقم مطابقت دارد. این قسمت از الگو با سه رقم اول شماره تلفن مطابقت دارد.

- `[-\s]؟` یک گروه اختیاری است که شامل خط فاصله (-) یا فاصله (\s) است.

- «(\d{4})» دقیقاً با چهار رقم مطابقت دارد. این قسمت از الگو با چهار رقم آخر شماره تلفن مطابقت دارد.

- `$` پایان یک خط را بیان می کند.

5. این تابع از «re.match(الگو، تلفن)» استفاده می کند تا بررسی کند که «تلفن» ارائه شده با الگوی تعریف شده مطابقت دارد یا خیر. اگر مطابقت وجود داشته باشد، «درست» را برمی‌گرداند. در غیر این صورت، "False" را برمی گرداند.

6. سپس اسکریپت تابع «test\_phone()» را با شماره تلفن های مختلف آزمایش می کند.

- چهار تست اول ("Test 1"، "Test 2"، "Test 3" و "Test 4") شماره تلفن های معتبر در قالب های مختلف هستند و انتظار می رود که عملکرد برای همه آنها "True" را برگرداند.

- سه آزمایش آخر ("تست 5"، "تست 6" و "آزمون 7") شماره تلفن های نامعتبر هستند و انتظار می رود این تابع برای همه آنها "نادرست" را برگرداند.

7. اسکریپت توضیحات تست و نتایج مربوطه را چاپ می کند.

توجه داشته باشید که عبارت معمولی استفاده شده در این قطعه کد یک نسخه ساده شده است و ممکن است تمام قالب‌های شماره تلفن معتبر ممکن برای کشورهای مختلف را پوشش ندهد. اعتبار سنجی شماره تلفن می تواند پیچیده باشد، با در نظر گرفتن تغییرات در کدهای کشور، کد منطقه، و فرمت شماره های محلی. اگر به اعتبارسنجی شماره تلفن قوی‌تری نیاز دارید، توصیه می‌شود از یک کتابخانه تخصصی استفاده کنید یا با مشخصات قالب شماره تلفن برای کشورهای خاصی که هدف آن هستید مشورت کنید.

پایان