بسم الله الرحمن الرحیم

راهنمای Developer ماژول parser

تیر ۹۷

فهرست مطالب

فصل ۱

۱-۱نگارش ها

در جدول زیر تاریخچه نگارش های مختلف APIآورده شده است.

تاریخ	نسخه	توضیحات
اسفند ۹۶	۱,۰	نسخه اولیه document آماده شد.
فروردین ۹۷	۱,۱	• تغییر property مربوط به خروجی ها • اضافه شدن توضیحات مربوط به property ها
تیر ۹۷	۱,۰	• خروجی به روز رسانی شد. • Progressbar به سیستم اضافه شد. • موارد optional به ورودی اضافه شدند.

۱-۲ توضیحات اولیه

این ماژول جهت استخراج المان های مختلف از یک متن یا یک فایل نوشته شده است.در حال حاضر استخراج ۴ المان به نام های ip, phone, email، domain، account و url توسط این ماژول صورت می گیرد و همچنین ورودی این ماژول یا یک متن از جنس string هست و یا pathکه مسیر ذخیره یک فایل را در فریم وورک مشخص می کند. ماژول parser با توجه به path این فایل را باز کرده و المان های لازم را از آن استخراج می کند. لازم به یادآوری است که این ماژول صحت سنجی یا validation را هم بر روی المان های استخراجی انجام می دهد.یعنی المان های valid را به عنوان خروجی بر می گرداند. این ماژول در حال حاضر برای استخراج email و ip و اکانت از regular expression و برای استخراج url و phone و domain_name از کتابخانه های مشخص استفاده می کند.در این ماژول از دو کلاس اصلی استفاده می شود و هر کلاس متد های مشخص خود را دارند که در قسمت های بعدی مفصل در مورد آنها صحبت خواهیم کرد. این ماژول دارای دو متد می باشد که شماره متد انتخابی به عنوان ورودی باید پاس داده شود، متد ۱ از روی رشته ورودی را می خواند و باید رشته مورد نظر درون درخواست با کلید content ارسال گردد. متد ۲ از روی فایل می خواند و باید آدرس فایل با کلید path ارسال گردد.

ورودی	توضیحات
method_id	شناسه متد. در صورتی که ورودی یک رشته باشد این سناسه برابر ۱ است.
content	رشته ای که باید پارس شود.

جدول مربوط به ورودی ماژول parse از طریق رشته

ورودی	توضیحات
method_id	شناسه متد. در صورتی که ورودی مسیر یک فایل باشد این شناسه برابر ۲ است.
path	مسیر فایلی که باید پارس شود.

جدول مربوط به ورودی ماژول parse از طریق فایل

کلید	مقدار	نوضیحات
On_demand	[1,2,3,4,5,12]	ارسال این کلید در ورودی به شکل رو به رو، باعث می شود فقط type_entity های خواسته شده در لیست on_demand پارس شوند و در خروجی نمایش داده شوند. در صورتی که این مقدار ارسال نشود یا با فرمت اشتباه داده شود تمامی موجودیت های ممکن در خروجی ظاهر می گردند.
region	IR	با توجه به این که برای شناسایی شماره تلفن ها از ماژول طراحی شده توسط گوگل استفاده شده است، در خروجی فقط شماره هایی می آیند که همراه با پیش شماره کشور باشند. در صورت ارسال این مقدار به شماره هایی که دارای پیش شماره کشور نیستند، پیش شماره متناظر با region ارسالی به آن ها اضافه می شود و در صورتی که پس از اضافه شدن این پیش شماره valid شوند، در خروجی به همراه پیش شماره اضافه شده توسط مقدار ارسالی ظاهر می شوند. در صورتی که این مقدار ارسال نشود مقدار region برابرnone قرار می گیرد و تمامی شماره ها به پیش شماره نیاز خواهند داشت

جدول ورودی‌های optional

۱-۳ کلاس ParseElements

این کلاس جهت استخراج المان های گفته شده در بخش توضیحات به کار می رود. دارای ۴ متد است که هر متد برای استخراج یکی از المان ها به کار می رود. توضیحات لازم راجع به کار هر ماژول و همچنین ورودی و خروجی ها، در بخش های بعدی آورده شده است.

۱-۳-۱ متد parse_email()

این متد جهت استخراج ایمیل به کار می رود و محتوای دریافت شده را به عنوان ورودی دریافت کرده و لیستی از ایمیل های valid را به عنوان خروجی بر می گرداند. این متد برای استخراج ایمیل از regular expression مشخص استفاده کرده و همچنین برای validation ایمیل ها نیز از یک کتابخانه مشخص استفاده می نماید.

```
['hamid@radcom.ir', 'mohsen.baghdadi@gmail.com']
```

نمونه خروجی متد parse_email

۱-۳-۲ متد parse_phone()

این متد جهت استخراج تلفن به کار می رود و محتوای دریافت شده را به عنوان ورودی دریافت کرده و لیستی از تلفن های valid را به عنوان خروجی بر می گرداند. این متد برای استخراج تلفن از همان کتابخانه ای استفاده می کند که گوگل جهت استخراج و validation شماره های بین المللی استفاده می نماید. نمونه خروجی این متد در تصویر زیر قابل مشاهده است.

```
['+982166634721', '+982149261000', '+982189788621', '+982577683956',
'+989152682136', '+989122682136', '+989303122396', '+989012682136', '+989223122396',
'+982537740971', '+982537255890', '+983867543298', '+988367894532', '+989121234567']
```

نمونه خروجی متد parse_phone

۱-۳-۳ متد parse_ip()

این متد جهت استخراج ip به کار می رود و محتوای دریافت شده را به عنوان ورودی دریافت کرده و لیستی از ipها را به عنوان خروجی بر می گرداند. این متد برای استخراج ip از regular expression مشخص استفاده می نماید. نمونه خروجی این متد در تصویر زیر قابل مشاهده است.

```
['94.182.146.0', '39.166.95.9', '178.189.92.118', '198.2.202.33', '171.96.152.89',
'153.149.104.76', '106.187.52.191', '194.187.214.204', '59.78.160.247', '61.156.3.166']
```

نمونه خروجی متد parse_ip

۱-۳-۴ متد parse_url()

این متد جهت استخراج url به کار می رود و محتوای دریافت شده را به عنوان ورودی دریافت کرده و لیستی از urlها را به عنوان خروجی بر می گرداند. این متد برای استخراج url از کتابخانه مشخص استفاده می نماید. نمونه خروجی این متد در تصویر زیر قابل مشاهده است.

```
['http://jsonviewer.stack.hu', 'radcom.ir', 'gmail.com', 'http://example.com',
'https://stackoverflow.com/questions/6883049/regex-to-find-urls-in-string-in-python']
```

نمونه خروجی متد parse_url

نکته: urlهایی که با regex مربوط به اکانت های facebook، linkedin، twitter، Instagram match شوند، با type ۵ در خروجی ظاهر می شوند.

۱-۴ کلاس FetchResult

این کلاس جهت آماده کردن خروجی متدهای بخش قبلی به شکل entity_property نوشته شده است. دارای ۴ متد است که هر متد خروجی هر یک از ۴ متد بخش قبل را به شکل استاندارد entity_property در می اورد و خروجی مناسب را که شامل المان به همراه property آن است برمی گرداند. در بخش بعدی کارکرد هر یک از این ۴ متد و همچنین ورودی و خروجی انها به طور کامل توضیح داده می شود.

۱-۴-۱ متد prepare_email_result()

این متد یک آرگومان emailرا به عنوان تنها ورودی خود می گیرد و خروجی را به شکل entity_propertyو در قالب jsonدر می اورد. پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

ورودی	توضیحات
email	ایمیل استخراج شده از متد های بخش قبلی

جدول مربوط به ورودی متد prepare_email_result()

method	Output_parameters

prepare_email_result	{
	"data": "info@domain.ir",
	"type": 2,
	"properties": [
	{
	"local_address": "info",
	"type": 5
	},
	{
	"domain_name": "domain.ir",
	"type": 12
	},
	{
	"organization": "",
	"type": 11
	},
	{
	"owner": "",
	"type": 11
	},
	{
	"is_valid": "",
	"type": 0
	}
	]
	}

جدول مربوط به پارامترهای خروجی متد prepare_email_result()

توجه !

دقت شود که در جدول بالا پارامتر type مربوط به entity خروجی است که چون email است ، idآن برابر ۲ است. لیست entity ها در این لینک موجود است.

توجه !

توضیحات و جزییات مربوط به پارامتر properties در جدول بالا در این لینک موجود است.

۱-۴-۲ متد prepare_phone_result()

این متد یک آرگومان phoneرا به عنوان تنها ورودی خود می گیرد و خروجی را به شکل entity_propertyو در قالب jsonدر می اورد.در حال حاضر از property این بخش استفاده نمی کنیم.ولی برنامه ریزی شده که در اینده از انها نیز استفاده نماییم. پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

ورودی	توضیحات
phone	تلفن استخراج شده از متد های بخش قبلی

جدول مربوط به ورودی متد prepare_phone_result()

method	Output_parameters
prepare_phone_result	{ "data": "+98911111111", "type": 4, "properties": [{ "operator": "", "type": 11 }, { "location": "", "type": 11 }, { "country_code": "", "type": 0 }, { "phone_type": "", "type": 0 }] }

جدول مربوط به پارامترهای خروجی متد prepare_phone_result()

توجه !

دقت شود که در جدول بالا پارامتر type مربوط به entity خروجی است که چون phone است ، idآن برابر 4 است.

لیست entity ها در این لینک موجود است.

توجه !

توضیحات و جزییات مربوط به پارامتر properties در جدول بالا در این لینک موجود است.

۱-۴-۳ متد prepare_ip_result()

این متد یک آرگومان ip را به عنوان تنها ورودی خود می گیرد و خروجی را به شکل entity_property و در قالب json در می اورد.در حال حاضر از property این بخش استفاده نمی کنیم.ولی برنامه ریزی شده که در اینده از انها نیز استفاده نماییم. ورودی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

ورودی	توضیحات
ip	ip استخراج شده از متد های بخش قبلی

جدول مربوط به ورودی متد prepare_ip_result()

method	Output_parameters

prepare_ip_result	{
	"data": "192.168.1.1",
	"type": 3,
	"properties": [
	{
	"country": "",
	"type": 11
	},
	{
	"state": "",
	"type": 11
	}
	],
	"special_properties": [
	{
	"is_site_local": true,
	"type": 0
	},
	{
	"is_link_local": true,
	"type": 0
	},
	{
	"is_reserved": true,
	"type": 0
	},
	{
	"is_private": true,
	"type": 0
	},
	{
	"is_global": true,
	"type": 0
	},
	{
	"is_multicast": true,
	"type": 0
	},
	{
	"is_loopback": true,

	"type": 0
	},
	{
	"is_unspecified": true,
	"type": 0
	},
	{
	"version_type": "",
	"type": 0
	}
	]
	}

جدول مربوط به پارامترهای خروجی متد prepare_ip_result()

توجه !

دقت شود که در جدول بالا پارامتر type مربوط به entity خروجی است که چون ip است ، idآن برابر 3 است.لیست entityها در این لینک موجود است.

توجه !

توضیحات و جزییات مربوط به پارامتر properties در جدول بالا در این لینک موجود است.

۱-۴-۴ متد prepare_url_result()

این متد یک آرگومان urlرا به عنوان تنها ورودی خود می گیرد و خروجی را به شکل entity_property و در قالب json در می اورد. پارامترهای ورودی و خروجی این متد در زیر قابل مشاهده است. و همچنین نمونه خروجی متد در این لینک قابل مشاهده است.

ورودی	توضیحات
url	url استخراج شده از متد های بخش قبلی

جدول مربوط به ورودی متد prepare_url_result()

method	Output_parameters
prepare_url_result	{ "data": "https://domain.com/page1.html", "type": 1, "properties": [{ "domain_name": "domain.com", "type": 12 }, { "tld": "com", "type": 11 }], "special_properties": [{ "query": "", "type": 0 }, { "fragment": "", "type": 0 }, { "scheme": "", "type": 0 }, { "path": "page1.html", "type": 0 }] }

جدول مربوط به پارامترهای خروجی متد prepare_url_result()

توجه !

دقت شود که در جدول بالا پارامتر type مربوط به entity خروجی است که چون url است ، idآن برابر 1 است.لیست entityها در این لینک موجود است.

توجه !

توضیحات و جزییات مربوط به پارامتر properties در جدول بالا در این لینک موجود است.

۱-۴-۵ متد prepare_domain_name_result()

این متد یک آرگومان urlرا به عنوان تنها ورودی خود می گیرد و خروجی را به شکل entity_property و در قالب json در می اورد.
پارامترهای ورودی و خروجی این متد در زیر قابل مشاهده است.

method	Output_parameters
prepare_domain_name_result	{ "data": "domain.com", "type": 12, "properties": [{ "name": "domain", "type": 11 }, { "tld": "ir", "type": 11 }, { "subdomain": "", "type": 11 }] }

جدول مربوط به پارامترهای خروجی متد prepare_domain_name_result

توجه !

دقت شود که در جدول بالا پارامتر type مربوط به entity خروجی است که چون domain_name است، id آن برابر ۱۲ می باشد.لیست entity ها در این لینک موجود است.

توجه !

توضیحات و جزییات مربوط به پارامتر properties در جدول بالا در این لینک موجود است.

۱-۴-۶- خروجی نهایی

method	Output_parameters
result	{ "results": [{ "data": "https://domain.com/page1.html", "type": 1, "properties": [{ "domain_name": "domain.com", "type": 12 }, { "tld": "com", "type": 11 }], "special_properties": [{ "query": "", "type": 0 }, { "fragment": "", "type": 0 }, { "scheme": "", "type": 0 }, { "path": "page1.html", "type": 0 }] }, { "data": "domain.com", "type": 12, "properties": [{ "name": "domain", "type": 11 }, { "tld": "ir", "type": 11 }, { "subdomain": "", "type": 11 }] }, { "data": "+98911111111", "type": 4, "properties": [{

```
      "operator": "",
      "type": 11
     },
     {
      "location": "",
      "type": 11
     },
     {
      "country_code": "",
      "type": 0
     },
     {
      "phone_type": "",
      "type": 0
     }
    ]
   },
   {
    "data": "192.168.1.1",
    "type": 3,
    "properties": [
     {
      "country": "",
      "type": 11
     },
     {
      "state": "",
      "type": 11
     }
    ],
    "special_properties": [
     {
      "is_site_local": true,
      "type": 0
     },
     {
      "is_link_local": true,
      "type": 0
     },
     {
      "is_reserved": true,
      "type": 0
     },
     {
      "is_private": true,
      "type": 0
     },
     {
      "is_global": true,
      "type": 0
     },
     {
      "is_multicast": true,
      "type": 0
     },
     {
      "is_loopback": true,
      "type": 0
     },
     {
      "is_unspecified": true,
```

```
        "type": 0
       },
       {
        "version_type": "",
        "type": 0
       }
      ]
     },
     {
      "data": "info@domain.ir",
      "type": 2,
      "properties": [
       {
        "local_address": "info",
        "type": 5
       },
       {
        "domain_name": "domain.ir",
        "type": 12
       },
       {
        "organization": "",
        "type": 11
       },
       {
        "owner": "",
        "type": 11
       },
       {
        "is_valid": "",
        "type": 0
       }
      ]
     }
    ]
}
```


خروجی نهایی ماژول parser

خروجی این ماژول در این لینک موجود است.

۵-۱ محتوای موجود در socket request

برای استفاده از ماژولparser باید یک درخواست از طریق socket به فریم وورک ارسال شود. بدنه درخواست ارسالی شامل پارامترهای data و type است. پارامتر type مشخص کننده نوع ورودی ماژول(unstructue,file,domain,path,…) است که id مربوط به انواع مختلف ورودی ها و توضیحات مربوطه که در این لینک موجود است، در این فیلد قرار می گیرد. پارامتر data هم همان داده ارسالی است که شامل پارامترهای ورودی ماژول parser است که یا یک content از نوع string است و یا یک path که نشان دهنده آدرس فایل html در فریم وورک می باشد.

```
{

  "data": {

        "content":"the email address of our websit is hamid@radcom.ir",
        "method" : "1"

       },

  "type": 1
}
```


نمونه body یک socket request با type 1


```
{

  "data": {

        "path":"/storage/crawl/6C1A8503/08720264-cc26-4464-a626.html",

        "method" : "2"

       },

  "type": 9
```


نمونه body یک socket request با type 9

در تصاویر نمونه بالا داریم:

data : داده ارسالی برای استفاده در ماژول که شامل پارامترهای ورودی ماژول هست.که یا یک content بوده و یا یک path

type : نوع ورودی ماژول را مشخص می کند که می تواند از نوع unstructured,path,file و یا انواع دیگر باشد و باید idورودی موردنظر در این قسمت قرار گیرد.

content: محتوای مورد نظر که خواستار استخراج المان ها از ان هستیم.

path : مسیر ذخیره فایل html در سیستم را مشخص می کند.

۱-۶ جدول خطاها

جدول زیر شامل خطاهایی است که ماژول parser در پاسخ به کلاینت بر می گرداند. خطا ها با شماره مشخصی نشان داده می شوند.

کد	نام	توضیحات
۱۰۲	InvalidInputError	پارامتر ورودی ارسال نشده است.

۱-۷- آپدیت progressbar

برای نمایش progressbar به صورت real time. محل به روز رسانی progressbar درون ماژول است. بدین منظور هر قسمت از ماژول که قسمتی از نتیجه را محاسبه می کند. قبل یا بعد از آن باید update_progressbar صدا زده شود.

در ماژول parser به علت سبک بودن ماژول، progressbar تنها دو بار صدا زده می شود. یکبار پس از استخراج مومودیت ها و بار دو پس از parse کردن موجودیت ها.