

بسم الله الرحمن الرحيم

راهنمای Developer twitter

تیر ۹۷

| | |
|---|----|
| ۱-۱ نگارش ها..... | ۲ |
| ۲-۱ توضیحات اولیه..... | ۲ |
| ۳-۱ جستجو در twitter..... | ۲ |
| ۱-۳-۱ متد search()..... | ۲ |
| ۴-۱ متد show_user()..... | ۶ |
| ۵-۱ متد get_friends()..... | ۱۰ |
| ۴-۱ محتوای موجود در socket request..... | ۱۳ |
| ۸-۱ جدول خطاها..... | ۱۸ |
| ۹-۱ آبدیت progressbar..... | ۱۸ |

فصل ۱

۱-۱ نگارش ها

در جدول زیر تاریخچه نگارش های مختلف API آورده شده است.

| تاریخ | نسخه | توضیحات |
|--------|------|-------------------------------|
| تیر ۹۷ | ۱،۰ | نسخه اولیه document آماده شد. |

۲-۱ توضیحات اولیه

این ماژول جهت جستجو یک عبارت یا ترم، پیدا کردن لیست دوستان، پیدا کردن لیست دنبال کننده ها، نمایش اطلاعات یک پروفایل در twitter با استفاده از api نوشته شده است. کلاس مربوط به این ماژول تنها ۴ متد دارد که نحوه کار این متدها در قسمت بعدی شرح داده می شود. کد مربوط به این ماژول ۱۱۴ می باشد. [جدول ماژول ها](#).

۳-۱ جستجو در twitter

همانطور که در بخش توضیحات گفته شد، کلاس مربوط به ماژول twitter یک متد به نام search() دارد. این متد عبارت جستجو و همچنین تعداد نتایج مورد درخواست که بر اساس تعداد صفحه است را به عنوان پارامتر ورودی دریافت می کند و نتیجه جستجو را بر می گرداند. در قسمت بعدی در مورد این متد توضیحات بیشتری ارائه خواهد شد.

۱-۳-۱ متد search()

این متد دو آرگومان به نام query که عبارت جستجو است و همچنین max_result که تعداد نتایج مورد نظر (بر حسب تعداد صفحه است) را به عنوان ورودی می گیرد و خروجی که شامل کلیه اطلاعات مربوط به تعداد صفحات مورد درخواست نتیجه جستجو است را در قالب یک لیست برمی گرداند. ارسال max_result اختیاری است و در صورت نفرستادن مقدار آن برابر ۴ قرار می گیرد. همچنین در صورتی که بیشتر از ۲۰۰ نتیجه خواسته شود، تعداد ۲۰۰ رکورد برگردانده می شود چون api مربوط به توییتر برای هر درخواست تا ۲۰۰ رکورد پاسخ می دهد. جداول زیر پارامترهای ورودی و خروجی را نشان می دهد. همچنین یک نمونه خروجی در این [لینک](#) موجود است. شناسه این متد برابر ۱ است.

| ورودی | توضیحات |
|-----------------|--|
| method_id | شناسه تابع، برای این تابع برابر ۱ است |
| query | عبارت جستجو را نشان میدهد. مانند: "fringe" |
| consumer_key | کلید مورد نیاز برای درخواست به توییت |
| consumer_secret | کلید مورد نیاز برای درخواست به توییت |
| access_key | کلید مورد نیاز برای درخواست به توییت |
| access_secret | کلید مورد نیاز برای درخواست به توییت |
| max_result | تعداد نتایج درخواستی که بر حسب تعداد صفحه است. |

جدول مربوط به ورودی متد search

توجه!

دقت شود که آرگومان max_result به عنوان پارامتر ورودی، اختیاری است و در صورت تنظیم نکردن این آرگومان، مقدار پیش فرض 4 رکورد برای آن در نظر گرفته می شود و بیشینه آن ۲۰۰ می باشد.

| method | Output_parameters |
|--------|--|
| Search | <pre>{ "has_mdia": true, "results": [{ "special_properties": [{ "sub_type": 2, "type": 0 }, { "screen_name": "", "type": 5 }, { "location": "", "type": 8 }, { "protected":, "type": 0 }, { "verified":, "type": 0 }] }] }</pre> |

```

{
  "followers_count":,
  "type": 0
},
{
  "friends_count":,
  "type": 0
},
{
  "listed_count":,
  "type": 0
},
{
  "favourites_count":,
  "type": 0
},
{
  "statuses_count":,
  "type": 0
},
{
  "type": 0,
  "created_at": ""
},
{
  "type": 0,
  "geo_enabled": ""
},
{
  "type": 11,
  "lang": ""
},
{
  "contributors_enabled": "",
  "type": 0
},
{
  "more": {
    "url": "",
    "ref": "",
    "file_name": ""
  },
  "type": 13,
  "profile_background_image_url": "
},
{
  "profile_background_image_url_thumbnail": "
",
  "type": 13,
  "more": {
    "url": "",
    "ref": "",
    "file_name": ""
  }
},
{
  "profile_banner_url": "
",
  "type": 13,
  "more": {
    "url": "",
    "ref": "",
    "file_name": ""
  }
}

```

```

    }
  },
  {
    "profile_banner_url_thumbnail": "",
    "type": 13,
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    }
  },
  {
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    },
    "type": 13,
    "profile_image_url": ""
  },
  {
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    },
    "type": 13,
    "profile_image_url_thumbnail": ""
  }
],
"data": "",
"type": 5,
"properties": [
  {
    "name": "",
    "type": 11
  },
  {
    "description": "",
    "type": 0
  },
  {
    "url": [
      {
        "title": ""
      },
      {
        "title": ""
      }
    ],
    "type": 1
  }
]
}
]
}

```

جدول مربوط به پارامترهای خروجی متد search

توجه!

دقت شود که در جدول بالا پارامتر type مربوط به entity درون همان بلاک است. لیست entity ها در این [لینک](#) موجود است.

نمونه خروجی این متد در [لینک](#) زیر موجود است.

توجه!

در این متد، عکس پروفایل توسط ماژول دانلود می شود و لینکی که در جواب به کاربر داده می شود، مربوط به فایل دانلود شده توسط سرویس دهنده است. البته لازم به ذکر است لینک اصلی عکس در قسمت more-url آورده می شود.

توجه!

رکورد has_media به این معنی است که این ماژول یک media را در مسیر ذخیره سازی system api ذخیره می کند.

۱-۴- show_user

این متد دو آرگومان به نام user_id و max_result دریافت می کند و مطابق جدول زیر اطلاعات مربوط به پروفایل متناظر با user_id داده شده را در خروجی می دهد.

| ورودی | توضیحات |
|-----------------|---|
| method_id | شناسه تابع، برای این تابع برابر ۲ است |
| user_id | آی دی یا screen_name پروفایل : "۱۴۵۵۵۸۸۹" |
| consumer_key | کلید مورد نیاز برای درخواست به توییت |
| consumer_secret | کلید مورد نیاز برای درخواست به توییت |
| access_key | کلید مورد نیاز برای درخواست به توییت |
| access_secret | کلید مورد نیاز برای درخواست به توییت |

جدول مربوط به ورودی متد show_user

| method | Output_parameters |
|-----------|--|
| Show_user | <pre> { "has_media": true, "special_properties": [{ "sub_type": 2, "type": 0 }, { "screen_name": "", "type": 5 }, { "location": "", "type": 8 }, { "protected": "", "type": 0 }, { "verified": "", "type": 0 }, { "followers_count": "", "type": 0 }, { "friends_count": "", "type": 0 }, { "listed_count": "", "type": 0 }, { "favourites_count": "", "type": 0 }, { "statuses_count": "", "type": 0 }, { "type": 0, "created_at": "" }, { "type": 0, "geo_enabled": "" }, { "type": 11, "lang": "" }, { "contributors_enabled": "", "type": 0 }, { </pre> |

```

    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    },
    "type": 13,
    "profile_background_image_url": "
  },
  {
    "profile_background_image_url_thumbnail": "",
    "type": 13,
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    }
  },
  {
    "profile_banner_url": "",
    "type": 13,
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    }
  },
  {
    "profile_banner_url_thumbnail": "",
    "type": 13,
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    }
  },
  {
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    },
    "type": 13,
    "profile_image_url": "
  },
  {
    "more": {
      "url": "",
      "ref": "",
      "file_name": ""
    },
    "type": 13,
    "profile_image_url_thumbnail": "
  }
],
"results": [
  {
    "data": "username",
    "type": 5,
    "ref": {
      "section": "screen_name",
      "task": "twitter_profile",

```



```

        "twitter_account": "football"
    }
},
{
    "data": "location",
    "type": 8,
    "ref": {
        "section": "location",
        "task": "twitter_profile",
        "twitter_account": "football"
    }
},
{
    "data": "name",
    "type": 11,
    "ref": {
        "section": "name",
        "task": "twitter_profile",
        "twitter_account": "football"
    }
}
],
"properties": [
    {
        "name": "",
        "type": 11
    },
    {
        "description": "",
        "type": 0
    },
    {
        "url": [
            {
                "title": ""
            },
            {
                "title": null
            }
        ],
        "type": 1
    }
]
}

```

جدول مربوط به پارامترهای خروجی متد show_user

توجه!

دقت شود که در جدول بالا پارامتر type مربوط به entity درون همان بلاک است. لیست entity ها در این [لینک](#) موجود است.

نمونه خروجی این متد در [لینک](#) زیر موجود است.

توجه!

در این متد، عکس پروفایل توسط مازول دانلود می شود و لینکی که در جواب به کاربر داده می شود، مربوط به فایل دانلود شده توسط سرویس دهنده است. البته لازم به ذکر است لینک اصلی عکس در قسمت more-url آورده می شود.

توجه!

رکورد has_media به این معنی است که این مازول یک media را در مسیر ذخیره سازی system api ذخیره می کند.

۵-۱- متد get_friends()

این متد دو آرگومان به نام user_id و max_result دریافت می کند و مطابق جدول زیر اطلاعات مربوط به دوستان پروفایل متناظر با user_id داده شده را در خروجی می دهد. پارامتر max_result تعداد نتایج مورد درخواست را بر حسب تعداد نفر نشان می دهد. ارسال max_result اختیاری است و در صورت نفرستادن مقدار آن برابر ۴ قرار می گیرد. همچنین در صورتی که بیشتر از ۲۰۰ نتیجه خواسته شود، تعداد ۲۰۰ رکورد برگردانده می شود چون api مربوط به توئیتر برای هر درخواست تا ۲۰۰ رکورد پاسخ می دهد. جداول زیر پارامترهای ورودی و خروجی را نشان می دهد.

| ورودی | توضیحات |
|-----------------|--|
| method_id | شناسه تابع، برای این تابع برابر ۳ است |
| user_id | آی دی یا screen_name پروفایل : "۱۴۵۵۵۸۸۹" |
| consumer_key | کلید مورد نیاز برای درخواست به توئیتر |
| consumer_secret | کلید مورد نیاز برای درخواست به توئیتر |
| access_key | کلید مورد نیاز برای درخواست به توئیتر |
| access_secret | کلید مورد نیاز برای درخواست به توئیتر |
| max_result | تعداد نتایج درخواستی که بر حسب تعداد صفحه است. |

جدول مربوط به ورودی متد get_friends

توجه!

دقت شود که آرگومان max_result به عنوان پارامتر ورودی، اختیاری است و در صورت تنظیم نکردن این آرگومان، مقدار پیش فرض 4 رکورد برای آن در نظر گرفته می شود و بیشینه آن ۲۰۰ است.

| method | Output_parameters |
|-------------|---|
| Get_friends | <pre> { "has_media": true, "results": [{ "special_properties": [{ "sub_type": 2, "type": 0 }, { "screen_name": "", "type": 5 }, { "location": "", "type": 8 }, { "protected": "", "type": 0 }, { "verified": "", "type": 0 }, { "followers_count": "", "type": 0 }, { "friends_count": "", "type": 0 }, { "listed_count": "", "type": 0 }, { "favourites_count": "", "type": 0 }, { "statuses_count": "", "type": 0 }, { "type": 0, "created_at": "" }, { "type": 0, "geo_enabled": "" }, { "type": 11, "lang": "en-gb" }, { "contributors_enabled": false, "type": 0 }, { "more": { "url": "", "ref": "", "file_name": "" }, "type": 13, "profile_background_image_url": "" }, { "profile_background_image_url_thumbnail": "", "type": 13, "more": { "url": "", </pre> |

```

        "ref": "",
        "file_name": ""
    },
    {
        "profile_banner_url": "",
        "type": 13,
        "more": {
            "url": "",
            "ref": "",
            "file_name": ""
        }
    },
    {
        "profile_banner_url_thumbnail": "",
        "type": 13,
        "more": {
            "url": "",
            "ref": "",
            "file_name": ""
        }
    },
    {
        "more": {
            "url": "",
            "ref": "",
            "file_name": ""
        },
        "type": 13,
        "profile_image_url": ""
    },
    {
        "more": {
            "url": "",
            "ref": "",
            "file_name": ""
        },
        "type": 13,
        "profile_image_url_thumbnail": ""
    }
],
"data": "",
"type": 5,
"properties": [
    {
        "name": "",
        "type": 11
    },
    {
        "type": 0
    },
    {
        "url": [
            {
                "title": ""
            },
            {
                "title": null
            }
        ],
        "type": 1
    }
]
}

```

جدول مربوط به پارامترهای خروجی متد `get_friends`

توجه!

دقت شود که در جدول بالا پارامتر type مربوط به entity درون همان بلاک است. لیست entity ها در این [لینک](#) موجود است.

نمونه خروجی این متد در [لینک](#) زیر موجود است.

توجه!

در این متد، عکس پروفایل توسط مازول دانلود می شود و لینکی که در جواب به کاربر داده می شود، مربوط به فایل دانلود شده توسط سرویس دهنده است. البته لازم به ذکر است لینک اصلی عکس در قسمت more-url آورده می شود.

توجه!

رکورد has_media به این معنی است که این مازول یک media را در مسیر ذخیره سازی system api ذخیره می کند.

۱-۶- get_follower

این متد دو آرگومان به نام user_id و max_result دریافت می کند و مطابق جدول زیر اطلاعات مربوط به followerهای پروفایل متناظر با user_id داده شده را در خروجی می دهد. پارامتر max_result تعداد نتایج مورد درخواست را بر حسب تعداد نفرنشان می دهد. ارسال max_result اختیاری است و در صورت نفرستادن مقدار آن برابر ۴ قرار می گیرد. همچنین در صورتی که بیشتر از ۲۰۰ نتیجه خواسته شود، تعداد ۲۰۰ رکورد برگردانده می شود چون api مربوط به توئیتر برای هر درخواست تا ۲۰۰ رکورد پاسخ می دهد. جداول زیر پارامترهای ورودی و خروجی را نشان می دهد.

| ورودی | توضیحات |
|-----------------|--|
| method_id | شناسه تابع، برای این تابع برابر ۴ است |
| user_id | آی دی یا screen_name پروفایل : "fifa" |
| consumer_key | کلید مورد نیاز برای درخواست به توییتر |
| consumer_secret | کلید مورد نیاز برای درخواست به توییتر |
| access_key | کلید مورد نیاز برای درخواست به توییتر |
| access_secret | کلید مورد نیاز برای درخواست به توییتر |
| max_result | تعداد نتایج درخواستی که بر حسب تعداد صفحه است. |

جدول مربوط به ورودی متد get_follower

توجه !

دقت شود که آرگومان max_result به عنوان پارامتر ورودی، اختیاری است و در صورت تنظیم نکردن این آرگومان، مقدار پیش فرض 4 رکورد برای آن در نظر گرفته می شود.

| method | Output_parameters |
|--------------|---|
| Get_follower | <pre>{ "has_media": true, "results": [{ "special_properties": [{ "sub_type": 2, "type": 0 }, { "screen_name": "", "type": 5 }, { "location": "", "type": 8 }, { "protected": "", "type": 0 }, { "verified": "", "type": 0 }, { "followers_count": "", "type": 0 }, { "friends_count": "", "type": 0 }] }] }</pre> |

```

    },
    {
      "listed_count": "",
      "type": 0
    },
    {
      "favourites_count": "",
      "type": 0
    },
    {
      "statuses_count": "",
      "type": 0
    },
    {
      "type": 0,
      "created_at": ""
    },
    {
      "type": 0,
      "geo_enabled": ""
    },
    {
      "type": 11,
      "lang": "en-gb"
    },
    {
      "contributors_enabled": false,
      "type": 0
    },
    {
      "more": {
        "url": "",
        "ref": "",
        "file_name": ""
      },
      "type": 13,
      "profile_background_image_url": ""
    },
    {
      "profile_background_image_url_thumbnail": "",
      "type": 13,
      "more": {
        "url": "",
        "ref": "",
        "file_name": ""
      }
    },
    {
      "profile_banner_url": "",
      "type": 13,
      "more": {
        "url": "",
        "ref": "",
        "file_name": ""
      }
    },
    {
      "profile_banner_url_thumbnail": "",
      "type": 13,
      "more": {
        "url": "",
        "ref": "",
        "file_name": ""
      }
    },
    {
      "more": {
        "url": "",
        "ref": "",
        "file_name": ""
      },
      "type": 13,
      "profile_image_url": ""
    },
    {
      "more": {
        "url": "",

```

```

      "ref": " ",
      "file_name": ""
    },
    "type": 13,
    "profile_image_url_thumbnail": ""
  }
],
"data": "",
"type": 5,
"properties": [
  {
    "name": " ",
    "type": 11
  },
  {
    "type": 0
  },
  {
    "url": [
      {
        "title": ""
      },
      {
        "title": null
      }
    ],
    "type": 1
  }
]
}
}
}

```

جدول مربوط به پارامترهای خروجی متد `get_followers`

توجه!

دقت شود که در جدول بالا پارامتر `type` مربوط به `entity` درون همان بلاک است. لیست `entity` ها در این [لینک](#) موجود است.

نمونه خروجی این متد در [لینک](#) زیر موجود است.

توجه!

در این متد، عکس پروفایل توسط ماژول دانلود می شود و لینکی که در جواب به کاربر داده می شود، مربوط به فایل دانلود شده توسط سرویس دهنده است. البته لازم به ذکر است لینک اصلی عکس در قسمت `more-url` آورده می شود.

توجه!

رکورد `has_media` به این معنی است که این ماژول یک `media` را در مسیر ذخیره سازی `system api` ذخیره می کند.

۷-۱- socket محتوای

برای استفاده از مازول twitter 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 مازول (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 متد search می باشد. توضیحات مربوط به متد search () [در قسمت های قبلی](#) ارائه شد.

```
{
  "data": {
    "consumer_key": "con_key",
    "consumer_secret": "con_sec",
    "access_key": "acc_key",
    "access_secret": "acc_sec",
    "query": "fringe",
    "max_result": "2",
    "1": "method_id",
    "type": "0"
  }
}
```

در تصویر نمونه بالا داریم:

data : داده ارسالی برای استفاده در مازول که شامل پارامترهای ورودی متد search() هست.

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

query : عبارت موردنظر برای جستجو که یکی از پارامترهای ورودی متد موردنظر ماست.

consumer_key : کلید مورد نیاز برای درخواست به توییتر

consumer_secret : کلید مورد نیاز برای درخواست به توییتر

access_key : کلید مورد نیاز برای درخواست به توییتر

access_secret : کلید مورد نیاز برای درخواست به توییتر

max_result : تعداد نتایج مورد درخواست را مشخص می کند که یکی از پارامترهای ورودی متد موردنظر ماست. این پارامتر اختیاری است.

method_id : شماره شناسه تابع مورد نظر

۸-۱- جدول خطاها

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

| کد | نام | توضیحات |
|-----|----------------------|---|
| ۱۰۲ | InvalidInputError | زمانی که فرمت داده ورودی اشتباه باشد. |
| ۱۱۱ | CancelExecutionError | زمانی رخ می دهد که کاربر درخواست کنسل داده است. |
| ۱۱۴ | ResultNotFoundError | زمانی که نتیجه ای برای درخواست یافت نشده است. |
| ۱۰۷ | NetworkError | مشکلی در دسترسی به شبکه وجود دارد. |
| ۱۱۶ | TwitterApiConstraint | زمانی رخ می دهد که تعداد درخواست ها از مقدار محدودیت زمانی توییتر بیشتر باشد. |

جدول مربوط به خطاهای ماژول

۹-۱- آبدیت 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 twitter سه بار این متد صدا زده می شود: ۱- پس از ثبت درخواست. ۲- پس از گرفتن نتایج از api. ۳- پس از پارس کردن جواب.