بسم الله الرحمن الرحیم

راهنمای developer ماژول IP2Location

مرداد 97

فصل 1

1-1 نگارش

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

|  |  |  |
| --- | --- | --- |
| تاریخ | نسخه | توضیحات |
| تیر 97 | 1.0 | نسخه اولیه document آماده شد. |

۱-۲ توضیحات اولیه

این ماژول جهت دریافت مشخصات یک IP از طریق ارسال یک درخواست get به api مورد نظر نوشته شده است. این ماژول دارای دو متد اصلی به نام های get\_ip\_info و ip\_is\_proxy می باشد. توضیحات بیشتر در مورد متود های این ماژول در زیر آورده شده است. کد مربوط به این ماژول 115 می باشد. [جدول ماژول ها.](refrence_links/modules_code.pdf)

۱-۳ متد get\_ip\_info()

این متد دو آرگومان ip و API\_KEY را به عنوان ورودی دریافت می کند و یک دیکشنری از مشخصات ip وارد شده را به عنوان خروجی برمی گرداند. پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 1 می باشد |
| ip | ip مورد نظر جهت دریافت مشخصات آن |
| Api\_key | جهت اتصال به سرویس مورد نظر نیاز به API\_KEY آن می باشد. |

جدول مربوط به ورودی متد get\_ip\_info ()

|  |  |  |
| --- | --- | --- |
| Output\_parameters | method | |
| {  "properties": [ {  "country\_name": "country\_name",  "type": 11  },  {  "city\_name": "city\_name",  "type": 11  },  {  "latitude": "latitude",  "type": 0  },  {  "longitude": "longitude",  "type": 0  }  ],  "special\_properties": [  {  "region\_name": "region\_name",  "type": 11  },  {  "zip\_code": "zip\_code",  "type": 0  },  {  "time\_zone": "time\_zone",  "type": 0  },  {  "isp": "isp",  "type": 11  },  {  "domain": "hetzner.de",  "type": 12  },  {  "idd\_code": "idd\_code",  "type": 0  },  {  "area\_code": "area\_code", "type": 0  },  {  "mobile\_brand": "mobile\_brand",  "type": 11  },  {  "elevation": "elevation",  "type": 0  },  {  "usage\_type\_desc": "usage\_type + Please Read Document",  "type": 0  },  {  "mcc": "mcc",  "type": 0  },  {  "mnc": "mnc",  "type": 0  },  {  "net\_speed": "net\_speed",  "type": 0  }  ],  "results": [ {  "geo\_coordinates": " latitude, longitude ",  "type": 14  }  ]  } | | get\_ip\_info() |

جدول مربوط به پارامترهای خروجی متد get\_ip\_info()

نمونه خروجی کامل این ماژول در این [لینک](refrence_links/output_sample%20getipinfo.txt) موجود است.

۱-4 متد ip\_is\_proxy()

این متد دو آرگومان ip و API\_KEY را به عنوان ورودی دریافت می کند و یک دیکشنری از مشخصات ip وارد شده را به عنوان خروجی برمی گرداند. . پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 2 می باشد |
| ip | ip مورد نظر جهت دریافت مشخصات آن |
| api\_key | جهت اتصال به سرویس مورد نظر نیاز به API\_KEY آن می باشد. |

جدول مربوط به ورودی متد ip\_is\_proxy()

|  |  |  |
| --- | --- | --- |
| Output\_parameters | | method |
| {  "properties": [],  "special\_properties": [  {  "country\_name": "countryName",  "type": 11  },  {  "region\_name": "regionName",  "type": 11  },  {  "city\_name": "cityName",  "type": 11  },  {  "isp": "isp",  "type": 11  },  {  "proxy\_type\_desc": "proxyType + Please Read Document",  "type": 0  },  {  "is\_proxy": "isProxy",  "type": 0  }  ],  "results": []  } | ip\_is\_proxy() | |

جدول مربوط به پارامترهای خروجی متد ip\_is\_proxy()

نمونه خروجی کامل این ماژول در این [لینک](refrence_links/output_sample%20ipisproxy.txt) موجود است.

۱-5 متد ip\_info\_balance()

این متد یک آرگومان API\_KEY را به عنوان ورودی دریافت می کند و میزان بالانس API مربوط به ip\_info را در خروجی می‌دهد. پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 3 می باشد |
| api\_key | جهت اتصال به سرویس مورد نظر نیاز به API\_KEY آن می باشد. |

جدول مربوط به ورودی متد ip\_info\_balance()

|  |  |  |
| --- | --- | --- |
| Output\_parameters | | method |
| {  "results": [{"ip\_info\_balance": 0, “type”: 0}]  } | ip\_is\_proxy() | |

جدول مربوط به پارامترهای خروجی متد ip\_info\_balance()

۱-6 متد ip\_is\_proxy\_balance()

این متد یک آرگومان API\_KEY را به عنوان ورودی دریافت می کند و میزان بالانس API مربوط به ip\_is\_proxy را در خروجی می‌دهد. پارامترهای ورودی و خروجی و همچنین نمونه خروجی این متد در زیر قابل مشاهده است.

|  |  |
| --- | --- |
| ورودی | توضیحات |
| Method\_id | شماره تابع. مقدار این پارامتر برای این متد برابر 4 می باشد |
| api\_key | جهت اتصال به سرویس مورد نظر نیاز به API\_KEY آن می باشد. |

جدول مربوط به ورودی متد ip\_is\_proxy\_balance()

|  |  |  |
| --- | --- | --- |
| Output\_parameters | | method |
| {  "results": [{"ip\_is\_proxy\_balance":”0”, “type”: 0}]  } | ip\_is\_proxy() | |

جدول مربوط به پارامترهای خروجی متد ip\_is\_proxy\_balance()

1-7 محتوای موجود در درخواست get

برای استفاده ازاین ماژول می بایست یک درخواست از نوع get جهت دریافت اطلاعات به api مورد نظر به صورت زیر 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](refrence_links/entities_list.pdf) موجود است، در این فیلد قرار می گیرد. پارامتر data هم همان داده ارسالی است که شامل پارامترهای ورودی ماژول می باشد.

**{**

**“data”:{“IP”:” 185.81.98.126”, “API\_KEY”:” C571380781”},**

**“type”:0**

**}**

نمونه body یک socket request

در تصویر نمونه بالا داریم:

Data: داده ارسالی برای استفاده در ماژول که شامل IP و API\_KEY می باشد.

Type: نوع ورودی ماژول 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.

1-8- جدول خطاها

جدول زیر شامل خطاهایی است که ماژول IP2Location در پاسخ به کلاینت بر می گرداند. خطا ها با شماره مشخصی نشان داده می شوند.

|  |  |  |
| --- | --- | --- |
| توضیحات | نام | کد |
| زمانی که فرمت داده ورودی اشتباه باشد. | InvalidInputError | 102 |
| زمانی که کلید API اشتباه باشد. | WrongApikeyError | 104 |
| زمانی که شبکه دچار تداخل شود. | NetworkError | 107 |
| زمانی که نتیجه ای برگشت داده نشود. | resultNotFoundError | 114 |