

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

آزمون API با ابزار POSTMAN



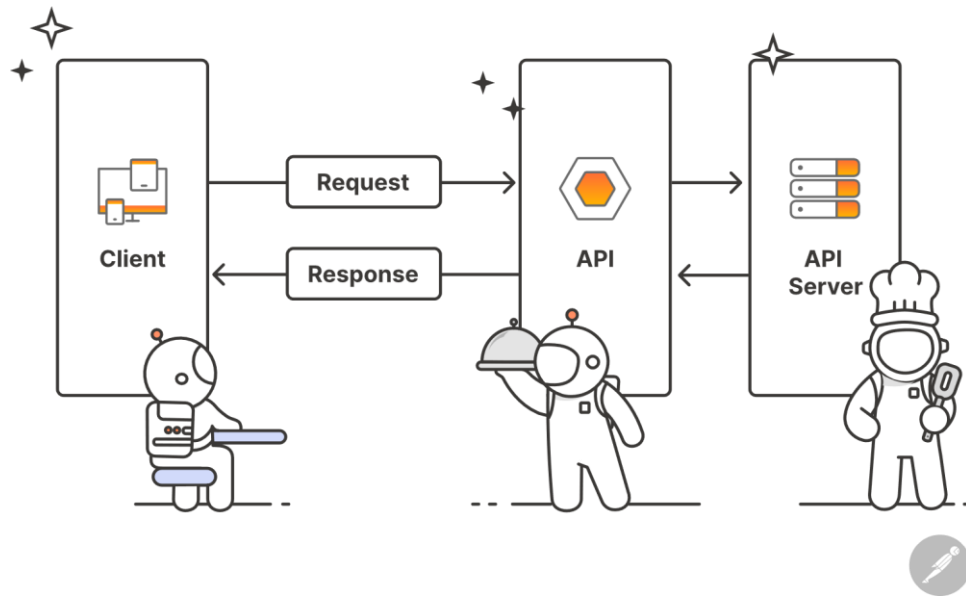
ارائه‌ی درس آزمون نرم‌افزار

ارائه توسط : یحیی پورسلطانی و سید مجتبی ابطحی

استاد : دکتر سید حسن میریان حسین آبادی

بهار ۱۴۰۳

API



- مخفف Application Interface است.

- کارکردهای لازم را به استفاده‌کنندگان عرضه می‌کند.

- بدون نیاز به ارائه‌ی کد اصلی.

- مزایا

- بالا رفتن امنیت به علت عدم دسترسی کاربران به کد مقصد.

- عدم ارسال اطلاعات اضافی.

- مثلاً برای دانستن اسامی دانشجویان، لزومی به داشتن کد ملی ایشان نیست.

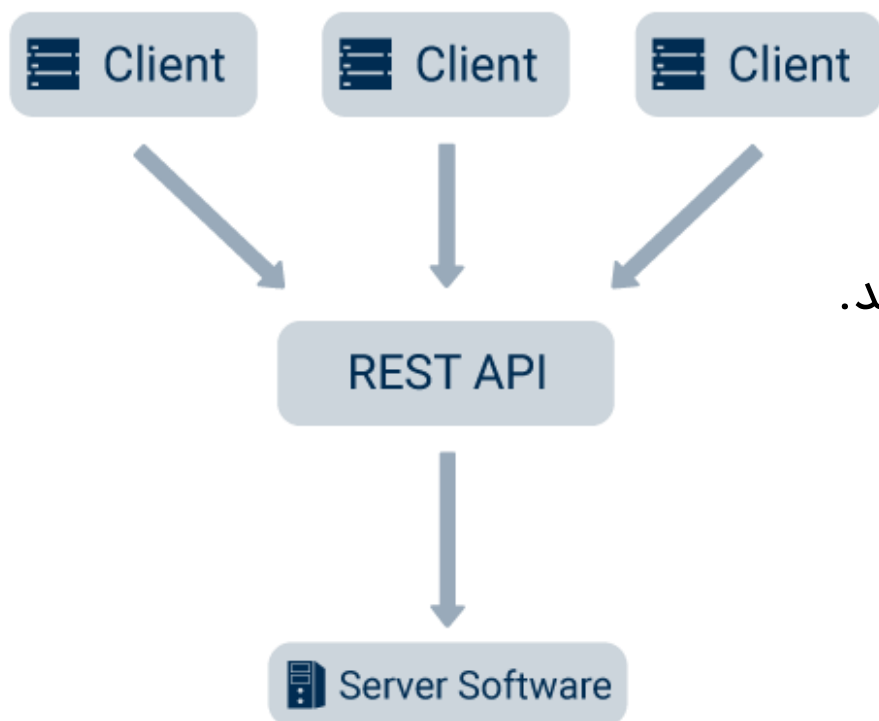
API REQUEST

- اجزای درخواست به یک API شامل موارد زیر است :
- پایانه (End Point) : آدرسی است که درخواست به آن ارسال می‌شود.
- فعل (Method) : روش پردازش درخواست (مثال : ایجاد یک رکورد)
- پارامترها
- سرآیند درخواست (Request Headers) : به صورت زوج کلید - مقدار و حاوی متا داده‌ی درخواست است.
- مثال : شناسه‌ی Session و یا رمز عبور
- بدنه‌ی درخواست : حاوی محتویات درخواست است.

API RESPONSE

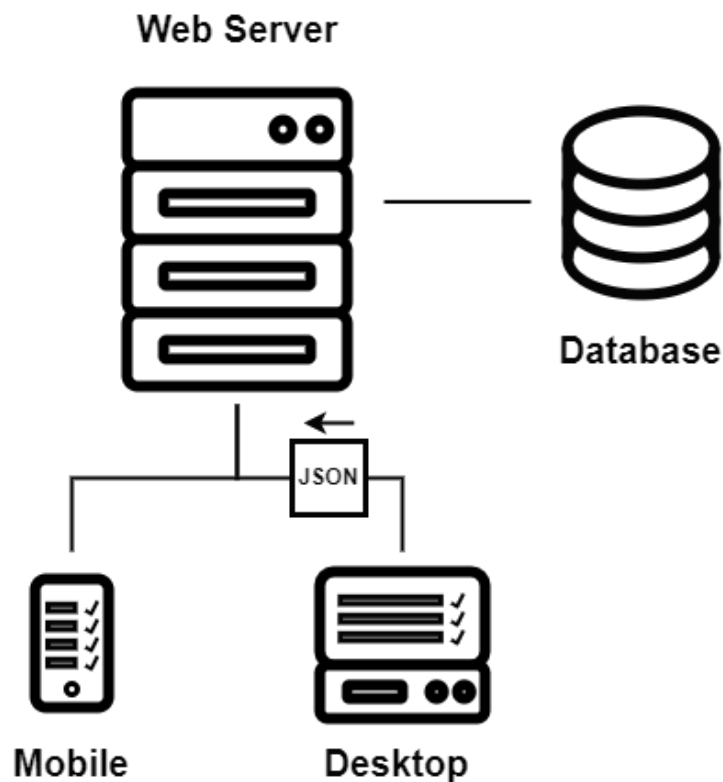
- کد وضعیت (Status Code) : کدهایی سه رقمی ناظر بر نتیجه‌ی درخواست هستند.
- سرآیند (Response Header) : اطلاعات تکمیلی در خصوص پاسخ
- بدنه پاسخ (Response Body) : پاسخ نهایی دریافتی

REST



- یک سبک طراحی و معماری است.
- هدف : ارتباط راحت تر بین سیستم‌های توزیع شده.
- یک استاندارد نیست، اما از استانداردهای دیگر استفاده می‌کند.
- مثال : از مفاهیمی از پروتکل HTTP استفاده می‌کند.
- مخفف **Representational State Transfer** است.
- **Representation** : می‌توان اطلاعات مورد نیاز را از سرور با یک استاندارد راحت (معمولا JSON) بازنمایی کرد.
- **Transfer State** : هر آنچه که باید به کاربر ارائه شود، مستقل از نوع کلاینت (وب، موبایل و...) در اختیار کاربر قرار داده می‌شود.

ویژگی‌های مهم معماری REST

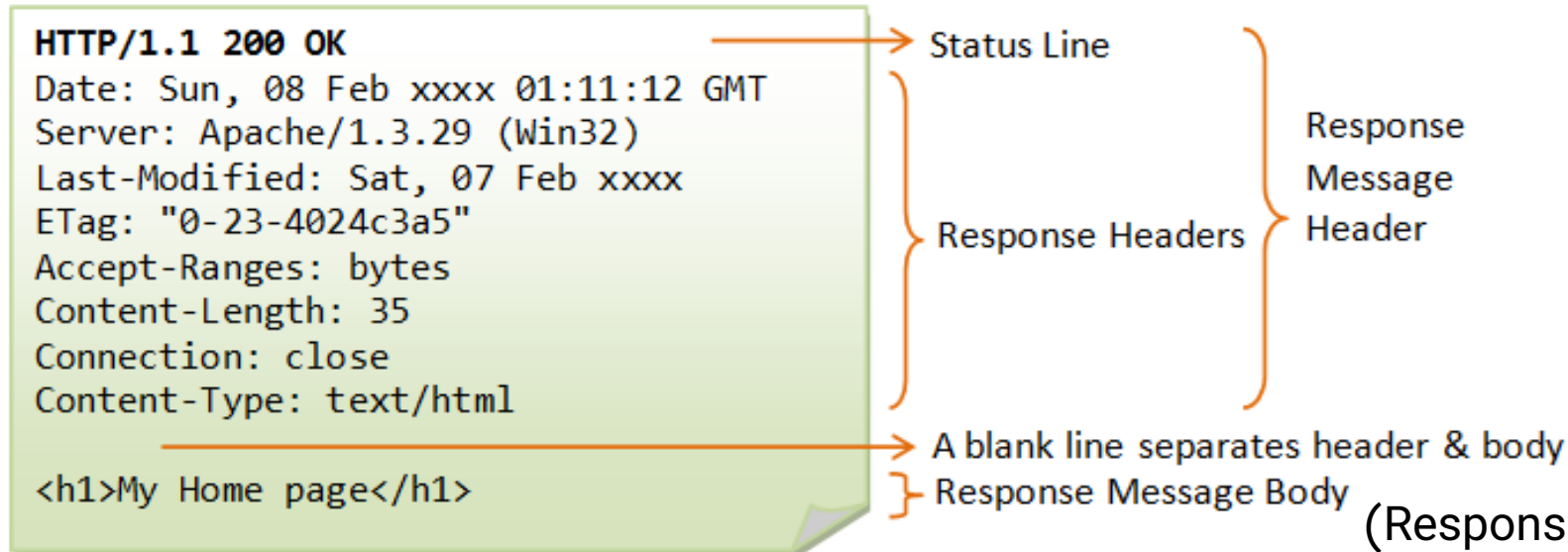


- روش ارتباط آن مبتنی بر پروتکل HTTP است.
- ارسال و دریافت اطلاعات بر مبنای نمادگذاری JSON انجام می‌شود.
- مستقل از پلتفرم و روش نمایش اطلاعات.

```
{"employees": [  
  { "firstName": "John", "lastName": "Doe" },  
  { "firstName": "Anna", "lastName": "Smith" },  
  { "firstName": "Peter", "lastName": "Jones" }  
]}
```

<https://builtin.com/software-engineering-perspectives/json>

کمی بیشتر در خصوص HTTP



▪ اطلاعات اولیه

▪ ورژن

▪ کد وضعیت

▪ کد عددی

▪ متن وضعیت

▪ اطلاعات تکمیلی (Response Header)

▪ یک خط خالی

▪ محتوا

کمی بیشتر در خصوص HTTP : کد وضعیت

فرم کد وضعیت	مفهوم
1XX	اطلاعات
2XX	موفقیت
3XX	انتقال
4XX	خطای سمت کاربر (مثل دسترسی غیر مجاز و یا آدرس اشتباه)
5XX	خطای داخلی سرور

افعال HTTP

مفهوم	فعل
دریافت محتوا از سرور	GET
ارسال داده های فرم به سرور	POST
ذخیره یک داده (و یا تغییر آن)	PUT
پاک کردن یک داده	DELETE

آزمون API

- نوعی از روش‌های آزمون نرم‌افزار است که به ما اطمینان می‌دهد که آیا API ما، کارایی لازم را برای کاربر نهایی (و یا به کارگیرنده‌ی آن) فراهم می‌کند؟
 - بر روی منطق برنامه متمرکز می‌شود.
 - جزو آزمون‌های Integration محسوب می‌شود.
- اطلاعاتی که به ما در خصوص هرکدام از پایانه‌ها می‌دهد.
 - مناسب بودن زمان پاسخ
 - اطمینان از صحت مکانیزم‌های امنیتی مثل احراز هویت
 - اطمینان از ارسال شدن و پردازش صحیح کد وضعیت

مثالی از یک سناریوی خیلی ساده

- ذخیره و بازیابی اطلاعات محصولات یک فروشگاه
- مشخصات شامل موارد زیر است : Id ، Title و Price
- عملیات مورد نظر
- ایجاد یک محصول با متد POST و ارسال اطلاعات مربوطه در Request Body
- اگر محصول با موفقیت ساخته شد، کد ۲۰۰ (موفق) و در غیر این صورت و در صورت ساخت محصول با کد تکراری، کد ۴۰۳ (درخواست غیر مجاز) دریافت شود.
- حذف آن با متد Delete و با ارائه شناسه آن محصول در Header
- در صورت موجود بودن و حذف درست، دریافت کد ۲۰۰ و در غیر این صورت دریافت کد ۴۰۴ صورت گیرد.
- دریافت فهرست محصولات و یا تک محصول با ارائه شناسه آن
- در صورت یافت نشدن آن تک محصول کد ۴۰۴ و در غیر این صورت اطلاعات محصول با کد ۲۰۰ ارائه شود.

سناریوی فرضی آزمون

۱. در ابتدا سه محصول در محصولات باشند.
۲. یک محصول با عنوان Socks و شناسه ۴ افزوده شود.
۳. امکان افزودن مجدد آن وجود نداشته باشد.
۴. محصول افزوده شده قابلیت بازیابی داشته باشد.
۵. محصول افزوده شده در فهرست تمام محصولات موجود باشد.
۶. محصول افزوده شده بتواند فقط یک مرتبه حذف شود.
۷. پس از حذف، در فهرست محصولات موجود نباشد و قابل بازیابی نیز نباشد.