

به نام خدا



## درس برنامه سازی پیشرفته

REST

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال دوم ۹۹-۹۸

---

اساتید:

مهدی مصطفی زاده، ایمان عیسی زاده، امیر ملک زاده، علی چکاه

نگارش و تهیه محتوا:

سپهر پورقناد

تنظیم داک:

امیرمهدی نامجو



## بچه‌ها سلام!

تو این قسمت قراره با همدیگه با یه سری مفاهیم مثل REST، RESTful، API، Request و ... آشنا بشیم. این مفاهیم متاسفانه جزو قسمت‌های واجب درس ap شما نیست ولی جز مهم‌ترین چیزها و مفیدترین چیزهایی هست که می‌تونید شما تو این درس یاد بگیرید (بین خودمون باشه! من کلا اون قسمتی ته ap که درباره‌ی Socket و Socket Programming هست رو درست یاد نگرفتم متاسفانه (شما مثل من نباشید) ولی عوضش اینا رو خوب بلدم). این قسمت تو صنعت خیلی کاربرد داره. اون بچه‌هایی که دوست دارن که بعد ها برن سر کار پیشنهاد میشه که حتما حتما خیلی خوب این مفاهیمی که اینجا گفته میشه رو شنیده باشن و روش یه شهود خیلی خوبی داشته باشن.

## اصل مطلب

بذارید با یه مثال براتون شروع کنم که خوب شهود بگیرید نسبت به موضوع بعد هم چند تا لینک در ادامه هست که جزئیات رو خیلی خوب براتون توضیح داده.

فرض کنید شما در یک شرکت خیلی بزرگ کار می‌کنید که حدود ۱۰۰۰۰ نفر کارمند داره. شما اونجا مسئول انبار هستید و روزانه یه عالمه درخواست میاد براتون و شما موظفید که به همشون پاسخ بدین. این انبار ۱۰ تا در هم داره که هر کسی از هر جا دلش خواست می‌تونه بیاد و هر چی دلش خواست رو هر جوری دلش خواست، درخواست کنه. شما که می‌بینید مدیریت این همه درخواست داره شما رو از پا درمیاره تصمیم می‌گیرید که در یک اقدام مقتدرانه تمامی درها رو ببندید و فقط یک در رو باز بزارید و از این به بعد هر کسی نمی‌تونه هر جور که دلش خواست با شما حرف بزنه (بالاخره شما مدیر انبارید !!!).

برای همین شما یک برگه طراحی می‌کنید و به همه می‌گید که درخواست هاشون رو توی این برگه طبق روالی که توش مشخص شده بنویسن و به شما بدن. شما هم هر برگه‌ای که گرفتید اگر مورد تایید بود همونجا به درخواست اون فرد پاسخ می‌دید. حالا می‌بیند که حجم کارتتون بالا رفته. تصمیم می‌گیرید که درهایی که بستید رو باز کنید و برای هر کدوم



یه برگه خاص طراحی می‌کنید که روال درخواست نوشتن توی این برگه‌ها با همدیگه متفاوت و این برگه‌ها رو هم در اختیار همه قرار نمی‌دین .

اون درهای انبار مثل API می‌مونن . هر کسی که میاد در خواستی داره باید درخواستش رو از طریق این API ها به شما که Server هستید برسونه . حالا اون برگه ها هم هر کدوم یه Protocol هستن . یعنی شما که Server هستید با Client ”قرارداد” می‌کنید که با یه سری قوانین خاص با هم دیگه صحبت کنید . و این Protocol ها می‌تونن با همدیگه متفاوت باشن.

## REST

حالا REST یه جور Protocol هست (اهل فن در این زمینه به این چیزا Design Pattern هم میگن)، که شما طبق اون Request رو با یه فرمت خاصی برای سرور یه آدرس (همون url یا path) خاصی می‌فرستید و اون سرور در یک قالب خاصی به شما Response می‌ده. حالا که کلیات رو گرفتید می‌تونید REST رو خیلی راحت‌تر درک کنید . لینک زیر خیلی منسجم اومده REST و API REST رو توضیح داده و تقریباً هر چیزی که نیاز رو با مثال براتون توضیح داده:

<https://www.smashingmagazine.com/2018/01/understanding-using-rest-api/>

این لینک هم خیلی خوب درباره اینکه اصن api چیه و چه انواعی داره خوب توضیح داده:

<https://www.moesif.com/blog/api-guide/getting-started-with-apis/>

این لینک هم اومده پیشرفته‌تر به REST نگاه کرده و یه عالمه best practice (برید ببینید یعنی چی! بدرد زندگیتون می‌خوره!!!!) رو مطرح کرده و واسه بررسی HTTP request و تفاوت post و get و patch و delete و یه سری مباحث پیشرفته‌تر کلی لینک از جاهای مختلف براتون آورده هر جا براتون مبهم بود می‌تونید به لینکایی که داده یه سری بزنید.

<https://www.moesif.com/blog/api-guide/api-design-guidelines/>



این لینک هم خیلی خوبه. کلی چیز به گوشتون می‌خوره که تا حالا نشنیدید. این دوستمون بیشتر از اینکه به این پردازش که REST چیه اومده درباره معماری یه سیستمی که قراره با REST کار کنه صحبت کرده. خوبه که واقعا ببینید این لینکو چون دیدتون رو نسبت به مهندسی نرم افزار عوض می‌کنه (چون احتمالا تا اینجا همش AP رو دیدید و یه عالمه کد زدین فکر می‌کنید همش همینه و جذاب نیست) ولی چالش و مشکلاتی که تو طراحی یه سیستم RESTful هست رو بررسی می‌کنه و میگه برای اینکه یه سیستم خوب داشته باشیم نیازه که چه ویژگی‌هایی داشته باشه و... اگر خیلی جاهش رو نفهمیدید اصلا نترسید و نگران نباشید. اصلا قرار نیست از این تو جایی از پروژتون استفاده کنید. ولی عوضش این لینک میتونه کلی دید بهتون بده سر اینکه مهندسی نرم افزار واقعی نزدیک به چه چیزاییه!!!

<https://blog.feathersjs.com/design-patterns-for-modern-web-apis-1f046635215>

## پس پروژه چی شد این وسط؟؟

برای پیاده سازی REST در جاوا خودتون می‌تونید سرچ کنید و به عنوان نمونه این لینک پیاده سازی یه برنامه ساده REST تو جاوا رو توضیح می‌ده:

<https://happycoding.io/tutorials/java-server/rest-api#simple-example-rest-api>