

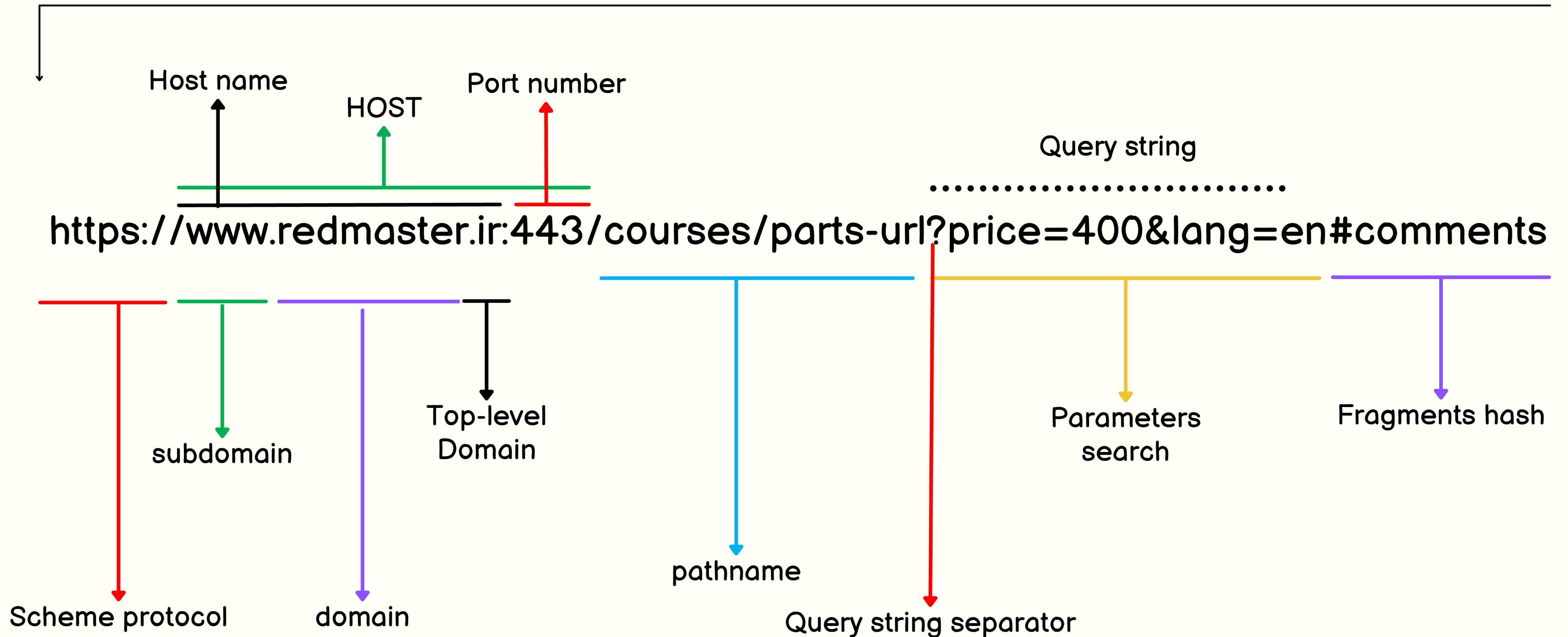
MAIN CONCEPT

REACT ROUTER DOM

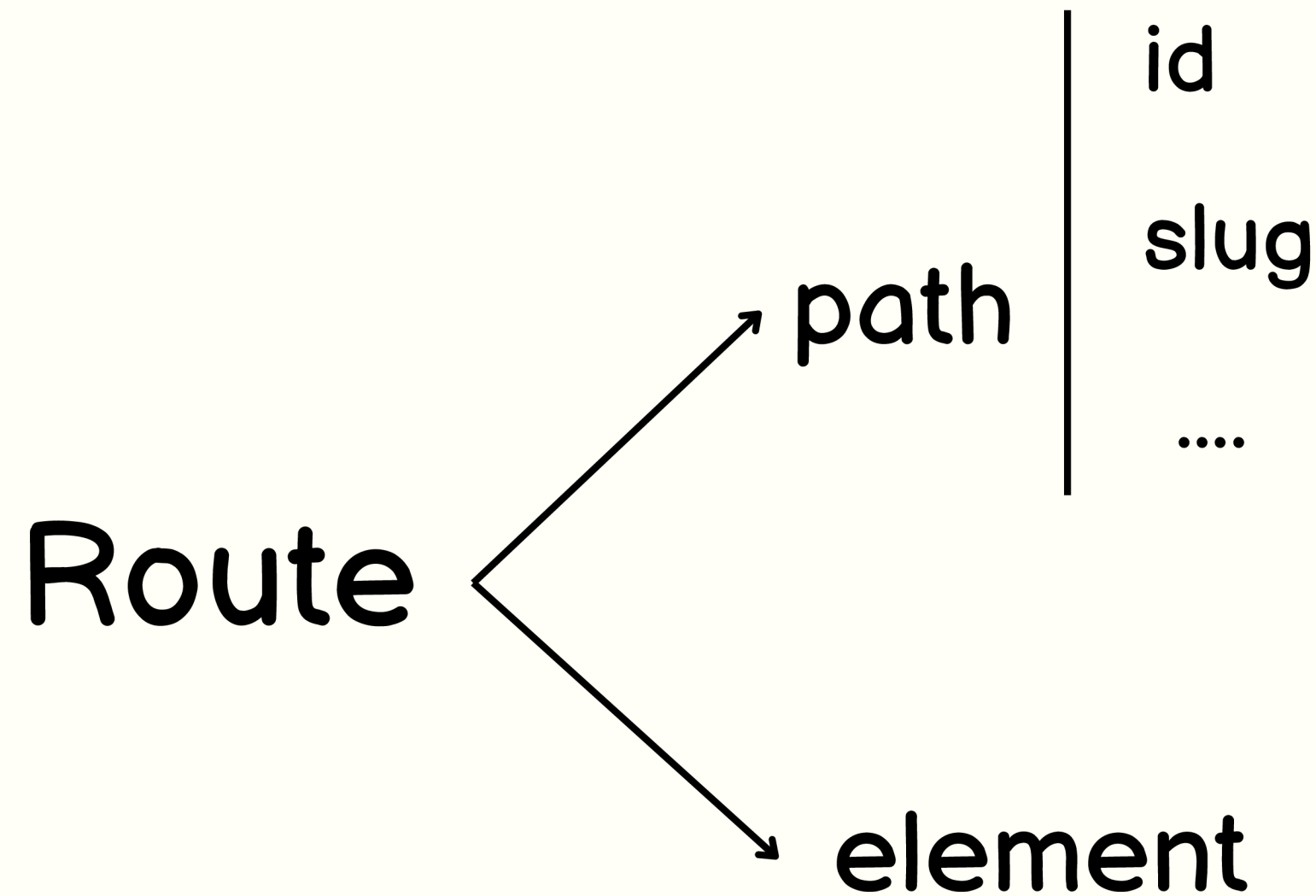
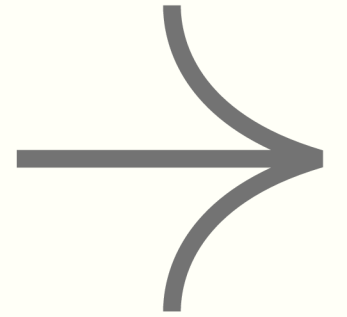
REDMASTER.ACADEMY



URL _ href



ROUTE



مسیری که ما در پروژه برای یک کامپوننت می سازیم ، حالا
میخوام بیشتر درباره Route صحبت کنم.

یک الگوی مسیر است. هنگامی که الگوی مسیر با URL فعلی
مطابقت داشته باشد، عنصر ارائه می شود.

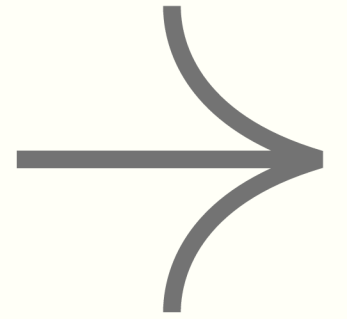


Segment

بخش هایی از یک URL یا Route که بین / قرار میگیرند.

/user/product → 2 segment

/user/product/123 → 3 segment



Dynamic Segment - Path Pattern

این مسیر ها تقریبا شبیه به URL هستند (ممکنه یکی باشند یا حداقل تا یک Segment از آنها با URL شبیه باشد)
Dynamic Route ها شامل یک کاراکتر های ویژه (:) برای تطبیق URL ها با مسیر استفاده میشوند.

شاید براتون سوال باشه کاربردش چیه ؟ چه استفاده ای دارد؟

ما میخواهیم 100 گوشی در مدل های مختلف رو صفحات فروشش رو نمایش بدیم.

اگر بخواهیم برای هر یک از این گوشی ها یک مسیر ثابت تعریف کنیم عملا به 100 مسیر ثابت نیاز داریم .

پس ما به کمک این قسمت میایم و یک الگو یا pattern می سازیم که بیاد و این 100 گوشی رو ساپورت کنه .

Dynamic Segment \longrightarrow /phone/:brand/:model

Dynamic Segment

/.../:.../

/.../*

این ها URL نیستند اشتباه نکنید اینها یک سری الگوها هستند pattern برای تطبیق مسیرها و URL ها

در این نوع segment ها به جای _: / هر چیزی میتواند قرار بگیرد. (بهش میگن URL params)

میتونیم هندلش کنیم که بر اساس این قسمت پویا یک درخواست بزنیم اگر در دیتابیس ما وجود نداشت خطا

برمیگرداند که برای درخواست ها است در ادامه به بحث کدنویسی برسیم خواهید دید.

اما این دوتا با هم متفاوت هستن حواستون باشه !!!

URL Params

مقدار تجزیه شده از URL که با یک بخش پویا مطابقت دارد یک Object هست

/ blog / :id

/ blog / 12 → URL Params { id : 12 }

/ blog / ali → URL Params { id : ali }

/ blog / true → URL Params { id : true }

URL Params

مقدار تجزیه شده از URL که با یک بخش پویا مطابقت دارد یک Object هست.

/ product / :brand / :model

/ product / apple / iphone16 → URL Params { brand : apple , model: iphone16 }

/ product / samsung / s8 → URL Params { brand : samsung , model: s8 }

/ product / nokia / 0011 → URL Params { brand : nokia , model: 0011 }

Splats

/ ... / *

/* که به اسم catchall و star شناخته میشود (بیشتر برای هندل کردن 404 استفاده میشه و پیاده سازی یکسری سناریوهای خاص)

اصطلاح catchall به سیستمی اشاره دارد که به گونه‌ای طراحی شده است تا تمامی مواردی را که در دسته‌های مشخص دیگر قرار نمی‌گیرند، جمع‌آوری یا مدیریت کند.

در واقع، مسیر catchall به‌عنوان یک گیرنده عمومی عمل می‌کند.

این الگو مسیر های زیر را شامل میشود

/blog/*

- /blog

- /blog/ali

- /blog/123

- /blog/ali/123

- /blog/ali/123/true

این الگو مسیر های زیر را شامل میشود

`/:lang?/course`

- `/course`

- `/fa/course`

- `/en/course`

- `/zn/course`

Optional Segments

`| ... ? |` _____

`| : ... ? |` _____

یک بخش یا Segment از یک مسیر را با اضافه کردن ? به انتهای مسیر به شکل optional تعریف میکنیم .

History stack



redmaster academy

وقتی که شما در صفحات وب جابجا می شوید، مرورگر آدرس‌ها را در یک پشته به نام History Stack ذخیره می‌کند.

برای دیدن تاریخچه صفحات قبلی که باز کردید، می‌تونید روی دکمه Back و Forward مرورگر کلیک راست کنید.

در React Router DOM هم این امکان رو داریم که بدون نیاز به ارسال درخواست به سرور، تاریخچه مرورگر رو دستکاری کنیم.

تاریخچه پشته، مجموعه‌ای از آدرس‌های صفحاتی هست که شما در یک جلسه مرور وب یا یک پنجره مرورگر باز کردید.

این تاریخچه به نوعی مسیر وب‌گردی شما رو نشان می‌دهد که از یک صفحه به صفحه دیگه می‌روید.

تاریخچه به شما این امکان را می‌دهد که به راحتی به صفحات قبلی و بعدی برگردید.

مرورگرها به صورت خودکار تاریخچه پشته را ذخیره می‌کنند و شما می‌تونید با استفاده از دکمه‌های Back و Forward در

مرورگر، این تاریخچه رو ببینید و دستکاری کنید.

Routing

1A

1B

1C

←

1A

History Stack

1A

1A 1B

1A 1B 1C

1A 1B

1A 1B 1A

Location



History

THANK YOU FOR WATCHING

contact us via:

Email Address:

hello@reallygreatsite.com

Website:

redmaster.academy