

# مکتب شریف

اولین بوتکمپ آموزشی - استخدامی ایران



سری هجدهم

مکتب 88





## بخش Express-MongoDB-Session

1-1. با استفاده از Express, MongoDB سیستم ثبت نام و ورود **Session-Based** پیاده سازی کنید که کاربران امکان ثبت اطلاعات و ورود به پروفایل شخصی خود را داشته باشند.

اطلاعات ثبت نامی کاربران شامل نام، نام خانوادگی، نام کاربری، رمز عبور، جنسیت، نقش کاربر و تاریخ ثبت نام در سامانه می باشد:

- نام (firstname) - **الزامی**: حداقل سه کاراکتر و حداکثر ۳۰ کاراکتر.
- نام خانوادگی (lastname) - **الزامی**: حداقل سه کاراکتر و حداکثر ۳۰ کاراکتر.
- نام کاربری (username) - **الزامی**: باید یکتا باشد.
- رمزعبور (password) - **الزامی**: حداقل طول هشت، حداقل یک عدد و یک حرف، توجه داشته باشید که password باید به صورت hash در دیتابیس ذخیره شود.
- جنسیت (gender) - **اختیاری**: این فیلد برای کاربر **اختیاری** است و اگر کاربر آن را وارد نکرد باید یک مقدار پیش فرض مثل "not-set" در نظر بگیرید.
- نقش کاربر (role) - **اختیاری**: شامل دو مقدار کاربر (user) و ادمین (admin) این فیلد برای کاربر **اختیاری** است و اگر کاربر آن را وارد نکرد باید مقدار پیش فرض "user" در نظر بگیرید.

- اگر ثبت نام موفقیت آمیز بود کاربر باید به صفحه لاگین هدایت شود.
- هنگام ورود (login) اگر نام کاربری یا رمز عبور مطابقت نداشت پیغام مناسب به کاربر نمایش داده شود.
- در صورتی که اطلاعات وارد شده صحیح باشد، کاربر به پروفایل خود منتقل گردد. توجه داشته باشید که پس ورود موفقیت آمیز کاربر امکان دسترسی به صفحه ورود و ثبت نام را ندارد.
- در صفحه پروفایل کاربر اطلاعات خود را مشاهده می کند، در صورت نیاز میتواند آنها را ویرایش کند (حتی رمز عبور) و یا اکانت خود را از سامانه حذف کند.
- قابلیت Logout نیز در صفحه پروفایل وجود دارد.
- پیاده سازی validation سمت سرور الزامی است.

طراحی صفحات کلاینت به صورت دلخواه است (سعی کنید کلاینت مناسبی پیاده سازی کنید که بتوان از آن در پروژه نهایی استفاده کرد).

## بخش Express-SQL-Sequelize

1-2. سیستم سلف دانشگاهی را در نظر بگیرید که شامل سه موجودیت **غذا**، **دانشجو** و **صورتحساب** است.

با استفاده از **Express, Sequelize** سرور زیر را پیاده سازی کنید:

- غذا (food):

- نام غذا - **الزامی**: حداقل ۳ کاراکتر و حداکثر ۳۰ کاراکتر
- قیمت غذا - **الزامی**: به ریال

- دانشجو (student):

- نام - **الزامی**: حداقل ۳ کاراکتر و حداکثر ۳۰ کاراکتر
- نام خانوادگی - **الزامی**: حداقل ۳ کاراکتر و حداکثر ۳۰ کاراکتر
- جنسیت - **الزامی**: دارای دو مقدار آقا و خانم
- کد دانشجویی - **الزامی**: طول کد دانشجویی باید **۹ رقم** و **یکتا** باشد و همچنین **نمی‌تواند** با عدد **صفر** شروع شود.

- صورتحساب (invoice):

- کد پیگیری - **الزامی**: طول کد دانشجویی باید **۱۲ کاراکتر** و **یکتا** باشد.

- رسم **نمودار ERD** و تشخیص رابطه های میان موجودیت ها
- پیاده سازی عملیات **CRUD** برای **غذا (food)**.
- پیاده سازی عملیات **CRUD** برای **دانشجو (student)**.
- پیاده سازی عملیات **CRUD** برای **صورتحساب (invoice)**.
- پیاده سازی **validation** سمت **سرور الزامی** است.

این تمرین نیازی به کلاینت (views) ندارد.



## نکات

- مهلت ارسال تمرین تا پایان روز **جمعه 1401/02/01** است. پاسخ تمرین را در **گیت هاب (github)** قرار دهید و **لینک مربوطه را در کارتا بل** خود ارسال بفرمایید.
- به ازای هر روز تاخیر در ارسال تمرین ۵ درصد نمره کسر خواهد شد.
- نام **ریپازیتوری** تمرین را به این صورت قرار دهید: Name\_hw18\_maktab88 به عنوان مثال: Mohammad\_Ali\_Kargar\_hw18\_maktab88.
- ارسال پاسخ تمرینی که در کلاس تحویل داده شده، **الزامی** است.
- در صورت لزوم یک فایل word به عنوان توضیح در کنار کدهای خود قرار دهید.
- در صورتی که سوالی دارید **در گروه راکت چت** بپرسید.