

بسمه تعالی

# تمرین شماره یک

برنامه نویسی وب

زمستان ۱۴۰۱

## فهرست مطالب

۲	۱	بخش HTTP
۲	۱.۱	سوال یک
۳	۲.۱	سوال دو
۳	۳.۱	سوال سه
۴	۲	بخش Git
۴	۱.۲	مقدمه
۴	۲.۲	بخش اول – amend
۴	۳.۲	بخش دوم – conflict merge
۴	۳	نکات تکمیلی

## ۱ بخش HTTP

### ۱.۱ سوال یک

آ - وارد سایت دیجی کالا شوید و URL سایت را با استفاده از fragmentation به گونه ای تغییر دهید که مستقیماً به تصویر enamad موجود در سایت اشاره کند. این نماد در پایین صفحه و به شکل زیر است:



ب - به ریپازیتوری React وارد شوید. URL را به گونه ای تغییر دهید که پاراگراف زیر در README هایلایت شود.

### Contributing Guide

Read our [contributing guide](#) to learn about our development process, how to propose bugfixes and improvements, and how to build and test your changes to React.

### Good First Issues

To help you get your feet wet and get you familiar with our contribution process, we have a list of [good first issues](#) that contain bugs that have a relatively limited scope. This is a great place to get started.

### License

React is [MIT licensed](#).

## ۲.۱ سوال دو

در این سوال قصد داریم با استفاده از ابزار خط فرمان curl درخواست HTTP ارسال کنیم. دستور اجرا شده در هر قسمت و نتیجه اجرای آن را به صورت اسکرین شات نمایش داده و به پرسش‌های مطرح شده پاسخ دهید.

آ - یک درخواست get به دامنه <http://google.com> ارسال کنید به طوری هدرهای پاسخ را در کنسول مشاهده کنید. کد وضعیت پاسخ چیست و به چه معناست؟ هدر Location چه چیز را مشخص می‌کند؟

ب - یک درخواست get به دامنه <http://www.google.com> ارسال کنید به طوری که کوکی‌های ست شده توسط این دامنه را در کنسول مشاهده کنید. نام کوکی‌های دریافتی را گزارش کنید.

پ - یک درخواست post به دامنه <https://reqres.in/api/users> با بدنه JSON زیر ارسال کرده و پاسخ را گزارش کنید.

```
{
  "first_name": "YOUR_FIRST_NAME",
  "last_name": " YOUR_LAST_NAME "
}
```

ت - امتیازی. یک دامنه تحریم شده مانند <https://developer.android.com> را انتخاب کرده و یک بار بدون پروکسی و بار دیگر با تنظیم پروکسی درخواستی به آن ارسال کنید و پاسخ‌های دریافتی را مقایسه کنید.

## ۳.۱ سوال سه

یک کلاینت ساده‌ی HTTP با شرایط زیر پیاده کنید (در استفاده از هر زبان یا کتابخانه‌ای آزاد هستید):  
یک آدرس HTTP در ورودی دریافت کنید. اگر آدرس وارد شده معتبر بود، برنامه یکی از متدهای زیر را به عنوان ورودی می‌گیرد.

GET / POST / PATCH/ PUT / DELETE

برای اعتبارسنجی یک آدرس HTTP می‌توانید از regex زیر استفاده کنید:

```
(?i)(http://)?(\[-a-zA-Z0-9@:._\+~#={1,256}\. [a-zA-Z0-9()]{1,6}\b)(\[-a-zA-Z0-9@:._\+~#?&//=]**)
```

برای تست regex می‌توانید از این لینک استفاده کنید.

پس از دریافت آدرس از کاربر و اعتبارسنجی آن، برنامه یک درخواست HTTP برای سرور مربوطه می‌فرستد.

• اگر پاسخ سرور یک سند html باشد، محتوا (بدنه پاسخ) به عنوان یک فایل با نام مناسب و پسوند html ذخیره می‌شود.

• اگر پاسخ سرور متن ساده باشد، بدنه پاسخ در خط فرمان چاپ می‌شود.

• اگر پاسخ سرور متنی در قالب JSON باشد، محتوا به عنوان یک فایل با نام مناسب و پسوند json ذخیره می‌شود.

- همچنین اگر سرور با کدهای وضعیت دیگری مانند ۴XX ۳XX و ۵XX به جای ۲۰۰ پاسخ دهد، انتظار می‌رود که چنین مواردی بازخوردی برای کاربر داشته باشد. به عنوان مثال برای کد ۴۰۴ می‌توانید عبارت "یافت نشد" را چاپ کنید. نیازی نیست برای تمام کدهای وضعیت پیغامی نظیر آن چاپ کنید. بلکه هدف از این بخش مدیریت کلی کدهای وضعیت مختلف و توانایی برنامه بر درک کدهای وضعیت HTTP و تفاوت قائل شدن بین کدهای خطا ۴XX و ۵XX در مقابل کدهای موفقیت ۲XX می‌باشد.

## ۲ بخش Git

### ۱.۲ مقدمه

این تمرین در راستای آشنایی بیشتر دانشجویان با Git بوده و شامل دو بخش می‌باشد. برای انجام این تمرین باید از ریپازیتوری <https://github.com/mrhashemian/git-exercise> یک private clone ایجاد کنید. پس از اتمام مهلت تمرین از شما خواسته خواهد شد تا یکی از تدریس‌یاران را برای نمره دهی به عنوان collaborator به ریپازیتوری خود اضافه کنید. برای انجام دادن هر بخش می‌بایست ابتدا به branch مربوطه checkout کنید، سپس فایل setup.sh را که داخل دایرکتوری هر قسمت هست اجرا کنید. برای توضیحات تکمیلی به فایل‌های README.md موجود در ریپازیتوری مراجعه کنید.

### ۲.۲ بخش اول – amend

در قسمت اول این بخش باید کامیت آخری که به اشتباه انجام شده است را بدون ایجاد کامیت جدید درست کنید. در قسمت دوم باید همین کار را برای کامیتی که قبل‌تر ایجاد شده است انجام دهید.

### ۳.۲ بخش دوم – conflict merge

طبق توضیحات این بخش، باید شاخه merge-conflict را یک‌بار با شاخه merge-explicit به صورتی که یک کامیت برای ادغام ایجاد شود و بار دیگر با شاخه merge-implicit به صورتی که کامیتی برای ادغام ایجاد نشود، ادغام کنید.

## ۳ نکات تکمیلی

- پاسخ به تمرین‌ها باید به صورت انفرادی صورت گیرد.
- مهلت ارسال پاسخ‌ها تا ۲۳:۵۹ روز دوشنبه ۲۹ اسفند می‌باشد.
- در صورت وجود هر گونه ابهام یا سوال در رابطه با تمرینات از گروه تلگرامی درس استفاده نموده و از تدریس‌یاران درس راهنمایی‌های لازم را کسب نمایید.