

Table

در این سوال قصد داریم یک جدول به شکل زیر طراحی کنیم:

1	2		3
4	5	6	7
8	9	10	11
	12	13	14
15	16		

بخشی از پیاده‌سازی در فایل‌های `table.html` و `style.css` انجام شده است. شما باید با توجه به جزئیاتی که در ادامه می‌آیند، فایل `table.html` را تکمیل کنید.

پروژه اولیه

پروژه اولیه را از اینجا دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
jadval_bazi
├── style.css
└── table.html
```

جزئیات

باید یک `<table>` را به گونه‌ای به `<body>` در فایل `table.html` اضافه کنید که:

- دارای ۱۶ خانه مانند شکل فوق باشد.

- شناسه (id) هر خانه به صورت cell_i باشد که i عدد داخل آن خانه است.
- تنها از tag های td ، tr و table استفاده کنید؛ زیرا برای این موارد در فایل style.css ویژگی‌هایی در نظر گرفته شده که آن‌ها را در مکان مورد نظر برای داوری در صفحه قرار می‌دهد.

فایل table.html را پس از تکمیل، فشرده و ارسال کنید.

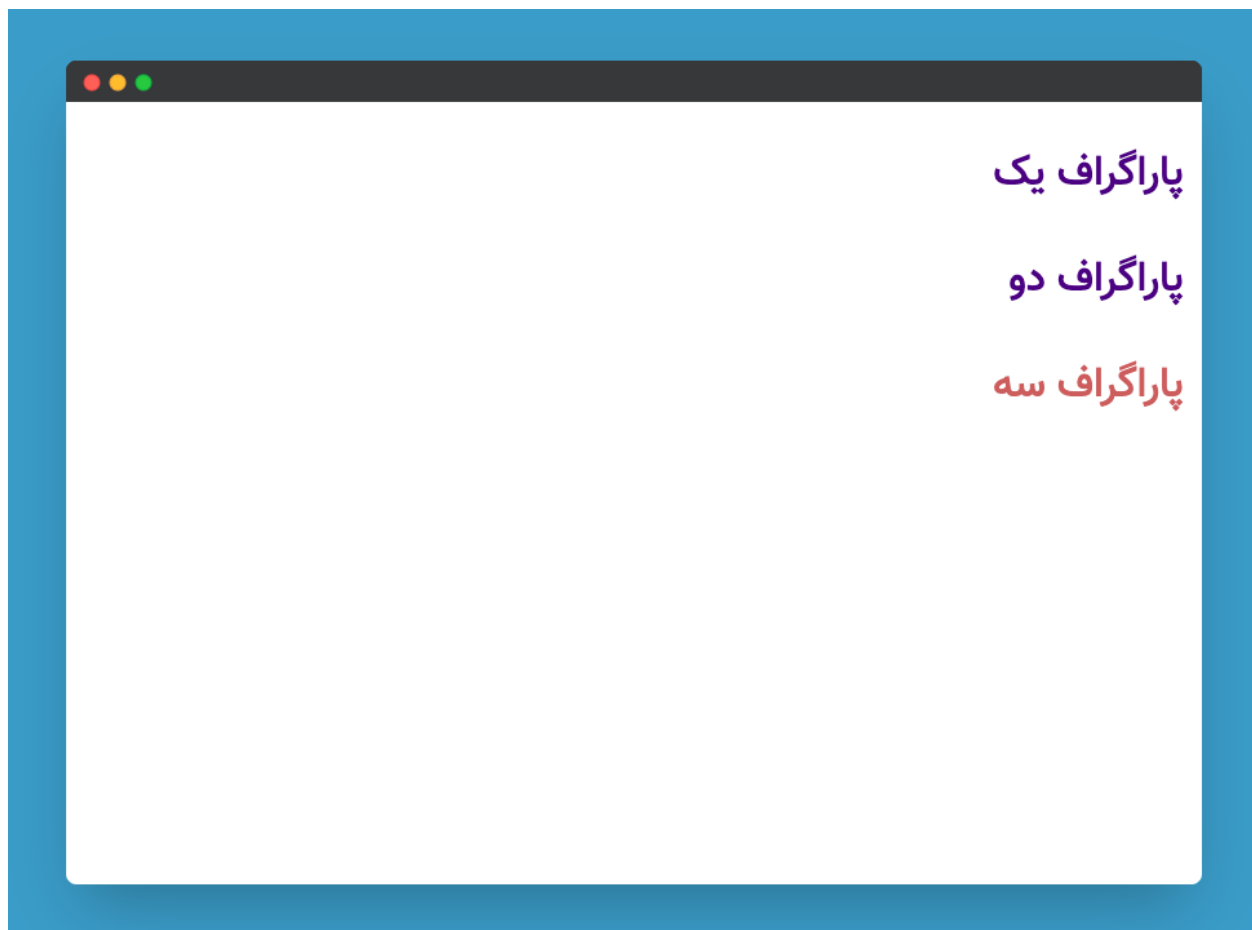
نکات

- فایل style.css بخشی از تنظیمات مربوط به ابعاد خانه‌های جدول، تنظیمات مربوط رنگ و فونت، و جهت‌گیری متن درون خانه‌ها را انجام می‌دهد. این موارد به هیچ وجه نباید از درون table.html تغییر کنند.
- دقت کنید که عناصر شناسه‌ی (id) صحیح داشته باشند.
- پروژه را با ساختار زیر ارسال کنید.

```
your-zip-file.zip
└─ table.html
```

Nested Selector

ظاهر کلی برنامه بدین صورت است:



پروژه اولیه

پروژه اولیه را از این لینک دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
nested-selector
├── index.html
└── styles.css
```

جزئیات

در فایل html داده شده دو تگ p در داخل یک div با ایدی container و یک تگ p دیگر بیرون از آن قرار گرفته است.

شما باید با استفاده از انتخاب‌کننده‌های مناسب استایلی بسازید که باعث شود تگ‌های داخل container دارای رنگ indigo و تگ خارج از آن دارای رنگ indianred باشد.

نکات

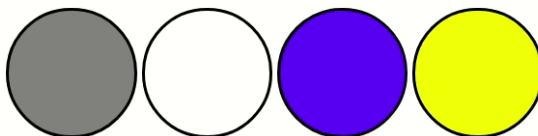
- شما تنها مجاز به اعمال تغییرات در فایل styles.css هستید.
- پروژه را با ساختار زیر ارسال کنید.

```
your-zip-file.zip
└─ styles.css
```

Background Color Switch

در این سوال قصد داریم برای دکمه هایی که از قبل تعریف شده اند عملکردی اضافه کنیم تا با کلیک بر روی هر کدام از آنها رنگ پس زمینه به همان رنگ دکمه تغییر پیدا کند. ظاهر کلی برنامه بدین صورت است:

Color Scheme Switcher



پروژه اولیه

پروژه اولیه را از [اینجا](#) دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
background-color-switch
├─ index.html
└─ styles.css
```

جزئیات

تغییرات لازم را در فایل `app.js` اعمال کنید.

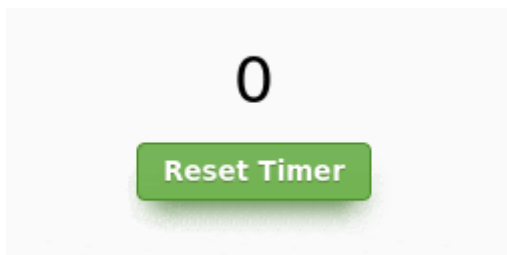
نکات

- توجه کنید که دآوری خودکار بر مبنای نام کلاس های عناصر انجام می شود.
- پروژه را با ساختار زیر ارسال کنید.

app.js

Timer

در این سؤال قرار است که یک زمان‌سنج ساده با JavaScript طراحی کنید. به این صورت که از لحظه load شدن زمان شروع به افزایش کند و با فشردن دکمه‌ی Reset Timer این زمان مجدداً صفر شود.



پروژه اولیه

پروژه اولیه را از [اینجا](#) دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
timer
├── index.html
└── style.css
```

جزئیات

شما باید با اضافه کردن یک فایل `app.js` این زمان‌سنج را پیاده‌سازی کنید. پس از هر بار لود شدن صفحه و نیز فشردن دکمه‌ی Reset Timer این زمان باید مجدداً صفر شود.

نکات

- زمان باید در عنصر با کلاس `timer` نمایش داده شود (نام کلاس در داوری تاثیر دارد).
- در این سؤال کد شما **نباید** به ساعت سیستم وابسته باشد (مثلاً نباید از `Date` استفاده کنید). عدد زمان‌سنج نباید با تغییر ساعت به هم بریزد.
- پروژه را با ساختار زیر ارسال کنید.

app.js

Desired Input

در این سوال باید به یک فیلد ورودی کمک کنید تا اعداد بزرگتر از ۱۰۰۰ یا کوچکتر از ۱ را تشخیص دهد. در فایل‌های اولیه، فیلد ورودی به شکل زیر پیاده‌سازی شده است.

Desired Input

همان‌طور که می‌بینید، با دریافت ورودی‌های مختلف رنگ پس‌زمینه‌ی آن ثابت است. شما باید فایل اولیه‌ی `desired_input.html` را به نحوی تکمیل کنید که با وارد کردن اعداد بزرگتر از ۱۰۰۰ یا کوچکتر از ۱، رنگ پس‌زمینه‌ی آن تغییر کند. نتیجه‌ی مورد انتظار به صورت زیر است.

Desired Input

پروژه اولیه

پروژه اولیه را از [اینجا](#) دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
desired_input
├── desired_input.html
└── w3.css
```

جزئیات

- با دریافت هر عدد کوچکتر از ۱ یا بزرگتر از ۱۰۰۰، رنگ پس‌زمینه‌ی ورودی که شناسه‌ی آن `input_number` است، به `#F5B7B1` تغییر می‌کند.
- اگر بلافاصله ورودی تغییر کند، رنگ پس‌زمینه هماهنگ با آن تغییر می‌کند.
- رنگ اولیه‌ی پس‌زمینه `#FFFFFF` است.

نکات

- توجه کنید که داوری خودکار بر مبنای شناسه (`id`) عناصر انجام می‌شود؛ پس شناسه‌ی هیچ عنصری را تغییر ندهید.
- پروژه را با ساختار زیر ارسال کنید.

```
your-zip-file.zip
├─ desired_input.html
└─ app.js(optional)
```

Autocomplete

می‌خواهیم با استفاده از جاوااسکریپت بر اساس یک لیستی از برخی استان های ایران ، یک Autocomplete مطابق با شکل زیر پیاده سازی کنیم.

A form consisting of a text input field with the placeholder text 'Cities' and a button labeled 'Select' to its right. A mouse cursor is pointing at the bottom center of the input field.

پروژه اولیه

پروژه اولیه را از [اینجا](#) دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است.

```
autocomplete
├─ app.js
├─ index.html
└─ styles.css
```

جزئیات

با شروع به تایپ کردن در `input` یک لیستی از پیشنهادات برای کاربر ظاهر می شود با کلیک کردن بر روی هر کدام از آنها ، `input` مقدار دهی می شود. تغییرات لازم را در فایل `app.js` اعمال کنید.

نکات

- توجه داشته باشید هر پیشنهاد باید در یک عنصر `div` با کلاس `item` ساخته شوند.
- در صورت یافت نشدن یک پیشنهاد می بایست `Not Found!` در یک عنصر با کلاس `not-found` ساخته شود.
- به محض کلیک شدن بر روی یک پیشنهاد مقدار داخل `input` به آن تغییر می کند.
- در صورت کلیک شدن بر روی `body` باید لیست ظاهر شده از بین برود.
- توجه کنید که دآوری خودکار بر مبنای نام کلاس های عناصر انجام می شود.
- پروژه را با ساختار زیر ارسال کنید.

`app.js`