

Front-End Developer وظیفه

طراح و کدنویس فرانت اند وظیفه تبدیل یک طرح آماده شده از یک وبسایت را به کد دارد. کدنویسی فرانت شامل طراحی ساختار و لایه بندی صفحات، قرار دادن محتوا، ساخت دکمه ها، قرار دادن عکس ها، ساخت منو ها، لینک دهی صفحات، ساخت انیمیشن ها، ساخت فرم ها و اعتبار سنجی آن ها و ... می شود.

طرح اولیه توسط یکی از 3 روش زیر در اختیار شما قرار می گیرید:


- آماده شدن طرح UI/UX توسط گرافیکست – حرفه ای
- با توجه به خواست کارفرما سایتی به عنوان نمونه به شما معرفی می شود (سایتی مثل دیجیکالا میخوام) – غیر حرفه ای
- ایده گرفتن خودتان از سایت هایی مثل dribbble ، screenlance ، awwwards ، land-book و ...

تکنولوژی های مورد استفاده شما


هسته اصلی سایت : HTML, CSS, JavaScript


کتابخانه ها و فریم ورک ها: React (Angular, Vue), SASS, Bootstrap (Tailwind)

مدیریت پروژه و کدها به صورت آنلاین: GitHub, Netlify

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

7 مرحله برای ساخت یک وبسایت کامل از صفر تا صد

1) پروژه خود را تعریف کنید


- هدف پروژه چیست؟ وبلاگ، نمونه کار، شرکتی، فروشگاه و...
- مخاطبین شما چه کسانی هستند؟

2) برنامه ریزی کنید


- آماده کردن محتوای سایت: متن ها، عکس ها، ویدیو ها، آیکون ها، کدهای رنگی، فونت ها و...
- راجع به وراثت سایت فکر کنید. (این که چه قسمتی داخل چه قسمت دیگری باشد)
- در مورد navigation یا همان منوی اصلی سایت فکر کنید. (اینکه شامل چه آیتم هایی می شود و...)


3) اگر طرح UI/UX آماده ندارید شروع کنید طرح ها و ایده هاتون را بکشید

- الهام بگیرید و در مورد طرحتون خوب فکر کنید.
- ایده هاتون را به صورت ساده بکشید.
- هر تعدادی که میخواید طرح اولیه رو کاغذ بکشید. (نیاز نیست کامل و عالی باشند)
- هیچوقت شروع به طراحی نکنید اگر نمی دونید که چی قراره بسازید.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 


tavakolianparham@gmail.com 

(4) شروع طراحی و ساخت وبسایت


- طراحی فرانت وبسایت با استفاده از تکنولوژی هایی که بلدید.
- از اتود های اولیه، ایده ها و محتواهایی که در مراحل 1 تا 3 آماده کردید استفاده کنید.
- مهم ترین بخش این بخش است!


(5) هنوز کار تمام نشده!

- بهینه سازی سرعت و کارایی وبسایت: قبل از هر چیزی باید از عکس های بهینه استفاده کنید، زیاد http request نداشته باشید، متن هاتون به اندازه باشه، انیمیشن های خیلی سنگین و غیر ضروری استفاده نکنید.
- سئو: این حوزه از وظایف اصبی شما نیست اما باید با بخشی از آن آشنا باشید.
 - داخلی: به محتوای صفحات مربوطه، استفاده از alt برای عکس ها، استفاده از Meta tag ها در head، استفاده از کلمات کلیدی در محتوا و انتخاب درست H1 ها
 - خارجی: برای تبلیغ و معرفی سایت در سایت های دیگر و استفاده از بک لینک ها.
 - تکنیکال: بهینه سازی جنبه های فنی سایت برای بهبودی رتبه موتورهای جستجو و ساده کردن ساختار وبسایت برای خزش یا Crawling

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

(6) لانچ و آپلود کردن شاهکارتون


- تمام آنچه نیاز دارید یک وب سرور است که وظیفه میزبانی وبسایت شما رو بر عهده بگیره.
(رایگان و برای نمونه کار : GitHub, Netlify)

(7) نگه داری و مراقبت از سایت (Site Maintenance)


- عملکرد و رفتار مخاطبین را تحت نظر بگیرید و هر زمانی که نیاز بود، تغییرات لازم را اعمال کنید (استفاده از Google Analytics)
- محتوای سایت را به روز نگه دارید اگر مقاله دارید.


نکات کاربردی برای پروژه (موارد بیشتر در پیچ های اینستاگرام)

- (1) سایت های دانلود عکس: unsplash.com, pexels.com, randomuser.me و برای دانلود وکتور و عکس ها می تونید از سایت Freepik هم استفاده کنید.
- (2) سایت های دانلود آیکون: هر سایتی که میخواید از آیکون هاش استفاده کنید داخلش روش استفاده از آیکون ها رو هم کامل توضیح داده. به طور مثال برای استفاده آیکون های Font Awesome کافیست لینک فایل css زیر رو داخل سند Html بزارید و بعد وارد سایتش بشید و قطعه کد آیکون مورد نظر رو بردارید و استفاده کنید.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

```
<link
  rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.1/css/all.min.css"
/>
```


(3) بهترین روش استفاده از آیکون ها استفاده از SVG ها است که روش مخصوص خودش را دارد. ولی اگر با استفاده از فونت آیکون ها آشنایی دارید از فونت آیکون ها استفاده کرده و بعد با استفاده از color, font-size و بقیه استایل های مربوط به فونت بهش استایل بدید.

(4) کار مداوم با ابزار Devtools در Google Chrome که با inspect باز میشه.


(5) استفاده از Pseudo Element ها و Pseudo Class ها


- Pseudo Element: برای استایل دهی به بخش خاصی از عناصر صفحمون مثل ::before که برای اضافه کردن محتوا به قبل از عنصر مورد نظر و ::after برای اضافه کردن محتوا به بعد از عنصر مورد نظر
- Pseudo Class: استایل دهی به عناصر صفحه وقتی در حالت خاصی هستند. مثل :hover, :first-child, :not

(6) استفاده از افزونه های کاربردی VS Code مثل prettier و...

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

7) گذاشتن favicon برای صفحاتتون با استفاده از سایت favicongenerator که با دادن لوگو بهش، بهتون چند خط کد و فایل zip بهتون می‌دهد و کدها را در تگ head گذاشته و با فایل‌هایی که از حالت فشرده خارج کردید لینک می‌کنید.


8) استفاده از CSS Variables مثل:

```
:root {
  --color-blue: rgb(125, 61, 194);
  --default-font-size: 15px;
}


body {
  font-size: var(--default-font-size);
}


div {
  background-color: var(--color-blue);
}
```

** root باید در بالای CSS تعریف شود.

02191035653 


@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 


tavakolianparham@gmail.com 

8) نوشتن HTML و CSS خوب

- طراحی ریسپانسیو:
 - تنظیم لایه بندی صفحه به شکلی که انعطاف پذیر باشد
 - مدیا کوئری ها
 - عکس های ریسپانسیو
 - استفاده از واحد های درست (استفاده از % و rem به جای px)
- خوانایی و تمیزی کد
 - تمیزی کد
 - قابل فهم بودن کد
 - قابلیت استفاده دوباره از کد در آینده
 - به روز بودن کد
 - نحوه صحیح اسم گذاری فایل ها
 - نحوه صحیح اسم گذاری کلاس ها
- کارایی وب
 - استفاده کمتر از درخواست های HTTP
 - نوشتن کدهای کمتر تا حد امکان
 - فشرده سازی کد بعد از پایان پروژه با ابزار های موجود
 - استفاده از یک پیش پردازنده یا preprocessor برای CSS مثل SASS
 - استفاده کمتر از عکس ها

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

- توجه به رزولوشن عکس ها و استفاده درست از آن ها با توجه به نیازمون (استفاده از عکس با رزولوشن بالا در جایی که نیاز نیست اشتباه است)

9) تنظیمات اولیه در CSS یا همان CSS Reset

```
*, *::after, *::before { margin: 0; padding: 0; box-sizing: inherit; }
```

```
body { box-sizing: border-box; }
```

10) استفاده از Emmet در VScode برای تسریع در نوشتن HTML.

Emmet در واقع یک پلاگین هست که با یک سری شُرت کد به ما کمک می‌کنه کدهامون رو سریع تر بنویسیم و این پلاگین به صورت پیش فرض روی VScode نصب است.


به طور مثال در موارد پایین اگر شُرت کد سمت چپ را بنویسید، نتیجه سمت راست را خواهید گرفت:

```
.my-div => <div class="my-div"></div>
```


```
ul.list => <ul class="list"></ul>
```


```
a#link => <a href="" id="link"></a>
```

```
li*6 => <li></li> <li></li> <li></li> <li></li> <li></li> <li></li>
```

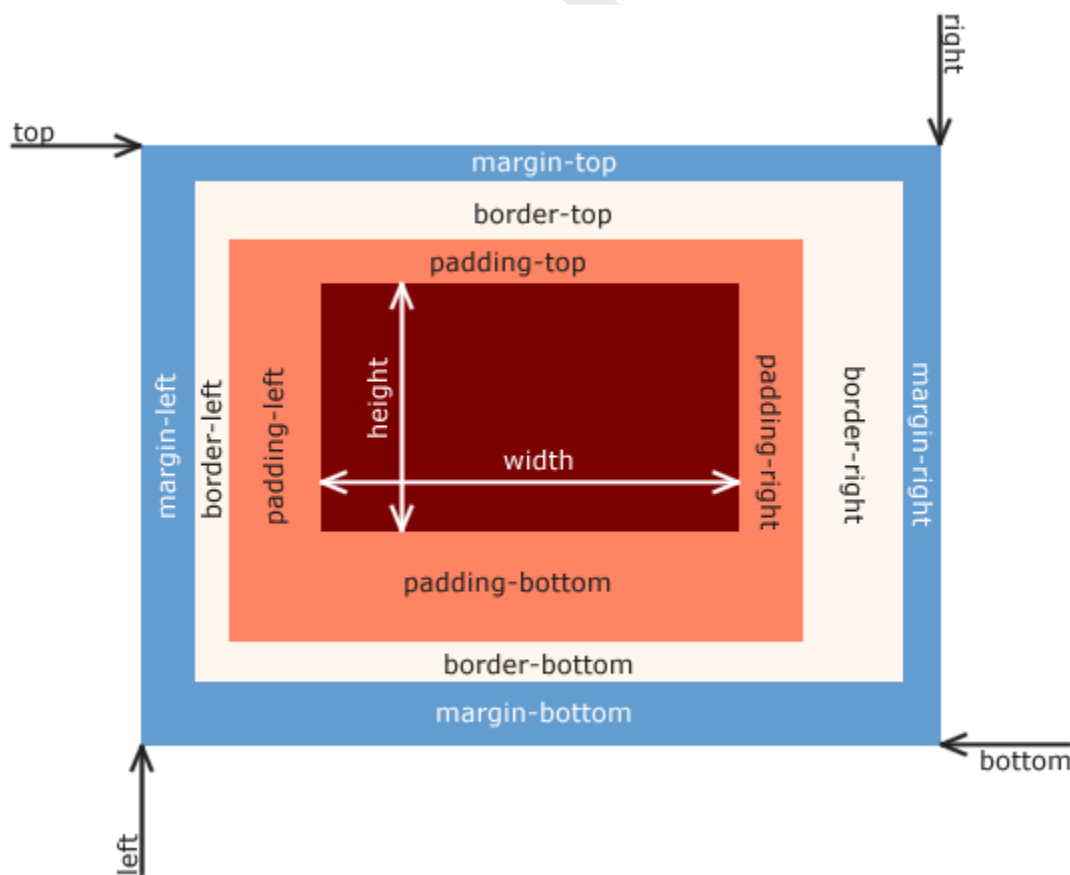
02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 


ecolearn1.ir 


tavakolianparham@gmail.com 


CSS BOX MODEL




با گذاشتن box-sizing روی border-box عرض و ارتفاع ما علاوه بر محتوا، padding و border هم شامل می‌شود.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

3 روش وسط قرار دادن عناصر با CSS


1)


```
parent-div {
  position: relative;
}


div {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}
```


2)

```
parent-div {
  display: flex;
  align-items: center;
  justify-content: center;
}
```

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

3)

```
div {
  display: block;
  margin: auto;
}
```

اسم گذاری کلاس ها – BEM METHODOLOGY

روش های مطرح اسم گذاری کلاس ها:

Object oriented CSS


S Max

BEM : B~Block E~Element M~Modifier (روش پیشنهادی)


Block: یک جزء مستقل که به تنهایی دارای معنا است و نقش container یا همان المانی را بازی می کند که بقیه المان ها داخلش هستند


Element: فرزند چه مستقیم و چه غیر مستقیم Block است که به تنهایی معنای مستقل ندارد

Modifier: یک ورژن متفاوت از Element یا Block است

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 


```
<div class="card">
  <div class="card__front">
    <h2 class="card__heading"></h2>
    <a href="#" class="card__btn--green"></a>
  </div>
  <div class="card__back">
    <p class="card__paragraph"></p>
    <a href="#" class="card__btn--white"></a>
  </div>
</div>
```

اصول پایه طراحی ریسپانسیو یا واکنشگرا


طراحی ریسپانسیو به این معنی است که وبسایت طراحی شده توسط شما باید در همه دستگاه ها از قبیل دسکتاپ ها، تبلت ها و موبایل ها به خوبی نمایش داده بشه.


لایه بندی انعطاف پذیر

برای اینکه محتوای سایت ما به بهترین شکل با صفحه نمایش کاربر وفق پیدا کند بهتره به جای استفاده از واحد های ثابتی مثل px از rem یا % برای تعریف اندازه هامون استفاده کنیم.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

تصاویر ریسپانسیو و منعطف

تصاویر داخل صفحه ما به شکلی متفاوت از محتوای متنی ما رفتار می‌کنند، پس باید روی تک تک عکس‌ها به صورت جداگانه تمرکز کنیم و با صفحه مروگر کاربر به راحتی وفق پیدا می‌کنند.


Media Queries

مدیا کوئری‌ها برای تغییر دادن استایل‌های CSS در عرض خاصی از صفحه نمایش (breakpoint) استفاده می‌شوند.


```
html {
  font-size: 62.5%; // 1rem = 10px
}


@media only screen and (max-width: 900px) {
  html {
    font-size: 50%; // 1rem = 8px
  }

  div {
    width: 60%;
  }
}
```

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 



- نوشتن در یک فایل جدا یا در انتهای فایل CSS

- دقت شود همیشه اول عرض بزرگتر میاد و بعد عرض کوچکتر


انتخاب breakpoint ها

1) روش بد: استفاده از عرض دستگاه های مشهور مثل اپل برای breakpoint ها که اغلب از این روش استفاده می کنند اما دو مشکل اساسی دارد. یک اینکه این روش آینده نگرانه نیست و ممکنه عرض این دستگاه ها تغییر کند و دوم اینکه شما با این روش همه کاربران با همه دستگاه های دیگر رو نادیده می گیرید


2) روش خوب: عرض همه دستگاه های پر استفاده رو پیدا کنیم و با قرار دادن آن ها در کنار هم به یک اشتراکی برسیم. برای پیدا کردن عرض ها می توانید از سایت Stat Counter استفاده کنید


عرض های پیشنهادی من: 600px, 900px, 1200px, 1800px

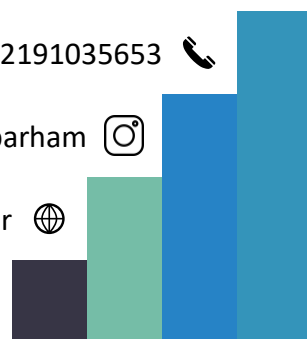
3) روش عالی: اعمال Media Query ها هر زمان که لازم شد (زمان بر است)

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 



ساخت لایه بندی صفحه (Float, Flexbox, Grid)

برای ساخت لایه بندی صفحات و اینک عناصر را در جایی که می‌خواهید قرار دهید می‌توانید از یکی از موارد زیر استفاده کنید:

:FLOAT


در گذشته تقریباً تمامی وبسایت‌ها با استفاده از Float ساخته می‌شدند. کار با Float راحت‌تر فقط باید یادتون باشه باید بعد از استفاده از Float از Clear هم استفاده کنید.

چرا؟ وقتی شما از Float استفاده می‌کنید، المان مربوطه از جریان عادی صفحه خارج شده و شناور خواهد شد و متن‌ها و المان‌های خطی (inline) به دور این المان Float شده قرار خواهند گرفت. (می‌تونید با تست کردن، بهتر متوجه این موضوع بشید)

فرض کنید من روی div، Float اعمال می‌کنم و بعد هم Clear می‌کنم:


```
div {
  float: left;
}

div::after {
  content: '';
  display: table;
  clear: both;
}
```

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

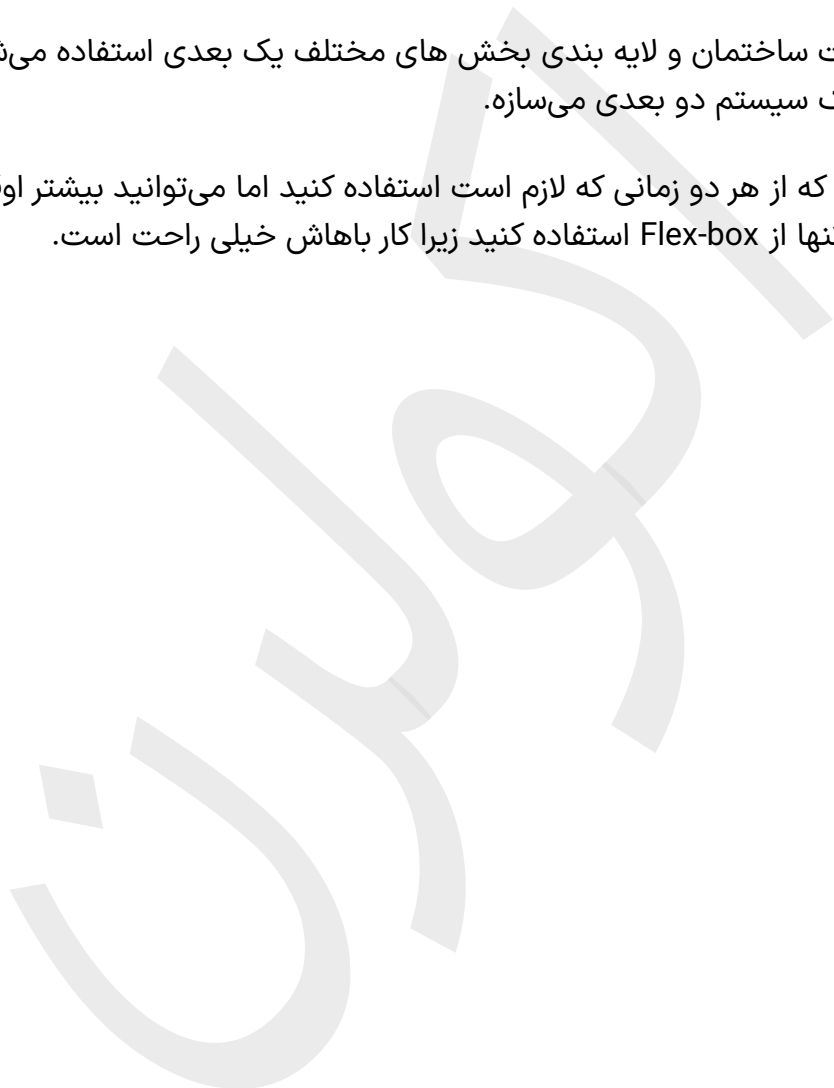
tavakolianparham@gmail.com 




:Grid & Flexbox


Flexbox برای ساخت ساختمان و لایه بندی بخش های مختلف یک بعدی استفاده می شود و css Grid برای ما یک سیستم دو بعدی می سازد.


بهترین کار این است که از هر دو زمانی که لازم است استفاده کنید اما می توانید بیشتر اوقات و برای پروژه پیش رو تنها از Flex-box استفاده کنید زیرا کار باهاش خیلی راحت است.

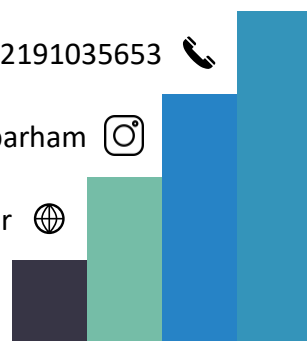


02191035653 

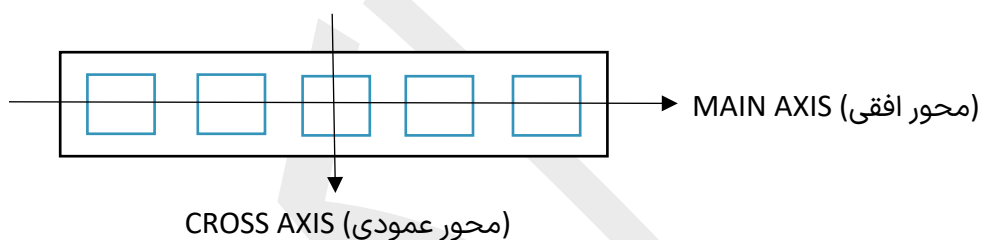
@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 


tavakolianparham@gmail.com 




برگه تقلب FLEXBOX برای استفاده شما




وقتی ما می‌خواهیم از Flexbox استفاده کنیم ابتدا باید Container یا همان Parent مربوطه رو پیدا کنیم و `display: flex;` رو براش بنویسیم اینطوری اون Container تبدیل به Flexbox و Container همیشه و همه فرزندهاش تبدیل میشن به Flex items.

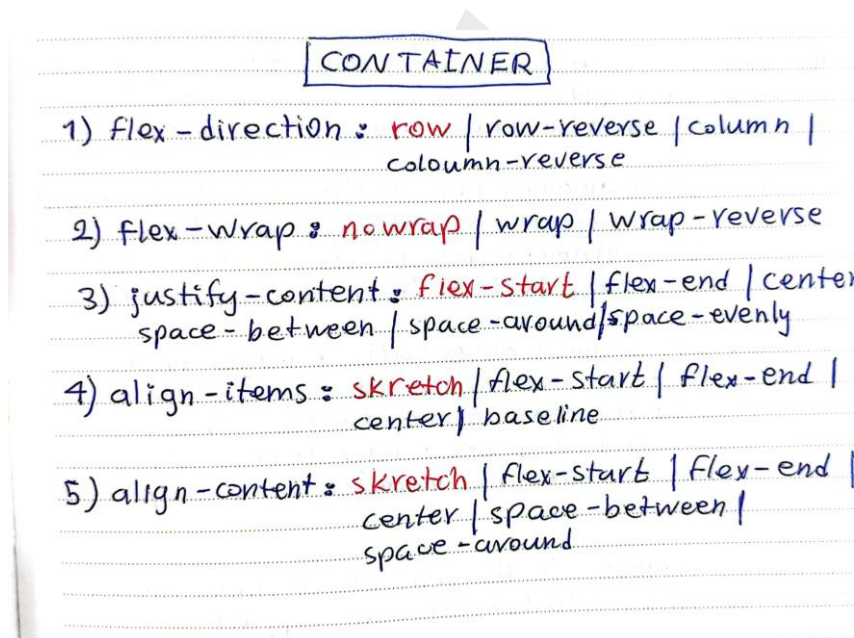
02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

پراپرتی های CSS برای FLEX CONTAINER:




justify-content برای مرتب سازی آیتم ها در محور افقی (Main Axis) استفاده می شود و مقادیر کاربردی این ها هستند:


justify-content: center باعث می شود همه آیتم ها در محور افقی در وسط قرار بگیرند.


justify-content: space-between باعث می شود آیتم ها تا جایی که می توانند از هم فاصله بگیرند.

justify-content: space-evenly باعث می شود آیتم ها همه به صورت مساوی از هم فاصله بگیرند و همچنین از چپ و راست CONTAINER هم به همون اندازه فاصله بگیرند.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 

2 مقدار flex-start و flex-end هم از اسمشون مشخصه که چه کاری انجام می‌دهند


مقدار پیش فرض **align-items** روی stretch است یعنی تمام آیتم‌ها باید از لحاظ ارتفاع با آیتمی که بیشترین ارتفاع رو داره برابر بشن و برای این کار آیتم‌های کوچکتر کشیده می‌شوند تا به اون ارتفاع برسند.

مقادیر دیگر برای **align-items**: center, flex-start, flex-end هست که آیتم‌ها را در محور افقی مرتب می‌کند.


flex-direction جهت قرارگیری آیتم‌های داخل container را مشخص می‌کند اگر می‌خواهید آیتم‌های کنار هم در یک خط افقی قرار بگیرند از **flex-direction: row** استفاده کنید و اگر می‌خواهید به صورت عمودی و زیر هم قرار بگیرند از **flex-direction: column** استفاده کنید. (دقت کنید وقتی **flex-direction** را روی column می‌زارید محور افقی و عمودی هم باهاش می‌چرخه یعنی justify-content روی محور عمودی اعمال میشه)

flex-wrap مشخص می‌کند که زمانی که آیتم‌ها دیگر فضایی نداشته باشند، به خط بعدی بروند (**flex-wrap: wrap**) یا خیر و خود آیتم‌ها بالاجبار کوچک بشوند تا در آن فضا جا بشوند (**flex-wrap: no-wrap**).

flex-wrap استفاده زیادی در بحث طراحی ریسپانسیو دارد و بهتره استفاده بشود.

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 


ecolearn1.ir 


tavakolianparham@gmail.com 





gap برای تعریف حداقل فاصله بین آیتم ها استفاده می‌شود. مثلا اگر `gap: 30px` بزاید فاصله بین آیتم ها از 30px کمتر نمی‌شود.

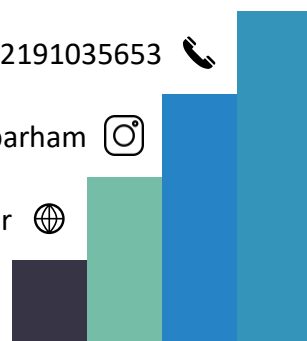
آوردن

02191035653 

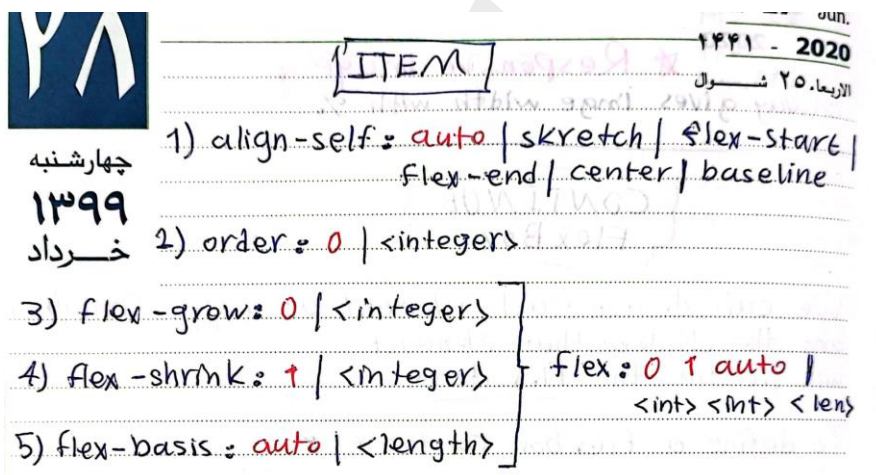
@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 



پراپرتی های CSS برای FLEX ITEMS:



وقتی Flexbox items کار می‌کنیم باید از پراپرتی flex استفاده کنیم که 3 تا مقدار میگیره:

flex: flex-grow flex-shrink flex-basis که flex-grow یا مقدار 0 میگیره یا مقدار 1 و به این معنی هستند که اون آیتم تا جایی که میتونه عرض رو اشغال کنه (1) یا نه (0). بعدی flex-shrink که این هم مقدار 0 یا 1 میگیره و به این معنی هست که اگر صفحه نمایش کوچک شد این آیتم هم باهاش کوچک بشه (1) یا خیر (0).

همچنین می‌تونید این 3 پراپرتی رو جدا هم برای آیتم ها تعریف کنید ولی اینکه همرو در یک خط و مثلا با استفاده از 50% 0 0 flex: تعریف کنید حرفه ای تر است.

وقتی فقط برای یک آیتم می‌نویسیم; flex: 1 یعنی اون آیتم هر چقدر فضای خالی مونده رو اشغال کنه.

02191035653

@ecolearn1 | @ecofrontend | @frontendparham


ecolearn1.ir


tavakolianparham@gmail.com





برای آیتم ها ما `align-self` هم داریم که می‌تونیم بر خلاف پراپرتی ست شده در `Flex Container` (در محور افقی وقتی `flex-direction: column`) هر آیتم را در محور عمودی جداگانه مرتب کنیم.

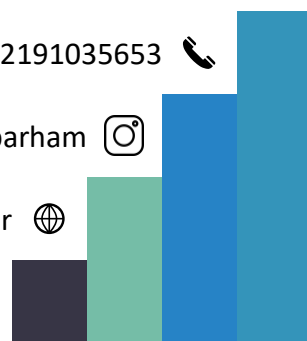


02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 




پروژه


نکات:


- لوگو، عکس های اصلی و فونت بهتون داده می شه (اگر خواستید می توانید عکس های دیگری پیدا کنید و جایگزین کنید اما باید مرتبط باشد)
- کد های رنگی سایت را با استفاده از Extension کروم به نام Colorzilla پیدا کنید یا داخل Vscode با استفاده از Color Picker یک رنگ نزدیک به آن رنگ را انتخاب کنید.
- سایت باید فارسی و از راست به چپ باشد.
- ساختار بندی سایت بهتره به این شکل باشد

```
<body>
  <header></header>
  <main>
    <section></section>
    <section></section>
  </main>
  <footer></footer>
</body>
```

02191035653 


@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 


tavakolianparham@gmail.com 

معیارهای امتیاز دهی پروژه

- (1) میزان مشابه بودن طرح شما با طرح اصلی
 - (2) ریسپانسیو بودن تا عرض 320px
 - (3) تمیزی کد
 - (4) خوانا بودن کد
 - (5) سرعت بالا اومدن صفحه
 - (6) نوشتن کد درست و معتبر (داشتن فقط یک H1 در صفحه، نوشتن alt برای عکس ها و...)
 - (7) عدم کپی کردن از بقیه (مشابه بودن همه استایل ها و کلاس های شما با فردی دیگر نمره منفی دارد)
 - (8) خلاقیت در طراحی (به کار بردن خلاقیت در طراحی انیمیشن ها و هاور افکت ها)
 - (9) اتمام در زمان مقرر
- ★ اگر سوالی داشتید در گروه تلگرام بپرسید که بقیه دوستان هم بتوانند استفاده کنند.
- ★ مدت زمان انجام پروژه 10 روز است. (قابل تغییر در شرایط خاص)
- ★ تحویل پروژه در 2 مرحله صورت می گیرد: (1) تحویل حضوری و انفرادی توسط شما و دادن توضیحات بخش های مختلف پروژه (2) بررسی کدها و خروجی توسط من.
- ★ نمره پروژه از 100 محاسبه خواهد شد.
- ★ لینک عکس های پروژه در [dribbble](https://dribbble.com/Digital-Banking-Page): [Digital-Banking-Page](https://dribbble.com/Digital-Banking-Page)

02191035653 

@ecolearn1 | @ecofrontend | @frontendparham 

ecolearn1.ir 

tavakolianparham@gmail.com 