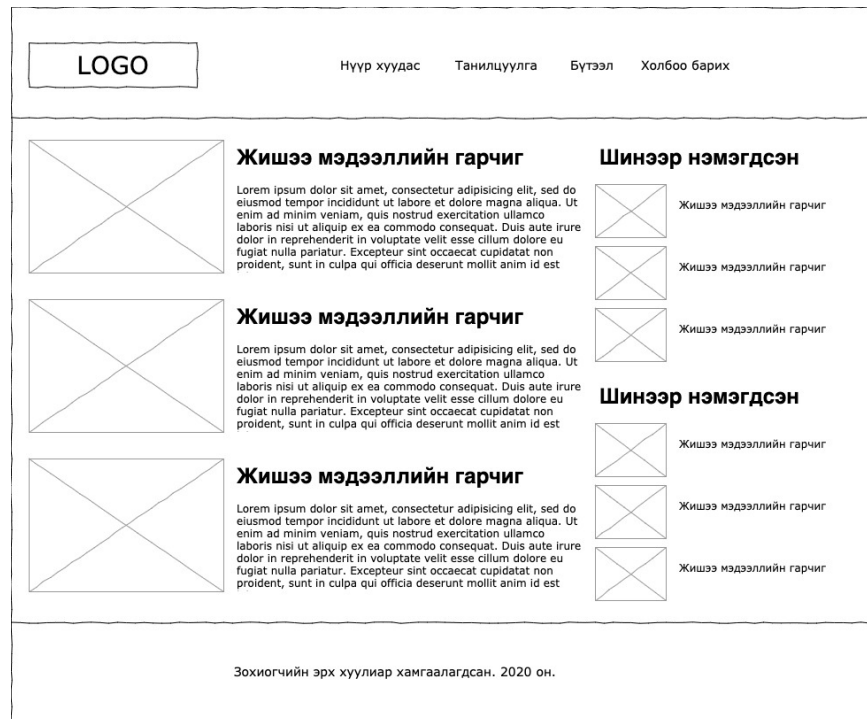


## Лабораторийн ажил 5. Танилцуулга веб сайтын нүүр хуудас

### Даалгавар

Энэ удаагийн лабораторийн ажлаар бид 5 дүгээр лекц дээрээс олж авсан CSS тухай мэдлэгээ ашиглан блог веб сайтын нүүр хуудсыг хийнэ. Хуудасны бүтэц дараах байдалтай байх ба хүссэн зураг, мэдээллээ ашиглаж болно. Үүнд:



### Даалгавар№1

DIV элемент ашиглан веб хуудсанд **дээрхи жишээ загвар** /Үндсэн хэсэг өөрийн шийдэл байж болно/ шиг байрлуулж загварчлах.

- Толгой хэсэг, Үндсэн мэдээллийн хэсэг, Баруун талын хэсэг, Хөл хэсэг хэсэгтэй байна
- Div элементийг тохируулж зэрэгцүүлж нэг мөрөнд дүрсэлнэ. /үндсэн, баргуун/

### Даалгавар№2

Веб хуудсан ашигласан CSS үүдийг **External-гадаад, Internal-дотоод, Inline-мөр** зарлалттай болгох.

- Гадаад CSS файлаас дуудаж, ашиглах.
- Дотоод зарлалт хийх
- Мөр дээр зарлалт хийх

### Үнэлгээ/Ашиглах

1. Хэвшүүлэх ТАГ болон DIV
2. DIV ба Flex ашиглах

лабораторийн цаг дээрээ шалгуулна

### Нэмэлт материал

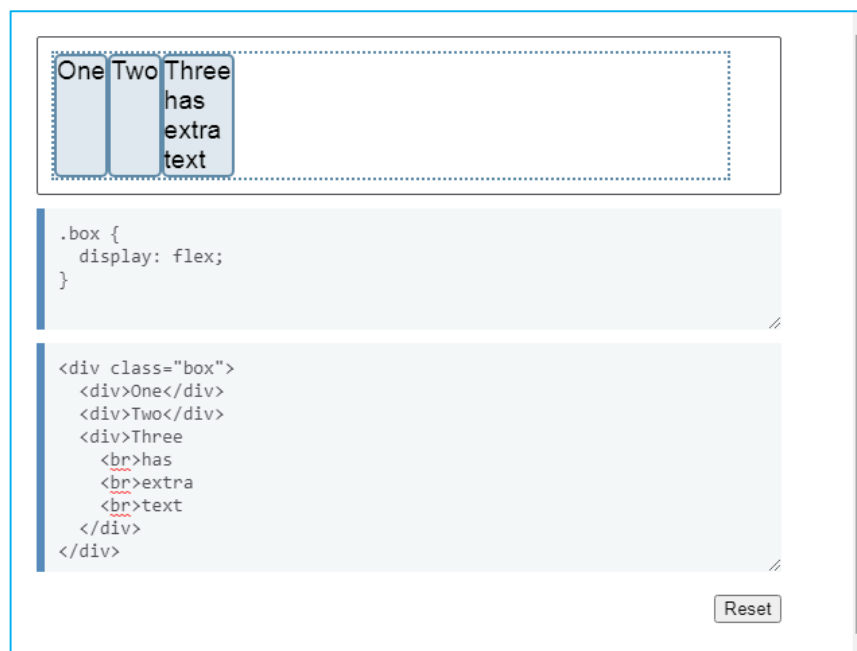
The flex container

Дэлгэрэнгүйг:

[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_flexible\\_box\\_layout/Basic\\_concepts\\_of\\_flexbox](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_flexible_box_layout/Basic_concepts_of_flexbox)

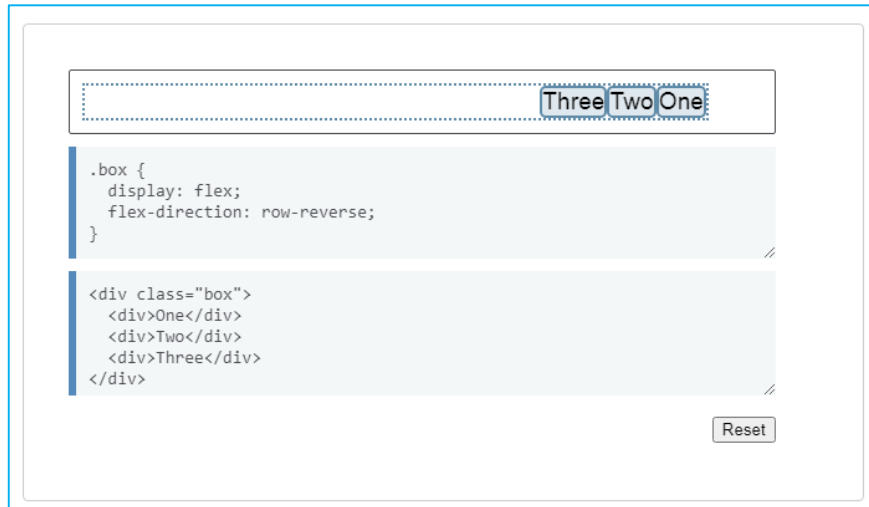
An area of a document that is laid out using flexbox is called a flex container. To create a flex container, set the area's display property to flex. When we do this, the direct children of that container become flex items. You can explicitly control whether the container itself is displayed inline or in block formatting context using inline flex or inline-flex for inline flex containers or block flex or flex for block level flex containers.

- Items display in a row (the flex-direction property's default is row).
- The items start from the start edge of the main axis.
- The items do not stretch on the main dimension but can shrink.
- The items will stretch to fill the size of the cross-axis.
- The [flex-basis](#) property is set to auto. This means that, in each case, it will be equal to the flex item [width](#) in horizontal writing mode, and the flex item [height](#) in vertical writing mode. If the corresponding width/height is also set to auto, the flex-basis content value is used instead.
- The [flex-wrap](#) property is set to nowrap. This means that the flex items will always remain in a single row or column, overflowing their container if their combined width/height exceeds the containing element width/height.



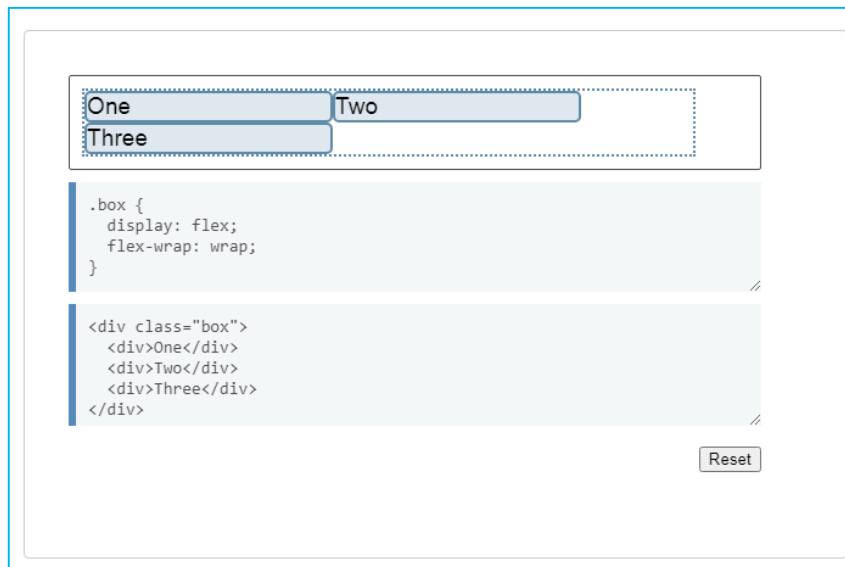
### [Changing flex-direction](#)

Adding the [flex-direction](#) property to the flex container allows us to change the direction in which our flex items display. Setting flex-direction: row-reverse will keep the items displaying along the row, however the start and end lines are switched.



### [Multi-line flex containers with flex-wrap](#)

While flexbox is a one dimensional model, it is possible to cause our flex items to wrap onto multiple lines. In doing so, you should consider each line as a new flex container. Any space distribution will happen across that line, without reference to the lines on either side.



### [The flex-flow shorthand](#)

You can combine the two properties `flex-direction` and `flex-wrap` into the [flex-flow](#) shorthand. The first value specified is `flex-direction` and the second value is `flex-wrap`.

```
.box {  
  display: flex;  
  flex-flow: row wrap;  
}
```

```
<div class="box">  
  <div>One</div>  
  <div>Two</div>  
  <div>Three</div>  
</div>
```

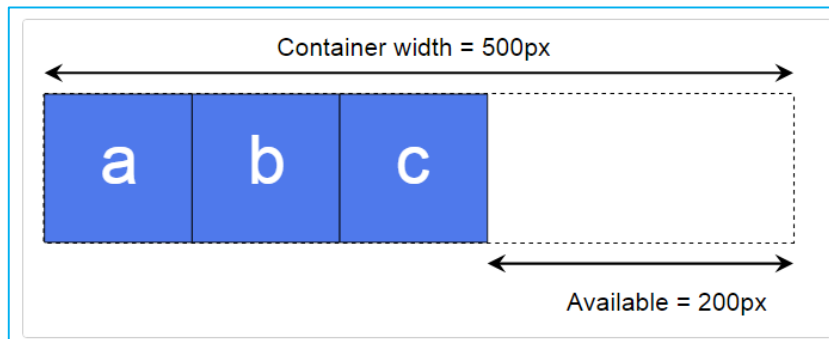
Reset

### Properties applied to flex items

To have more control over flex items we can target them directly. We do this by way of three properties:

- [flex-grow](#)
- [flex-shrink](#)
- [flex-basis](#)

If we have three 100 pixel-wide items in a container which is 500 pixels wide, then the space we need to lay out our items is 300 pixels. This leaves 200 pixels of available space. If we don't change the initial values then flexbox will put that space after the last item.



OneTwoThree

```
.box {
  display: flex;
}

.one {
  flex: 1 1 auto;
}

.two {
  flex: 1 1 auto;
}

.three {
  flex: 1 1 auto;
}

<div class="box">
  <div class="one">One</div>
  <div class="two">Two</div>
  <div class="three">Three</div>
</div>
```

Reset

OneTwoThree

```
.box {
  display: flex;
}

.one {
  flex: 1;
}

.two {
  flex: 1;
}

.three {
  flex: 1;
}

<div class="box">
  <div class="one">One</div>
  <div class="two">Two</div>
  <div class="three">Three</div>
</div>
```


Reset



```
.box {  
  display: flex;  
  align-items: flex-start;  
}
```

```
<div class="box">  
  <div>One</div>  
  <div>Two</div>  
  <div>Three  
    <br>has  
    <br>extra  
    <br>text  
  </div>  
</div>
```

Reset



```
.box {  
  display: flex;  
  justify-content: flex-start;  
}
```

```
<div class="box">  
  <div>One</div>  
  <div>Two</div>  
  <div>Three</div>  
</div>
```

Reset

- **HTML**
- **CSS**

CSS нь HTML элементийн шинж чанар, үүргийг өөрчилж хэлбэржүүлдэг. CSS-ийн ерөнхий бүтэц нь selector, property болон утга гэсэн 3 хэсгээс бүрдэнэ.

Selector {property: value}

*selector* нь HTML элементийн стандарт нэр байна. Харин *property* нь тухайн өөрчлөхийг хүсэж байгаа элемент буюу тагийн атрибут юм. Утга бол атрибутадаа хамааралтай хэсэг юм.

Жишээ нь: **body {color: black}**

Дээрх жишээний хувьд веб хуудасны дэвсгэр өнгө нь хар өнгөтэй болж харагдана. Хэрэв утга нь 2 болон түүнээс олон үгээр илэрхийлэгдэж байвал “” тэмдэглэгээг ашиглах ёстой.

Жишээ нь: **p {font-family: "sans serif"}**

Хэрэв нэг элемент дээр олон *property* тодорхойлж өгөх болвол **p{text-align:center;color:red}** хэлбэрээр зарлаж өгнө.

## Гадаад CSS /external css/

CSS нь гадаад, дотоод, мөн HTML элементэнд агуулагдах CSS гэсэн 3 янз байна. Гадаад CSS-ийг ашиглахдаа тусад нь файл болгон хадгалж тухайн файлыг HTML- ийн *head* хэсэгт дуудаж өгөх замаар ашиглана.

Жишээ нь:

*mystyle.css* файл

```
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/back40.gif") }
```

Дээрх *mystyle.css* файлыг веб хуудсанд дуудахдаа

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

## Дотоод CSS /internal css/

```
<head>
<style type="text/css">
  hr {color: sienna}
  p {margin-left: 20px}
  body {background-image: url("images/back40.gif") }
</style>
</head>
```

## Элемент дотор агуулагдах CSS /inline css/

Элемент дотор CSS бичихдээ дурын элементэд style атрибутыг ашиглаж бичнэ.

```
<p style="color: sienna; margin-left: 20px">
This is a paragraph
</p>
```

Нэг элементэд олон style тодорхойлж өгөх.

```
h3
{
    color: red;
    text-align: left;
    font-size: 8pt
}
```

Олон элементэд style нэмэх

Ингэхдээ групп үүсгэх замаар хэлбэржүүлнэ.

```
h1, h2, h3, h4, h5, h6
{
    color: green
}
```

Класс ашиглах

Класс ашиглах нэг элементийг хэдэн янзаар ч хэлбэржүүлж болдог.

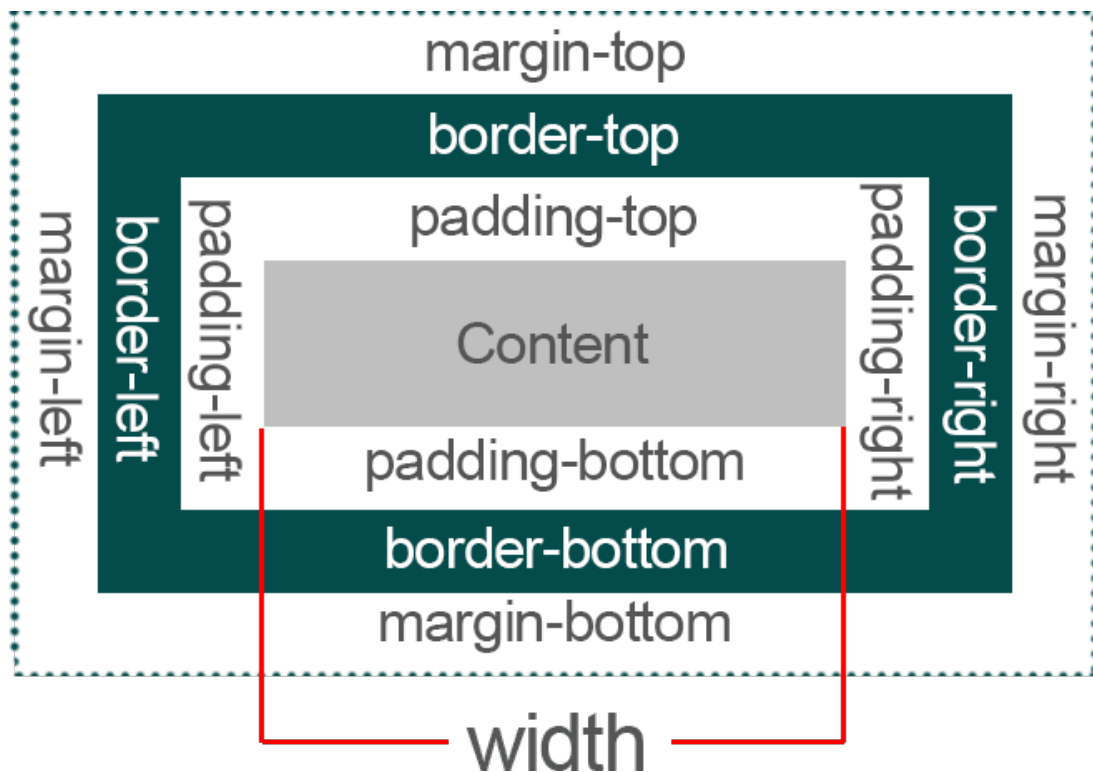
```
p.right {text-align: right}
p.center {text-align: center}
```

Дээрх жишээнд нэг элементийг 2 өөр хэлбэрээр тодорхойлж өгсөн ба түүнийг ашиглахдаа дараах хэлбэртэйгээр ашиглана.

```
<p class="right">
This paragraph will be right-aligned.
</p>
<p class="center">
This paragraph will be center-aligned.
</p>
```

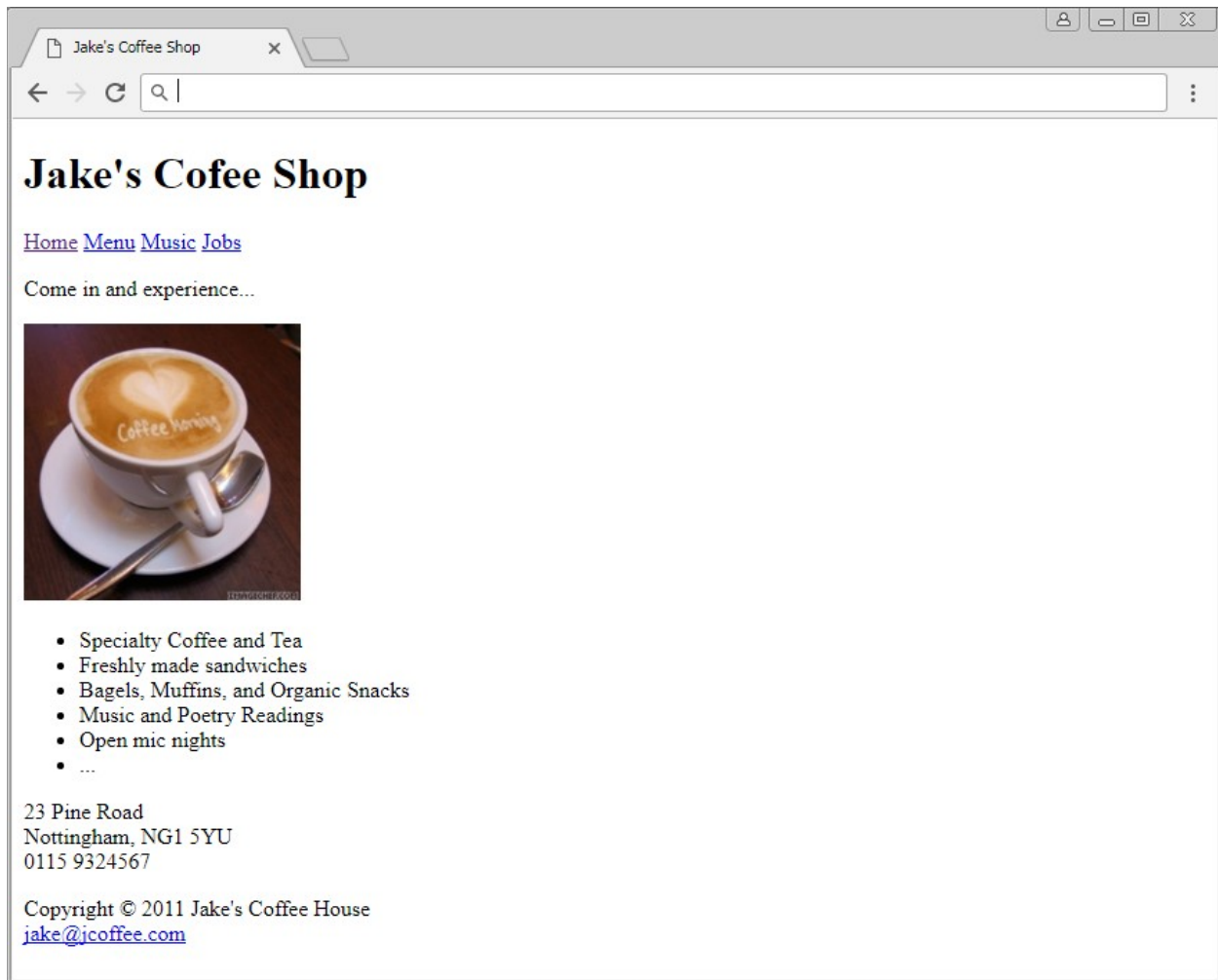


## Padding



## Даалгавар

Jack's coffee shop-ийн вебсайтын нүүр хуудсыг үүсгэх Exercise3.html файл нь өгөгдсөн ба дараах зүйлсийг дэлгэцэнд дүрсэлнэ.



## Зураг 1

### Exercise3.html файл

```
<html>
<head>
  <title>Jake's Coffee Shop</title>
</head>
<body>
  <h1> Jake's Cofee Shop </h1>

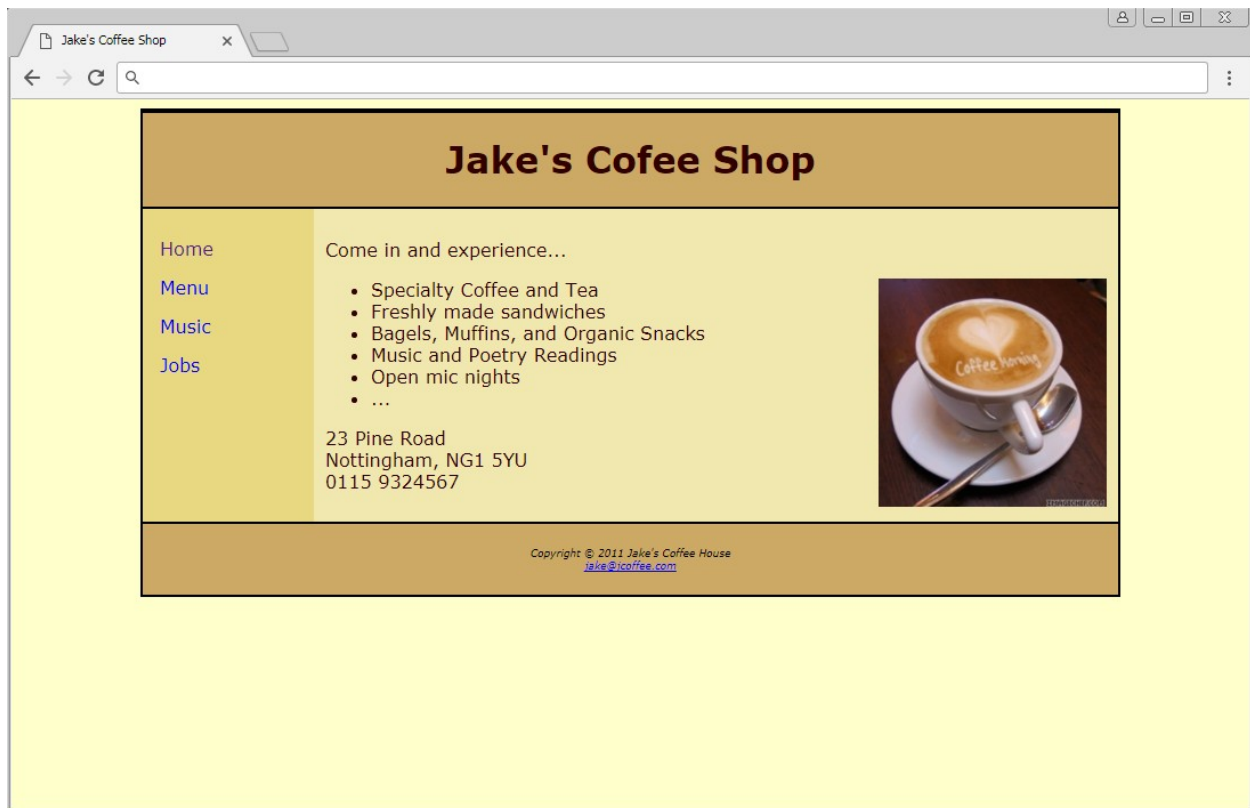
  <a href="home.html">Home</a>
  <a href="menu.html">Menu</a>
  <a href="music.html">Music</a>
  <a href="jobs.html">Jobs</a>
  <p>Come in and experience...</p>
  
  <ul>
```

```

<li>Specialty Coffee and Tea</li>
<li>Freshly made sandwiches</li>
<li>Bagels, Muffins, and Organic Snacks</li>
<li>Music and Poetry Readings</li>
<li>Open mic nights</li>
<li>... </li>
</ul>
<p>23 Pine Road<br> Nottingham, NG1 5YU <br> 0115 9324567</p>
<p> Copyright © 2011 Jake's Coffee House<br>
<a href="mailto:jake@jcoffee.com">jake@jcoffee.com</a></p>
</body>
</html>

```

Зураг 2-т үзүүлсэн шинэ веб хуудсыг дүрслэхээр өөрчлөх ба тохирох CSS-г бичнэ үү. Шинэ веб хуудас нь header буюу толгой хэсэг, 2 багана (нэг нь цэс, нөгөө нь үндсэн агуулгыг гаргах), мөн footer буюу хөлний хэсгүүдийг агуулна.



Зураг 2

Дараах өнгөнүүд хэрэглэгдэнэ. Үүнд:

- *Text: #330000*
- *Body background: #ffffcc*

- Header and footer background: #c0a060
- Left (navigation) column background: #E8D882
- Right (main content) column background: #f1e8b0
- Текст нь Verdana фонтоор дүрслэгдэнэ.

Дараах external CSS файлыг үүсгэнэ үү.

### mystyle.css файл

```
body { //дэвсгэр өнгө болон Verdana, Arial, sans-serif фонтыг
                                             тодорхойлж өг.
}
```

### #container

```
{
    margin: auto;
    width:80%;
    background-color:.....; //өнгийг тодорхойл
    border: 4px double #000000;
} //body таг дотор бичигдэж буй бүх агуулгыг агуулах
хэсгийн хэлбэржүүлэлт
```

```
#header { border-bottom: 2px double #000000;
          border-top: 2px solid #000000;
          background-color: .....; //өнгийг тодорхойл
          text-align: .....; //төхсвийн зэрэгцүүлэлт нь
          голдоо байна.
        }
```

```
#leftcolumn { float:left;
               padding-top:10px;
            }
```

```
.navBar { text-decoration:none;
          margin: 15px;
          display:block;
        } //Home, Menu, Music, Jobs холбоосуудад хэрэгжих хэлбэржүүлэлт
```

```
#rightcolumn { // зүүн талаасаа авах margin хэмжээг 150px гэж
               тодорхойл
               background-color: .....; //өнгийг тодорхойл
               padding:10px;
            }
```

```
.floatright { padding-right:20px;
               float:right;
            } //зургийг баруун талруу хөвөлттэй гаргах
```

**#footer {**

- Дээрээсээ авах хүрээг 2px, double, хар өнгөтэй тодорхойл.
  - Padding хэмжээг (top, right, bottom, left) 20px, 0, 20px, 0
  - Арын дэвсгэр өнгө
  - Текстийн өнгө (хар)
  - Фонтны хэмжээ (xx-small)
  - Фонтны стайл (italic)
  - Текстийн зэрэгцүүлэлтийг (center) тус тус тодорхойл.
- ```
}
```

```
h1 {    background-color:#663300
        //дээр болон доороос авах padding хэмжээг тус тус
        10px
        гэж тодорхойл.
    }
```

## Styling Links with Pseudo-Classes

Home, Menu, Music, Jobs холбоосуудын төлвийг өөрчлөх CSS бичнэүү. (unvisited, visited, mouse over, selected link)