

# WEB 008-gurux    Madrimov Shoxruxbek    Variant 21

Билет №21	Мухаммад ал-Хоразмий номидаги Тошкент ахборот технологиялари университети “Мультимедиа технологиялари кафедраси” “Web дастурлашга кириш” фанидан Якуний назорат саволлари	Тасдиқлайман “МТ” кафедраси муdiri _____ Э.Ш.Назирова
1. HTML5 да формалар учун янги атрибутлари. Изоҳланг ва мисоллар келтиринг. 2. CSS селекторлар турлари: комбинациялашган селекторлар. Изоҳланг ва мисоллар келтиринг. 3. PHP файллар билан ишлаш. Изоҳланг ва мисоллар келтиринг.		

## 21-variant

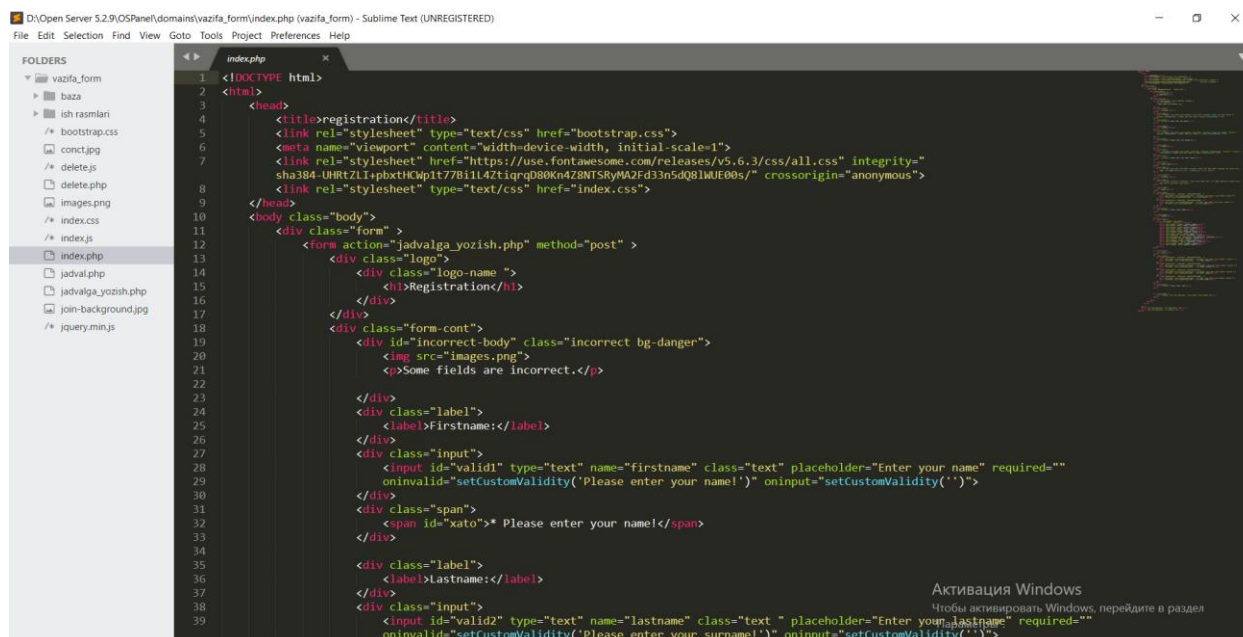
1. HTML5 da formalar uchun yangi atributlari. Izoxlang va misollar keltiring.
2. CSS selektorlar turlari: kombinatsiyalashgan selektorlar. Izoxlang va misollar keltiring.
3. PHP fayllar bilan ishlash. Izoxlang va misollar keltiring.

### 1. HTML5 da formalar uchun yangi atributlari. Izoxlang va misollar keltiring.

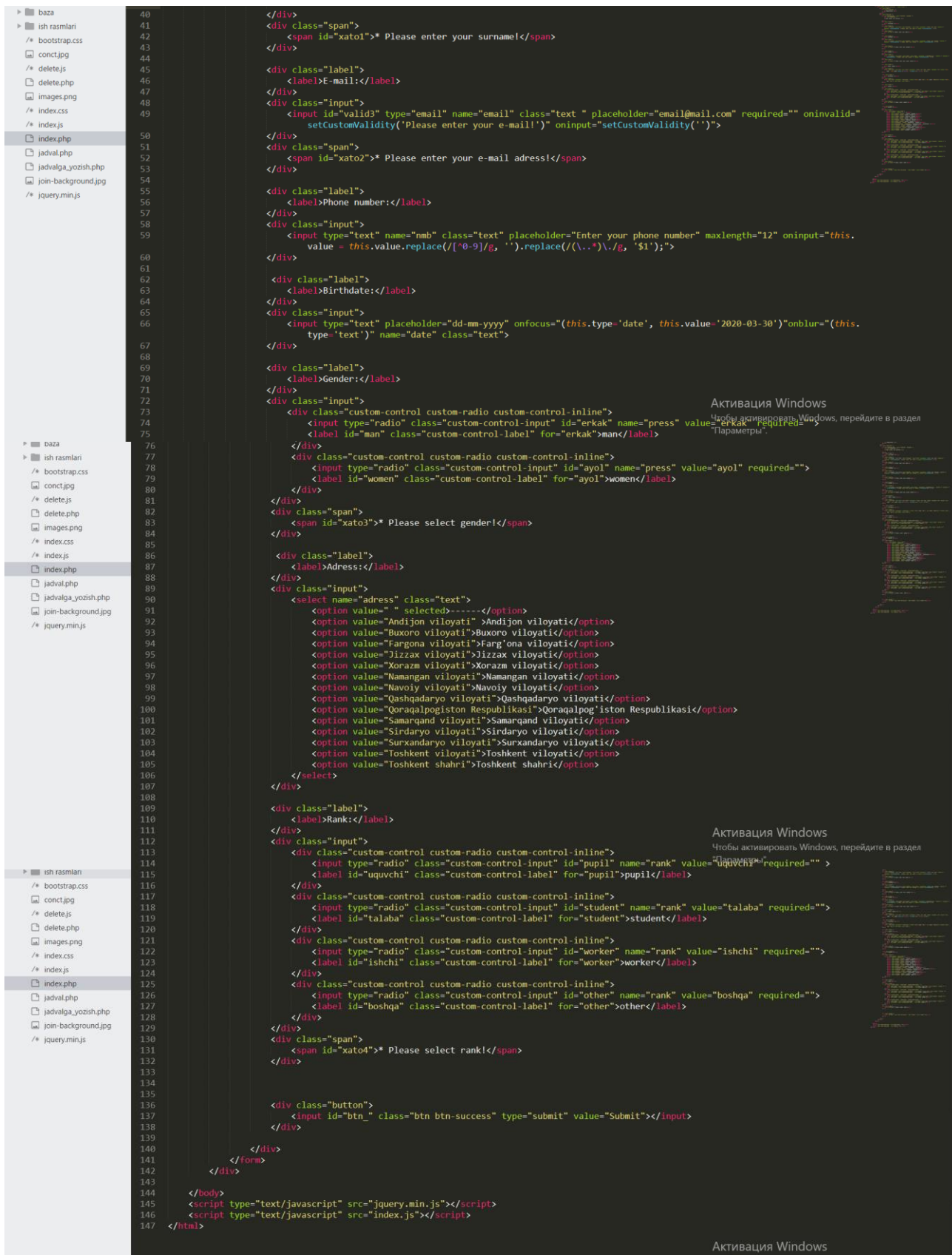
Web saytlarda formalar juda ko‘p joyda ishlatiladi. Misol uchun, **registratsiyada, so‘rovlarda, avtorizatsiyada, kommentariyalarda** va hakazo. Demak, zamonaviy saytlarni formalarsiz tasavvur qilib bo‘lmas ekan. Maqolamiz aynan shu formalarga bag‘ishlanadi, ya’ni formaga oid teglarga. Html orqali faqat formani tashqi ko‘rinishini qilish mumkin, undagi ma’lumotlarni qayta ishlash veb dasturlash( **php, asp, ...**) orqali amalga oshiriladi.

Formaga tegishli teglar ko‘proq bo‘lgani va ularni tushunish yengilroq bo‘lishi uchun maqolani ikki qismga bo‘ldik.

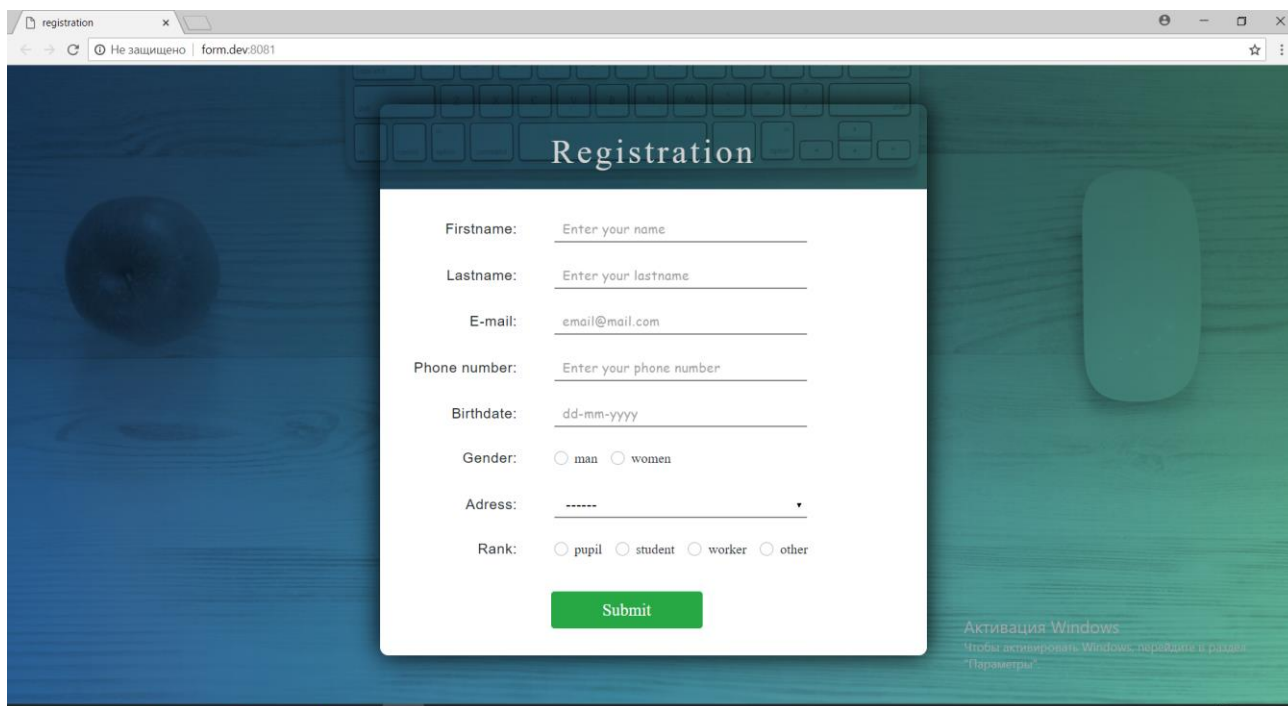
Demak, dastlabki maqola uchun kodlarni keltiraman.



The screenshot shows a web browser window with a registration form. The form has two input fields for 'Firstname' and 'Lastname'. Below the 'Firstname' field, there is a red error message: 'Some fields are incorrect.' The 'Lastname' field is also present. The form is styled with a dark background and red text for error messages. The browser's address bar shows the URL 'http://localhost:8080/vazifa\_form/index.php'. The page title is 'registration'.



Natija:



Bu kodlarda quyidagi notanish teglar ishlatilgan:

1. **<form>...</form>** — veb sahifaga forma joylashtirish, forma foydalanuvchi va veb server bilan aloqani ta'minlab beradi. Agar forma elementlari( **button, checkbox, radio, button, ...**) kiritilmasa, bu teg brauzerga hech nima chiqarmaydi, uning attributlari:

**accept-charset** — serverga jo'natilayotgan axborot kodirovkasini o'rnatish.

`<form accept-charset = "windows-1251">...</form>`

**action** — forma orqali jo'natilayotgan ma'lumotlar qaysi sahifa orqali qayta ishlanishi kerakligi beriladi, biror adres ko'rsatiladi.

`<form action = "index.php">...</form>`

formadagi barcha axborotlar «index.php» sahifasiga olib o'tilib, u yerda tahrirlanadi.

**autocomplete** — formani avtomat to'ldirishni boshqarish, brauzer keshida qandaydir ma'lumotlar bo'lsa, shular orqali forma to'latilishi mumkin, uning qiymati: **on**(yoqig'), **off**(o'chiq).

**enctype** — jo'natilayotgan axborotlarni kodlash, uning qiymatlari:

\* `application/x-www-form-urlencoded` — bo'sh joylar o'rniga «+» belgisini qo'yadi va qolgan yozuvlarni, uning 16lik tizimidagi qiymatlari bilan to'ldiradi;

\* `multipart/form-data` — axborotlar kodlanmaydi, fayllarni jo'natishda ishlatiladi;

\* `text/plain` — bo'sh joylar «+» belgisi bilan yozildai, yozuvlar kodlanmaydi.

**method** — axborotlarni jo‘natish turi. Bu haqida «[HTML da GET va POST](#)» maqolada to‘liq berilgan.

**name** — formaga nom berish, bu nom skriptlar yozishda ishlatiladi.

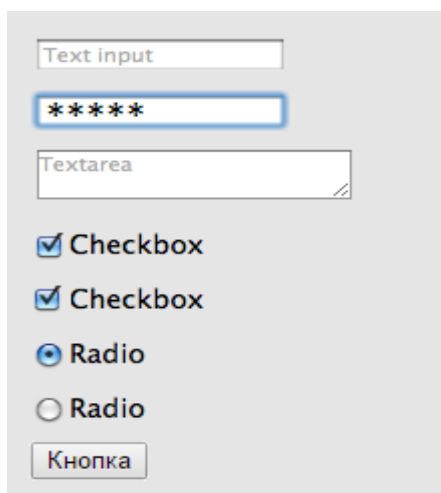
**novalidate** — formaga kiritilgan ma’lumotlarni brouzer tomonidan tekshirish. Brouzer pochta adreslarni, url adreslarni tekshirishi mumkin.

**target** — jo‘natilayotgan axborotni qayta ishlovchi veb sahifani ochilish turi, uning qiymatlari: `_blank`, `_self`, `_parent`, `_top`. «a» tegidagi «target» attributi bilan bir hil, shuning uchun qayta yozib o‘tirmadim.

2. `<input>...</input>` — bu teg orqali forma elementlarini yaratish mumkin, ya’ni formani asosi hisoblanadi. Bu elementlarni yuqoridagi kodda keltirdim va keyingi maqoladagi kodda ham keltiraman, lekin faqat shu yerda tushuntirib o‘taman. Bu tegning attributlari:

**type** — forma elementi turi, bu atribut formada har xil ko‘rinishdagi elementlar hosil qilib beradi, ularni ba’zilarini misollarda ko‘rsatdim, ularning qiymatlari quyidagilar:

- \* `button` — tugma(knopka) hosil qilish;
- \* `checkbox` — bir necha tanlovni amalga oshirib beruvchi bayroqcha hosil qilish;
- \* `file` — serverga jo‘natiladigan fayl adresini kiritish, fayl elementi;
- \* `hidden` — ko‘rinmas element, sahifada ko‘rinmaydi, lekin ko‘p hollarda ishlatiladi;
- \* `image` — rasmi qator, rasmga bosilganda, axborotlar serverga jo‘natiladi;
- \* `password` — oddiy matnli element, lekin yozuvlari ko‘rinmaydi, parol yozish uchun;
- \* `radio` — bir necha tanlovalardan faqat bittasini tanlash imkonini beruvchi element;
- \* `reset` — forma qiymatlarini dastlabki holatga qaytarish;
- \* `submit` — formadagidagi axborotlarni serverga jo‘natish;
- \* `text` — oddiy matn kiritish elementi.



The image shows a collection of HTML form elements on a light gray background. At the top is a single-line text input field with the placeholder text 'Text input'. Below it is a password input field, also single-line, with five asterisks '\*\*\*\*\*' and a blue border. Next is a multi-line text area with the placeholder text 'Textarea' and a small icon in the bottom right corner. Below the text area are two checked checkboxes, each with a blue checkmark and the label 'Checkbox'. Then there are two radio buttons; the first is selected with a blue dot and labeled 'Radio', while the second is unselected and also labeled 'Radio'. At the bottom is a button with the Russian word 'Кнопка' (Knopka) written on it.

**align** — tip rasm(image) bo‘lganda, rasmni tekislash, qiymatlari: bottom, left, middle, right, top.

**alt** — rasmlı tugmada, rasm ochilmasa, uning o‘rnida paydo bo‘ladigan yozuv.

**autocomplete** — elementni avtomat to‘ldirishni o‘rnatish.

**border** — rasm atrofidagi ramka qalinligini o‘rnatish.

**checked** —checkbox yoki radio uchun dastlabki holatni «belgilash» ko‘rinishida namoyon qilish.

```
<input type = "checkbox"/>Messi
```

```
<input type = "checkbox" checked = "checked"/>Rooney
```

```
<input type = "checkbox"/>Neymar
```

☐ Messi ☒ Rooney ☐ Neymar

**disabled** — elementni blokirovka qilish, ya’ni dastlabki holatni o‘zgartirib bo‘lmaydi.

```
<input type = "text" disabled value = "Yozolmaysan"/>
```

**form** — boshqa forma bilan identifikator orqali bog‘lanish.

**formaction** — axborotlarni qayta ishlaydigan sahifa adresini berish(form tegi action attributi bilan bir hil).

**formmethod** — axborotlari qaysi metod orqali serverga jo‘natishni belgilash(Get, Post. Form tegi method attributi bilan bir hil).

**maxlength** — foydalanuvchi tomonidan maksimal kiritilishi mumkin bo‘lgan axborot o‘lchami(kiritilgan raqamgacha bo‘lgan simvollar).

**multiple** — element tipi fayl bo‘lganda, bir paytda bir necha fayllar kiritilishi mumkinligini bildiradi.

**name** — forma elementiga takrorlanmas nom berish, bu nom axborot serverga jo‘natilayotganda ishlatiladi.

**readonly** — yozuvli elementda ishlatilsa, faqat yozuvni ko‘rish imkonini yaratadi, o‘zgartirib bo‘lmaydi.

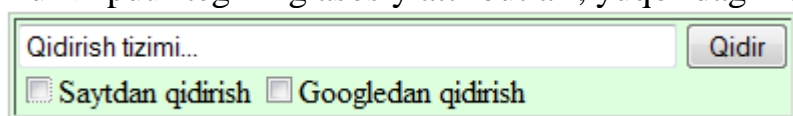
**size** — yozuvli elementlarda(text, password) yozuv yozish mumkin bo‘lgan joyning uzunligini o‘rnatish.

**src** — tip rasm(type = image) bo‘lgan elementni adresini ko‘rsatish.

**tabindex** — klaviaturadan nechi marotaba «tab» tugmasi bosilsa, shu elementga o‘tish kerakligini o‘rnatish. Qiymat raqamlardan iborat bo‘ladi.

**value** — forma elementiga dastlabki qiymatni berish.

Bu <input> tegining asosiy atributlari, yuqoridagi kodning natijasi quyidagicha bo‘ladi:

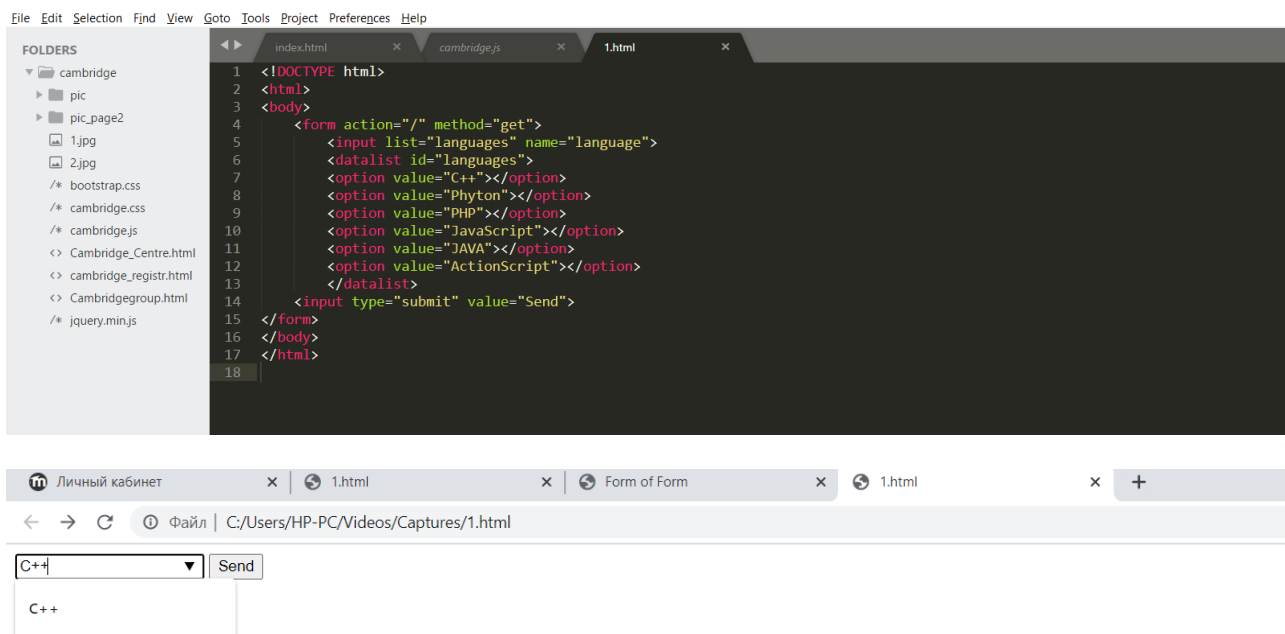
The image shows a web form with a light green background. At the top, there is a text input field with the placeholder text "Qidirish tizimi..." and a button labeled "Qidir" to its right. Below the input field, there are two radio buttons. The first one is selected and is followed by the text "Saytdan qidirish". The second one is not selected and is followed by the text "Googledan qidirish".



Natija qoniqarli, lekin bu forma hech qanday vazifa bajarmaydi, faqat tashqi ko‘rinishni chizib beradi.

## HTML5ning yangi forma elementlari

Teg	Izohi
<datalist>	Kiritishni boshqarish uchun oldindan kiritilgan parametrlar ro'yhatini belgilaydi



Teg	Izoh
<output>	Hisobalshnatijasini aniqlashtegi

	Qiymati	Izohi
For	element_id	Hisoblash natijasi va hisoblashda ishlatiladigan elementlar o'rtasidagi munosabatni belgilaydi



## 2. CSS selektorlar turlari: kombinatsiyalashgan selektorlar. Izoxlang va misollar keltiring.

Biz CSS da selektorlarni beshta turi mavjud, bular:

- Oddiy selektorlar;
- Kombinatsiyalashgan selektorlar;
- Psevdo-elementlar selektori;
- Psevdo-siniflar selektori;
- Atributlar selektori;

### Kombinatsiyalashgan selektorlar

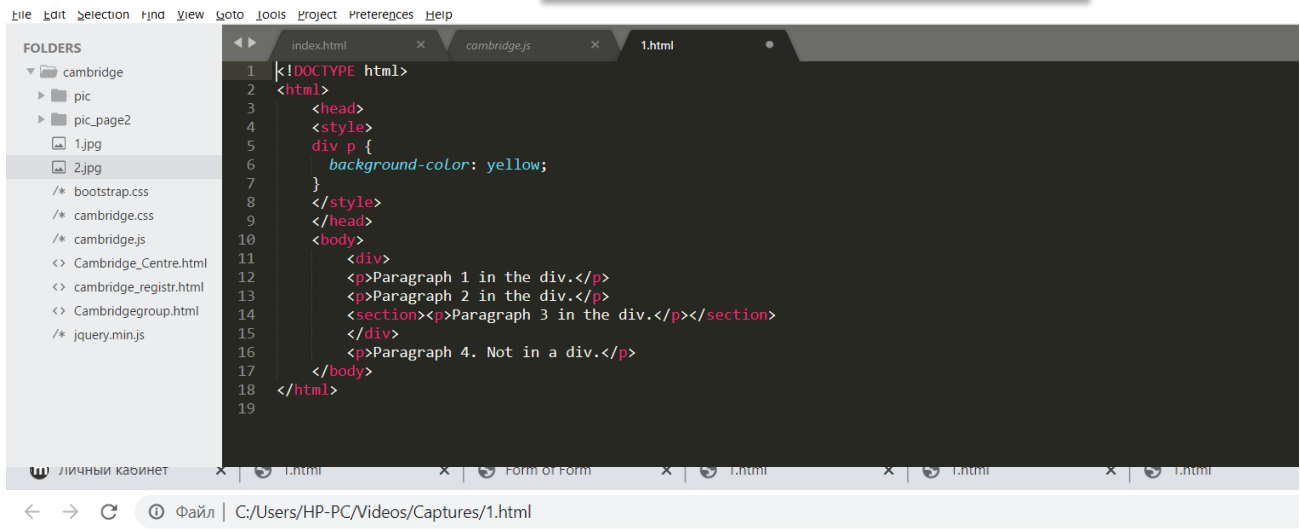
Kombinatsiyalashgan selektorlarda HTML teglarga quydagilardan biri orqali murojaat qilinib, CSS xususiyatlari o'rnatiladi:

Selektor turi	Misol	Izoh
element element	div p	<b>Avlod selektori</b> <div> teg ichidagi barcha <p>teglarini tanlash
element>element	div > p	<b>Farzand (Child) selektori</b> Birinci avlod tegi<div> bo'lgan barcha <p> teglarini tanlash
element+element	div + p	<b>Qo'shni selektorlar</b> <div> tegidan keyin joylashgan barcha <p> teglarni tanlash
element1~element2	p ~ ul	<b>Umumiy qo'shni selektorlar</b> <p> tegidan keyin joylashgan barcha <ul> teglarini tanlash



## CSS element element selektori

```
element element {  
    CSS qoidalar;  
}
```



Paragraph 1 in the div.

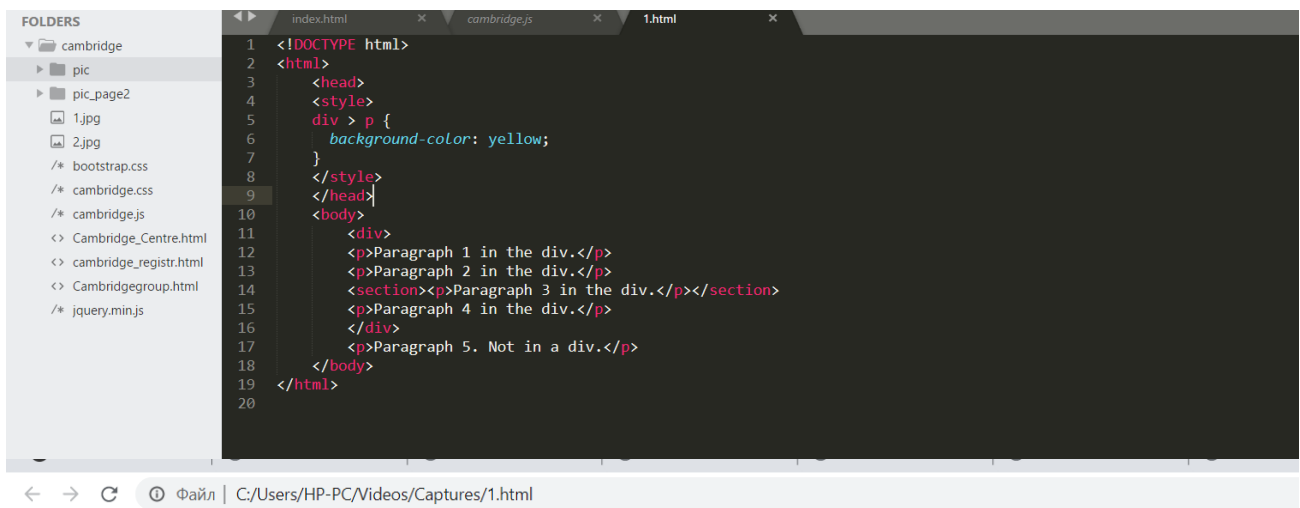
Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

## CSS element>element selektori

```
element > element {  
    CSS qoidalar;  
}
```



Paragraph 1 in the div.

Paragraph 2 in the div.

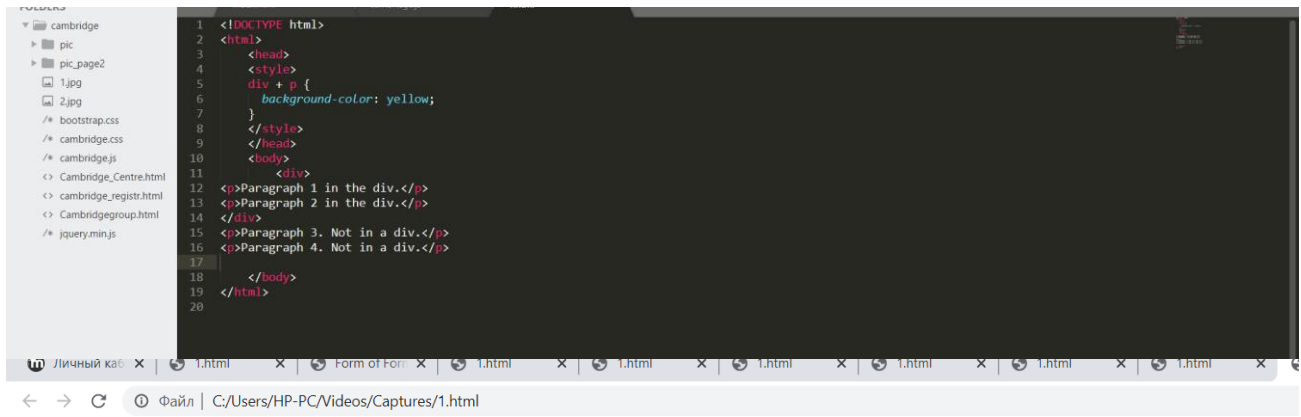
Paragraph 3 in the div.

Paragraph 4 in the div.

Paragraph 5. Not in a div.

```
element + element {  
    CSS qoidalar;  
}
```

## CSS element+element selektori



Paragraph 1 in the div.

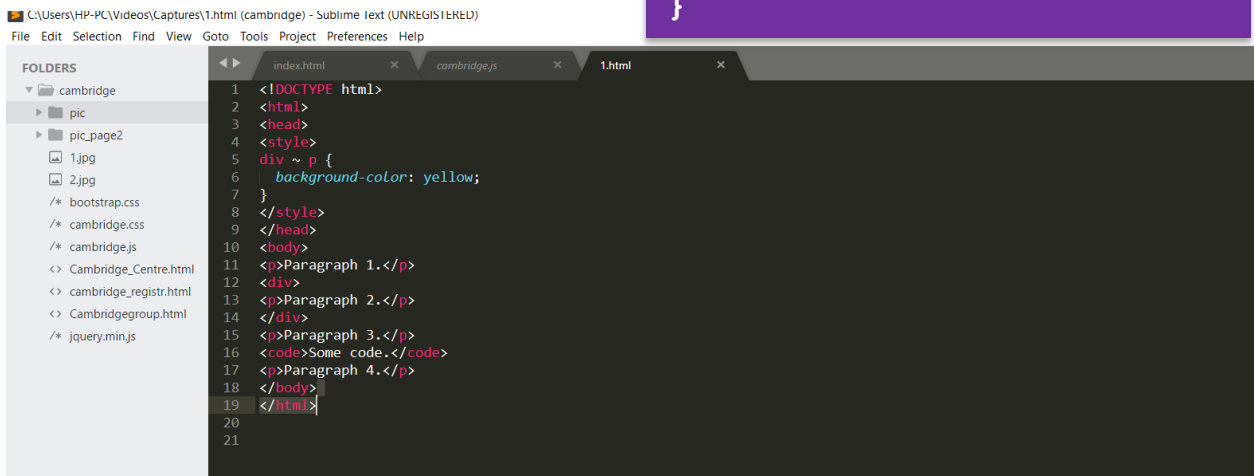
Paragraph 2 in the div.

Paragraph 3. Not in a div.

Paragraph 4. Not in a div.

```
element ~ element {  
    CSS qoidalar;  
}
```

## CSS element1~element2 selektori



Paragraph 1.

Paragraph 2.

Paragraph 3.

Some code.

Paragraph 4.

### 3. PHP fayllar bilan ishlash. Izoxlang va misollar keltiring.

PHP fayllar va kataloglarni boshqarish uchun ko'plab ichki funksiyalarni o'z ichiga oladi. Siz ushbu funksiyalardan foydalanib, fayllarni o'qish, yozish, o'chirish va ma'lumotlarni olishingiz mumkin.

#### PHP dan foydalanib faylni ochish.

PHP faylni ochish *fopen()* funksiyasi orqali amalga oshiriladi. Quyida sintaksisni ko'rishingiz mumkin:

`fopen(fayl nomi, rejim)`

Parametr ta'rifi.

fayl nomi: ochiladigan fayl nomini bildiradi. Bu talab qilinadigan parametr.

rejim: fayl qaysi rejimda ochilishi kerakligini bildiradi. Bu talab qilinadigan parametr.

#### Faylni ochish rejimlari:

Rejimlar	Ta'rifi
<b>r</b>	<b>Faqat o'qish uchun.</b> Fayl ko'rsatgichi fayl boshidan boshlanadi.
<b>w</b>	<b>Faqat yozish uchun.</b> Fayl ichidagilarni o'chiradi yoki yo'q bo'lsa yangi fayl hosil qiladi. Fayl ko'rsatgichlari fayl boshidan boshlanadi.
<b>a</b>	<b>Faqat yozish uchun.</b> Fayldagi ma'lumotlar saqlanadi. Faylko'rsatgichlari faylning oxiridan boshlanadi. Agar fayl mavjud bo'lmasa, yangi fayl hosil qiladi.
<b>x</b>	<b>Yozish uchun yangi fayl yaratadi.</b> Agar fayl allaqachon mavjud bo'lsa, FALSE qaytaradi.
<b>r+</b>	<b>O'qish/yozish uchun.</b> Fayl ko'rsatgichi fayl boshidan boshlanadi.
<b>w+</b>	<b>O'qish/yozish uchun.</b> Fayl ichidagilarni o'chiradi yoki yo'q bo'lsa yangi fayl hosil qiladi. Fayl ko'rsatgichlari fayl boshidan boshlanadi.
<b>a+</b>	<b>O'qish/yozish uchun.</b> Fayldagi ma'lumotlar saqlanadi. Faylko'rsatgichlari faylning oxiridan boshlanadi. Agar fayl mavjud bo'lmasa, yangi fayl hosil qiladi.
<b>x+</b>	<b>O'qish/yozish uchun yangi fayl yaratadi.</b> Agar fayl allaqachon mavjud bo'lsa, FALSE qaytaradi.

**Faylni yopish.**PHP faylni yopish uchun **fclose()** funksiyasidan foydalanamiz.

Namuna:

```
<?php
    $fayl=fopen("faylim.txt" , " r ");
    // bu yerda biron bir kod
    fclose($fayl);
?>
```

### **Faylni oxirini tekshirish.**

Har qanday faylni (EOF=end of any file) oxirigacha o'qish uchun PHP feof() funksiyasini taqdim etadi, feof() funksiyasidan foydalanib fayldagi barcha ma'lumotlarni siklga qo'yib, oxirigacha o'qish mumkin.

Sintaksis:

**feof(\$fayl);**

### **Fayllarni qatorma-qator o'qish**

PHPda fayllarni qatorma-qator o'qish uchun fgets() funksiyasi mavjud. Faylda bitta qatorni o'qish uchun fgets () funktsiyasi ishlatiladi. Bu funktsiya faylni linebyline(qatorma-qator) o'qiydi, bu satrlarni o'qib bo'lgach, fayl ko'rsatgichi avtomatik tarzda faylning keyingi satriga o'tadi.

Sintaksisi

**fgets(\$fayl);**

```
<?php
    $fayl=fopen("faylim.txt","r");
    while(!feof($fayl))
    {
        echo fgets($fayl);
        echo "<br>";
    }
    fclose($fayl);
?>
```

### **Faylni belgima-belgi o'qish.**

Fayllarni belgima-belgi o'qish uchun PHPda fgetc() funksiyasi mavjud. Faylda bitta belgini o'qish uchun fgetc() funktsiyasi ishlatiladi. Bu funktsiya character-by-

charatcer(belgi bilan) faylni o'qiydi, bu belgini o'qib bo'lgandan so'ng, fayl ko'rsatgichi avtomatik ravishda faylning keyingi belgilariga o'tadi. Ya'ni qatorma-qator belgilarni bitta satrda o'qiydi.

Sintaksisi:

**fgetc(\$fayl);**

```
<?php
    $fayl=fopen("fayl.txt" , "r");
    while(!feof($fayl))
    {
        echo fgetc($fayl);
    }
    fclose($fayl);
?>
```

### **Faylga ma'lumot kiritish**

.Faylga ma'lumot yozish uchun fwrite() funksiyasi mavjud:

Sintaksisi:

**fwrite(\$fayl,\$malumot);**

Namuna:

```
<?php
    $fayl= fopen('fayl.txt','w');
    fwrite($fayl,"salom");
    fclose($fayl);
    echo "Faylga ma'lumot yozildi.";
?>
```

### **Faylga ma'lumot qo'shish**

```
<?php
    $fayl=fopen("fayl.txt" , "a+");
    fwrite($fayl,"Yangi ma'lumot.");
    rewind($fayl); //rewind() funskiyasi ma'lumotni faylni boshiga o'tkazadi
    echo fread($fayl, filesize("fayl.txt"));
    fclose($fayl);?>
```