

ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ

САМАРҚАНД ФИЛИАЛИ

Ахборот ва педагогик технологиялар факультети

«Информатика ва ахборот технологиялари»
кафедраси

Я.Ёрбеков, Ф.Анорбоева

**Web дастурлаш фани бўйича амалий ва
лаборатория машғулотларини бажариш учун
услубий кўрсатмалар**

Самарқанд-2008

ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ

САМАРҚАНД ФИЛИАЛИ

Ахборот ва педагогик технологиялар факультети

«Информатика ва ахборот технологиялари»

кафедраси

Я. Ёрбеков, Ф.Анорбоева

**Web дастурлаш фани бўйича амалий ва
лаборатория машғулотларини бажариш учун
услубий кўрсатмалар**

Ушбу услубий кўрсатмалар 4-боскич бакалавиятура йўналишларининг IV курсида ўқитиладиган «Web дастурлаш» махсус фани дастури асосида яратилган.

Услубий кўрсатмалар ТАТУ Самарқанд филиали ўқув-услубий кенгашининг 2007 йил, декабр «№4» сон қарори билан нашрга тавсия қилинган.

Самарқанд-2008

Я.Ёрбеков, Ф.Анорбоева. Web дастурлаш фани бўйича амалий ва лаборатория машғулотларини бажариш учун услубий кўрсатмалар. ТАТУ Самарқанд филиали, 2008 й., 32 бет.

Ушбу услубий кўрсатмаларда Web дастурлаш фани бўйича қисқача назарий билимлар ва 9 та амалий машғулотларни ва 2 та лаборатория машғулотларини бажариш учун услубий кўрсатмалар берилган. Хар бир дарс учун сценарийлар яратиш топшириғи ва унинг бажарилиши мисолларда келтирилган. Сценарийларни ёзиш JavaScript тилида бажарилиши кўзда тутилган. JavaScript тили сценарийлар (скриптлар) тили хисобланади ва улар асосан Web саҳифаларда интерактив элементлар яратиш учун ишлатилади. Шу сабабли Web дастурлаш технологияси JavaScript тили ёрдамида ўрганилади.

Тақризчилар: Самарқанд иқтисодиёт ва сервис
институтини доценти,
ф.м.ф.н., В.Ф.Бурнашев.
ТАТУ Самарқанд филиали доценти,
ф.м.ф.н. Н.А.Ражабов.

Самарқанд-2008

М У Н Д А Р И Ж А

1. Кириш.....	5
2. Содда сценарийлар яратиш. HTML хужжатлар яратиш ва уларни форматлаш.....	5
3. Объектлар ва усулларни аниқлаш.....	7
4. JavaScript ходисалари. MouseOver ходисаларни ишловчиси.....	9
5. Ўзгарувчилар билан ишлаш. Фойдаланувчига суровлар.....	11
6. Хужжатнинг объектли модели. Асосий объектлар ва уларнинг хусусиялари.....	13
7. history объекти хусусияти.....	14
8. JavaScript функциялари. Уларни яратиш ва ишлатиш.....	17
9. Функциялар ёрдамида ойналар очиш	18
10. PHP нинг асосий синтаксиси.....	20
11. HTML ёрдамида шахсий web саҳифалар яратиш.....	21
12. Web саҳифалари учун CSS файлини яратиш.....	21
13. JavaScript ёрдамида интерактив Web саҳифаларини яратиш.....	26
14. Web саҳифалари учун кидириш дастурини яратиш.....	26
15. Фойдаланилган адабиётлар	31

Кириш

Web дастурлаш технологиясида JavaScript тили жуда кенг ишлатилади. Сценарийларни ёзиш JavaScript тилида жуда чиройли бажарилиши ва у хозирча ягона сценарийлар тили бўлгани учун жуда оммалашиб кетишига сабаб бўлди. JavaScript тили сценарийлар (скриптлар) тили ҳисобланади ва улар асосан Web саҳифаларда интерактив элементлар яратиш учун ишлатилади. JavaScript тилини жуда кўп браузерлар, шу жумладан қуйидаги браузерлар қўллайди: **Internet Explorer, Firefox, Netscape, Safari, Opera, Camino** ва бошқалар.

JavaScript коди, одатда мижоз Web-браузери томонидан берилади ва бу ҳолда у мижоз томонидан сценарий деб аталади. Аммо **JavaScript** коди **HTML** ҳужжатларини шакллантириш учун Web-сервер да ҳам бажариш мумкин. JavaScript кодини яратишда матн редактори ва Web браузер ишлатилади. Бу ишни бажаришда **HTML** ва **CSS** ларни билиш жуда қўл келади. Агар биз JavaScript даги кўникмаларимизни Web - сайтларда ишлатмоқчи бўлсак, у ҳолда Web-сайт ҳам керак бўлиб қолади. Матн редактори сифатида эса NotePad ни ишлатиш мумкин.

1-амалий машғулот.

Мавзу: Содда сценарийлар яратиш. HTML ҳужжатлар яратиш ва уларни форматлаш.

Бизнинг биринчи сценариймиз JavaScript да уни яратиш ва Web-саҳифага жойлаштириш асосларига бағишланади. Бу машғулотимизда сиз JavaScript да нималар яратиш мумкин ва нималарни яратиш мумкин эмаслиги билан ҳам танишасиз. Бу сценариймиз матнни Web-саҳифага чиқаради. Бизнинг матнимиз қизил рангли бўлади:

Сценарий

<HTML>

<SCRIPT LANGUAGE="javascript">

document.write

("Бу қизил рангли матн ")

</SCRIPT> <?HTML>

Сценарий ишлаши натижаси:

Бу қизил рангли матн

Сценарийни таҳлил қилами:

Биринчи каторда қуйидагилар ёзилган:

<SCRIPT LANGUAGE="JavaScript"> - бу HTML код булиб, браузерга шу ердан JavaScript бошланишини англатади. JavaScript нинг исталган сценарийси шу коддан бошланади. </SCRIPT> коди исталган сценарийни тугаллайди. Демак скрипт

<SCRIPT LANGUAGE="javascript"> дан бошланиб </SCRIPT> билан тугалланади.

DOCUMENT ёрдамида HTML ҳужжати аникланади. Бу ҳужжат узгаритирилиши мумкин:

document.write(" Бу қизил рангли матн ")

Шу таҳлитда сценарий қисмларини ўрганиб чиқиш мумкин. Демак, сценарий қуйидагича ташкил этилган. DOCUMENT ёрдамида HTML ҳужжати эълон қилинади. Бу ҳужжат ўзгартирилиши мумкин. Ёзилиши лозим бўлганлар қавслар ичига олинади.

Топшириқ

Юқоридаги сценарийни шундай ўзгартирингки, натижада матннинг иккита қатори чиқарилиши таъминлансин, қизил ва кўк рангли қаторлар. Бу ишни JavaScript нинг кўшимча буйруқлари ёрдамида амалга ошириш талаб қилинади.

Саҳифага қуйидагилар чиқарилиши лозим:

Бу кизил рангли матн

Бу кўк рангли матн

Бу ишни куйидагича амалга ошириш мумкин:

```
<HTML>
<SCRIPT type="text/javascript">
document.write
("<FONT COLOR='RED'><Бу кизил рангли матн</FONT><BR>")
document.write
("<FONT COLOR='BLUE'> Бу кук рангли матн </FONT>")
</SCRIPT> </HTML>
```

Натижа `document.write` га иккинчи қаторни қўшимча қилиш ва HTML кодига ўзгартиришлар киритилган. Шунингдек `
` буйруги қўшимча қилинган ва қуйидаги натижа олинган:

Бу кизил рангли матн

Бу кўк рангли матн

Хатолар хақида хабарлар

Биз асосан хатолар тўғрисида хабарлар чиққанда нима қилиш лозимлигини кураимиз. Биз JavaScript сценарий ёза бошлашимиз билан бу хабарлар пайдо бўлиб туради. Хатолар тўғрисидаги MSIE хабарлардан фойдаланишда хатолар тўғрисидаги хабарлар аввало учбурчакли белгича сифатида пастки чап бурчакда пайдо бўлади. Бу учбурчак ичида ундов белгиси бўлади. Бундан ташқари саҳифада хатолар учрагани хақида хабарлар келтирилади. Бу учбурчакка чертиб хатолар хақидаги хабарни очиш мумкин.

Хатолар тўғрисидаги хабарлар

Асосан хатолар икки хилда бўлади: синтаксистик ва бажарилиш вақтига боғлиқ хатолар. Синтаксистик хатолар нотўғри JavaScript конфигурацияси га боғлиқ бўлади. Бажарилиш вақтига боғлиқ хатолар буйруқларнинг нотўғри ишлатилишидан келиб чиқади.

Хатолар қатори

Хатолар қаторида хатолар хабари чиқарилганда бу хатони HTML хужжатнинг энг юқориги қисмидаги ёзувдан излаш лозим. Масалан қуйидаги мисолда 9 қаторда хатога йўл қўйилган. Бу синтаксистик хато, чунки параметр экземпляри шу қаторда тугалланмаган.

```
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<BODY>
<SCRIPT LANGUAGE="javascript">
document.write("саҳифа учун матн"
)
</SCRIPT>
</BODY>
</HTML>
```

Нима учун хато 9 каторда қайд килинган, чунки ҳисоб HTML хужжатнинг юқориги четидан бошланади. Бу хужжат қуйидаги кўринишга келтирилади.

```
(1-қатор) <HTML>
(2-қатор) <HEAD>
(3-қатор) <TITLE></TITLE>
(4-қатор) </HEAD>
(5-қатор) <BODY>
(6-қатор)
(7-қатор) <SCRIPT LANGUAGE="javascript">
(8-қатор) document.write("text for the page"
(9-қатор) )
(10-қатор) </SCRIPT>
(11-қатор) </BODY>
(12-қатор) </HTML>
```

Топшириқ

Қуйида хатоларга эга сценарий келтирилган. Уни юклашда браузер иккита хато тўғрисида хабар чиқаради. Уларни тўғирланг. Баъзи буйруқлар нотаниш бўлади, аммо бу ҳозир муҳим эмас. Хато тўғрисидаги хабарда уни тўғрилашга йўналтирувчи ахборотлар етарли бўлади. Агар сценарий тўғри бажарилса, саҳифада жорий сана акс этирилади. Аввал биринчи хато ҳақида хабар чиқиши ва у тўғрилангандан кейин иккинчиси ҳақида хабар чиқиши ҳам мумкин.

Хатоли код

```
<SCRIPT type="text/javascript">
...x
dothis = new Date()
month = dothis.getMonth()
month = (month * 1) + 1
day = dothis.getDate()
year = dothis.getFullYear()
document.wrte(" ",month,"/",day,"/",year," ")
</SCRIPT>
```

Ечилиш варианты

Скриптнинг вазифаси санани саҳифага чиқариш. Уни ишлашга мажбур қилиш учун хатоларни тўғрилаш лозим. Биринчи хато синтаксистик хато бўлади, яъни **...x** аниқланмаган. Бу хатони тўғрилашнинг энг осон усули уни ўчириб юборишдир. Сценарийни қайта ишга тушурганимизда иккинчи синтаксистик хато ҳақида хабар чиқади, **document.wrte** нинг функция эмаслиги ҳақидаги хабар бўлади, яъни **write** калит сўзи нотўғри ёзилган.

2-амалий машғулот.

Мавзу: Объектлар ва усулларни аниқлаш

Биз қуйидаги усулларни аниқлаймиз: **getDay()**, **getDate()**, **getMonth()**, **getYear()**, **getHours()**, **getMinutes()** ва **getSeconds()** (кун, число, ой ва йилни олиш ва соат, минут,

секундни олиш). Уларнинг хаммаси мавжуд, уларни олиб Web - саҳифага жойлаштириш мумкин. Аммо муаммо шундаки, улар фақат мавжуддирлар ва улар услуллардир, холос. Уларнинг харакати учун объект керак ва уни яратишимиз лозим.

Сценарий

```
<SCRIPT LANGUAGE="JavaScript">
//Сценарий аниқ сана ва вақтни ва саҳифага кириш вақтини чиқаради
Now = new Date();
document.write("Бугун " + Now.getDate()+
"- " + Now.getMonth() + "- " + Now.getFullYear() + ".
Сиз Web-саҳифага куйидаги вақтда кирдингиз: " + Now.getHours() +
":" + Now.getMinutes() + " и " + Now.getSeconds() + " секунд.")
</SCRIPT>
```

Шуни қайд қилиш лозимки, **document.write** қатори иккинчи қаторга кўчирилмайди. Бу ерда эса у бир нечта қаторга ёзилишидан мақсад эса уни ўқишни осонлаштиришдан иборатдир.

Сценарий ишлаши натижаси

Бугун 18-7-2008. Сиз бу саҳифага куйидаги вақтда кирдингиз: 8:42 ва 3 секунд.

Сценарийни таҳлил қиламиз

document.write қатори узунлиги экрандан чиқиб кетади. Агар бу қаторни иккига бўлсак, браузер хато чиқаради. Бу холда // сценарий ичидаги комментария эканлигини кўрсатади, у ундан кейин келадиган матн жараёнда ишлатилмаслигини билдиради.

Date (Сана) ва Time (Вақт)

Биз ойни аниқлаш усули **getMonth()** билан танишайлик. Энди бу усул қайси объектга таъсир қилиши мумкинлигини кўрайлик. **getMonth()** усули аслида Date объекти усулидир. Date объекти куйидаги буйруқ билан аниқланади:

```
Now = new Date(). getMonth() усули ана шу объект билан ишлайди. Демак аввало объектни яратишимиз лозим. Хозирги холда объект сифатида Now (хозир) ишлатилади. Уни биз исталганча ном билан номлашимиз мумкин эди. Now объекти шундай объектки, у newDate() шаклида аниқланади, сана албатта янги бўлиши лозим. Демак биз хар сафар саҳифага кирганимизда янги сана оламиз. Шундай қилиб бизда объект бор ва унга getMonth() усули билан таъсир қилиш мумкин.
```

Бизнинг саҳифамизга ой номи чоп қилиниши лозим, демак **document.write()** буйруғини хам ишлатишимиз лозим. Натижада куйидаги скриптга эга бўламиз:

```
<SCRIPT LANGUAGE="javascript">
//Бу сценарий саҳифага ой номерини чиқаради
Now = new Date();
document.write("Хозир ой " + Now.getMonth())
</SCRIPT>
```

Узун қаторларни яратиш куйидаги сценарийда келтирилади:

```
document.write("Бугун "
+ (Now.getMonth()+1)+
"- " + Now.getDate() + "- "
+ Now.getFullYear() + ".
Сиз Web-саҳифага куйидаги вақтда кирдингиз: "
+ Now.getHours() +
":" + Now.getMinutes() + " ва "
+ Now.getSeconds() +
```

" секунд")

Топшириқ

Шундай сценарий ёзингки, у Web-саҳифага санани энгашган тўғри чизик билан ажратиб чиқарсин. Табрик матни яшил рангда бўлсин ва сценарийни ким ёзганлиги ҳам кўрсатилсин.

Ечилиши варианти

Салом! Бугун бизда 18/8/2006.

Бу сценарийни қуйидагича ёзиш мумкин:

```
<SCRIPT LANGUAGE="JavaScript">
// Сценарий муаллифи tttt@mail.ru
Now = new Date();
var mpo = Now.getMonth();
var mpo1 = mpo + 1
document.write("<FONT COLOR='green'>Салом! Бугун бизда :</FONT>"
+ my.getDate() + "/"
+ mpo1 + "/"
+ my.getFullYear() + ".")
</SCRIPT>
```

Бу кодни браузерда текшириб натижа олиб кўринг.

3-амалий машғулот.

Маъзу: JavaScript ходисалари. MouseOver ходисаларни ишловчиси.

Бизга маълумки, (event) ходисаси ва ходисаларни ишловчи (event handler) ҳар ҳолда JavaScript га тегишли, аммо у HTML-кодга киритилган. Ходисалар ички киритилган бўлганликлари учун улар учун **<SCRIPT>** ва **</SCRIPT>** ишлатилмайди. Улар алоҳида сценарий эмаслар, улар унча катта бўлмаган саҳифа ва уни ўқувчи оралиғидаги мулоқатни амалга оширувчи интерфейслар ҳисобланади.

Ходиса — бу рўй берилувчидир. Улар Web-сайтларга динамика қўшимча қилади. Биз ҳали кўплаб ходисаларни кўриб чиқамиз. Биз аввало энг кўп ишлатилувчи **onMouseOver** (сичқонча курсорига кириш) ходисасини ўрганишдан бошлаймиз.

Сценарийси

```
<A HREF="http://www.mail.ru"
onMouseOver="window.status='Почта хизмати';
return true">Хавора</A>
```

Бунинг ҳаммаси битта қаторда ёзилиши лозим. Бу ерда у уни кўриб чиқиш осон бўлиши учун бир неча қаторга бўлиниб ёзилган.

Сценарий бажарилиши натижаси

Хаворага сичқонча кўрсатгичи олиб келинса браузернинг ҳолат қатори ўзгаради.

Сценарийни таҳлил қилиши

Биринчидан, **onMouseOver** - бу (Event Handler) гиперматнли хавора ходиса ишловчидир. Гиперматнли хавора формати ўзгаришсиз қолади. **onMouseOver** ходиса ишловчиси URL манзилдан кейин ёзилади. Браузер **onMouseOver=""** билиб олиши билан

(Event) ходисаси харакатга келтирилади. Шу вақтгача бир бири билан нуқта билан ажратилган элементлардан биринчиси объект ва иккинчиси усулни билдирар эди. Аммо бу холда объект сифатида **window** (ойна) ишлатилмоқда, **status** (статус) эса ойна хусусиятини билдиради. Хусусият отда ифодаланади ва у элементнинг катта бўлмаган қисми сифатида нуқтадан олдин ёзилади, усул эса феъл сифатида ифодаланади, масалан **write** (ёзмок) ёки **get** (олмок).

window.status дан кейин баробар белгиси ёзилади **=**, ундан кейин рўй берилувчи ёзилади, масалан, бизнинг холда бирлик қўштирноқга олинган матн бўлиб, у биз сичқонча кўрсатгичини гиперматнли хаворага олиб борганимизда холат қаторида пайдо бўлади. Охирги қаторда **return true** келмоқда, бу сўзлар рўй берувчи ходисага бевосита таъсир қилаолади. Агар бу ёзувлар бўлса сценарий холат қатори борлигини текшириб кўради. Агар текшириш муваффақиятли бўлса (**true**), ходиса рўй беради .

Бошқа хусусиятлар

HTML да саҳифа фони рангини **BGCOLOR** буйруги бошқаради. JavaScript да фон ранги **bgColor** билан белгиланади. Биз **onMouseOver** ходисаси ёрдамида ойнанинг фон рангини ўзгартирувчи хавора яратишимиз лозим.

1. Биринчидан, бу хавора бўлгани учун олдин ишлатилган форматни саклашимиз лозим.
2. Нима ўзгармоқда, ойнами ёки хужжатми? **bgColor** буйруги Web-саҳифа яратилишида қаёқга кўрсатмоқда? Хужжатгами. Демак, у керакли объектдир. Энди **window** ни **document** га алмаштирамиз.
3. Биз объектнинг фон рангини ўзгартирмоқчимиз, шу сабабли **status** ни **bgColor** га алмаштирамиз.
4. Матннинг пайдо бўлиши бизга бошқа керак эмас, шу сабабли унинг рангини ўзгартирамиз, масалан **pink** га ўзгартирамиз.
5. Бизнинг ўзгартирган рангимиз хаворага неча марта курсорни олиб боришимизга қарамай ўзгармай қолиши лозим, шунинг учун **return true** ни нуқтали вергулдан кейин кўямиз.

Қуйидагига келдик:

```
<A HREF="http://www.mail.ru"
onMouseOver="document.bgColor='pink';
return true">Шу ерга қирсиллатинг</A>
```

Топшириқ

Бу топшириқда янги **alert()** (огохлантириш) усулидан фойдаланиш тавсия қилинади. У саҳифага матн ва ОК кнопасини унча катта бўлмаган мулоқат ойнада чиқаради. Огохлатириш сичқонча кўрсатгичини хаворага хар сафар олиб борганимизда пайдо бўлишлиги таъминлансин. Бу буйруқ қуйидаги форматга эга:
alert('ойнага чиқарилувчи матн').

Ечилиши варианти

```
<A HREF="http://www.mail.ru"
onMouseOver ="alert(' Ойнани ёпиш учун ОК га чертинг');"
>Шу ерга чертинг</A>
```

4-амалий машғулот.

Мавзу: Ўзгарувчилар билан ишлаш. Фойдаланувчига суровлар

Бу машғулотида иккита концепция қаралади. Улардан биринчиси фойдаланувчидан ахборот сўралганда ишлатилади. Иккинчиси — ўзгарувчиларни яратиш — улар JavaScript да ишлаш мобойнида доимо ишлатилади. Биз ўз исмибизни киритишимиз лозим ва шу исмибиз билан ўзгарувчи боғланади. Ўзгарувчи қийматланганда, биз исмибизни `document.write` қаторига киритишимиз лозим. Бу оператор эса исмибизни саҳифага чиқаради.

Сценарий

```
<SCRIPT type="text/javascript">
```

```
/* Бу сценарий фойдаланувчидан олинган ахборотни саҳифага чиқариш учун хизмат қилади */
```

```
var user_name = prompt ("Пастки майдонда ўз исмибизни киритинг ", "Шу ерда");
```

```
document.write("Салом, " + user_name + "! Менинг саҳифамга хуш келибсиз !");
```

```
</SCRIPT>
```

Қавсдаги матн битта қаторга сиғдирилиши лозим.

Сценария ишлаши натижаси

Салом, OL. Менинг саҳифамга хуш келибсиз!

Сценарий тахлили

/ */ нимани билдиради?*

Бу символлар сценарий ичида коментариялар яратиш учун ишлатилади. Олдинги дарсларимиздан биз биламизки коментариялар яратиш учун иккилик `//` энгашган чизикни ҳам ишлатиш мумкин. Аммо у хар бир коментария қатори бошига қўйилиши лозим. Бу ердаги символлар ёрдамида коментариялар қаторлари шу символлар ўртасига ёзилади, нечта қатор бўлишидан қатъий назар фақат шу символлар ичига қаторларнинг ҳаммаси олинади.

Ўзгарувчиларни яратиш

JavaScript да ўзгарувчилар жуда муҳим аҳамиятга эгадирлар, шу сабабли уларни яратишни билиш лозим. Кўпинча усулларни кўплаб марта битта сценарийда ёзишга тўғри келади, биз масалан, `getDate()` бир марта ишлатганимиз сабабли кийинчилик бўлмади. Кўплаб ишлатилганда уларнинг қийматини битта ўзгарувчига чиқариш эҳтиёжи ҳам туғилади. Масалан, `d` ўзгарувчини олайлик. У холда фақат бир марта `getDate()` ёзилиб, сўнгра усул натижасини `d` га қийматлаш лозим. Сценарийдаги бошқа ҳамма холларда сана қўйилиши талаб қилинганда эса `d` ни ёзиш лозим. Мисолга қайтамыз.

Куйидаги скрипт қатори ўзгарувчини яратишга хизмат қилади:

```
var user_name = prompt ("Пастки майдонда ўз исмибизни киритинг ", "Шу ерда")
```

Ўзгарувчи куйидаги схема бўйича яратилди.

- `var` сўзи ундан кейин ёзилган сўз ўзгарувчи номи бўлишини эълон қилади .
- `user_name` (фойдаланувчи номи) — ўзгарувчи номи. У узун бўлиши шарт эмас. Масалан, оддий `N` ёзиш ҳам мумкин эди. Аммо одатда эслаш осон бўладиган номлар берилади.
- JavaScript да регистр аҳамиятга эга. Масалан, агар ўзгарувчи `Dog` билан белгиланган бўлса, у хар сафар катта `D` билан бошланиши лозим.
- Бу ерда тенглик белгиси `=` ўзгарувчи қиймати куйидаги буйруқ натижасига тенг бўлишини кўрсатади.
- Бизнинг мисолда эса ўзгарувчи сўров ойнаси ёрдамида олинган натижани ифодалайди.

Prompt буйруғи

Қуйидаги мисолда ўзгарувчиларни яратишда **prompt** (сўров) буйруғидан фойдаланиш келтирилади. Бу усул хабарли ва киритиш мойдонли ойнани чиқариш учун ишлатилади. Сўров формати қуйидаги форматга эга:

```
var variable_name = prompt("Ойнадаги матн ", " Киритиш майдони матни")
```

Бу ёзувдан кўриниб турибдики **var** ва қийматланаётган ўзгарувчи номи форматга киритилган.

Энди юқоридагиларни ҳисобга олиб яна сценарийга қайтайлик:

```
var user_name = prompt ("Ўз номингизни пастки майдонга киритинг ", "Шу ерда ");
document.write("Салом, " + user_name + "! Менинг саҳифамга хуш келибсиз!");
```

Тахлили

1. Бу ерда `user_name` ўзгарувчи номига сўров натижаси қийматланган .
2. `prompt` эса фойдаланувчига киритиш майдонида ўз номини ёзишини сўрамоқда .
3. Киритиш майдонида қуйидаги ёзилган: "Шу ерда"
4. Қатор охирига нуқтали вергул қуйилган.
5. `document.write` эса қуйидаги матнни чақиради "Салом, "
6. Плюс `+` белгиси ҳамма элементлар бир бири кетидан келишини билдиради.
7. `user_name` эса сўров натижасини сақлайди.
8. Навбатдаги плюс.
9. Нуқтали вергул.

Топшириқ

Биз энди етарли даражадаги қийинчиликга эга бўлган эффектларни JavaScript.буйруқлари ёрдамида яратиш билимига эгамиз.

Топшириқ сифатида қуйида келтирилган сценарий тахлилини бажариш талаб қилинади.

Топшириқ:

Қуйидаги сценарийни мустақил тахлил қилинг:

```
<SCRIPT type="text/javascript">
var name =
  prompt("Марҳамат,уз исмингизни ёзинг", "")
var d = new Date();
var y = d.getFullYear();
var m = d.getMonth() + 1;
var d = d.getDate();
var t = m + '/' + d + '/' + y + ' ';
document.write("<TITLE>")
document.write
("Салом "+name+ ". Бугун "
+t+ ". Кирганингиз учун раҳмат.");
```

```
document.write("</TITLE>")
</SCRIPT>
```

Тахлил варианты:

- `<SCRIPT LANGUAGE="javascript">` қаторидан бошлаймиз.

- **name** номли ўзгарувчи яратилади. Фойдаланувчи ўз номини **prompt** сўрови бўйича киритади.
- Бизга сана керак, шу сабабли сана учун **d** ўзгарувчиси яратилмоқда.
- Биз биламизки йил, ой ва число **d.getНимадир()** орқали аниқланади, ва сон ҳам **d.getНимадир()** орқали аниқланади, шу сабабли бу элементларнинг ўзгарувчи яратамиз: **y, m** ва **d**.
- Энди **m** ўзгарувчисидан кейин **+ 1** ни киритамиз.
- Тўлиқ ёзилган сана учун **t** номли ўзгарувчи яратилади.
- **TITLE** буйруги учта буйруқлар ёрдамида яратилади **document.write**.
- Учта буйруқдан биринчисини **<TITLE>** буйруги бошлайди.
- Иккинчи оператор матн қаторини яратади.
- Учинчи буйруқ **document.write <TITLE>** буйругини тугаллайди.

5-амалий машғулот.

Мавзу: Хужжатнинг объекти модели. Асосий объектлар ва уларнинг хусусиятлари

Концепцияси

JavaScript да объектлар иерархияси жуда муҳим аҳамиятга эга. Биз биламизки, шундай объектлар борки, масалан, **document** ва шундай усуллар борки, масалан, **write** улар объектларга таъсир қилишади. Энди хусусият концепцияларини кўриб чиқамиз. Хусусият объектнинг қисмини ёки унинг сифатини ифодалайди.

Сценарий

Қуйида биз бир хил схемада яратилган бир нечта сценарийларни кўриб чиқамиз: хар бир **объект.хусусият (object.property)** учун ўзгарувчи яратилади, сўнгра ўзгарувчи **document.write()** буйруққа киритилади.

"navigator" (браузер) объекти хусусияти

```
<SCRIPT LANGUAGE="javascript">
var an = navigator.appName;
var av = navigator.appVersion;
var acn = navigator.appCodeName;
var ua = navigator.userAgent;
document.write("Сиз фойдаланаяпсиз <B>" +an+ "</B>,
версия " +av+ ".
<BR>Кодли номи " +acn+ ", сарлавха " +ua+ ".");
</SCRIPT>
```

Юқорида келтирилган матн битта қаторда ёзилиши лозим.

"document" объекти хусусияти

```
<SCRIPT LANGUAGE="javascript">
var bgc = document.bgColor;
var fgc = document.fgColor;
var lc = document.linkColor;

var al = document.alinkColor;
var vlc = document.vlinkColor;
var url = document.location;
var ref = document.referrer;
```

```

var t = document.title;
var lm = document.lastModified;
document.write("Бу саҳифа фони ранги <B>" +bgc+ "</B>.")
document.write("<BR>Шу саҳифа матни ранги <B>" +fgc+ "</B>.")
document.write("<BR>Шу саҳифа фавораси ранги <B>" +lc+ "</B>.")
document.write("<BR> Шу саҳифанинг фаол фавора ранги
<B>" +al+ "</B>.")
document.write("<BR>Саҳифанинг кирилган фавора ранги
<B>" +vlc+ "</B>.")
document.write("<BR>Шу саҳифа URL <B>" +url+ "</B>.")
document.write("<BR>Бунгача сиз куйидаги саҳифада эдингиз <B>" +
ref+ "</B>.")
document.write("<BR>Шу саҳифа сарлавҳаси (TITLE) <B>" ++ "</B>.")
document.write("<BR>Хужжатга киритилган охириги ўзгартирилишлар : <B>"
+lm+ "</B>.")
</SCRIPT>

```

"history" объекти хусусияти

```

<SCRIPT LANGUAGE="javascript">
var h = history.length;
document.write("Бу саҳифагача сиз кирганлар" +h+ "саҳифалар.")
</SCRIPT>

```

"location" (манзил) объектининг икки хусусияти

```

<SCRIPT LANGUAGE="javascript">
var hst = location.host
document.write("Саҳифа жойлашган жой : <B>" + hst + "</B>" )
</SCRIPT>
<SCRIPT LANGUAGE="javascript">
var hstn = location.hostname
document.write("Саҳифа жойлашган жой : <B>" + hstn + "</B>." )
</SCRIPT>

```

Сценарий бажарилиши натижаси

Бу саҳифа фони ранги #334775.
 Бу саҳифа матни ранги #334775.
 Бу саҳифа фаворасининг ранги #fbbcac.
 Бу саҳифа жорий фаворасининг ранги #d0550b.
 Бу саҳифага кирилган фавора ранги #fbbcac.
 Бу саҳифа URL <http://jsp.newmail.ru/les7.htm>.
 Бунгача сиз куйидаги саҳифада бўлгансиз.
 Шу саҳифа сарлавҳаси.
 Охириги тузатишлар киритилган вақт: Fri, 27 Jun 2003 00:09:36 GMT.
 Бунгача сиз 1 саҳифада булгансиз.
 Саҳифа www.mail.ru да жойлашган.

6-амалий машғулот.

Мавзу: history объекти хусусияти

Сценарийси

```

<SCRIPT LANGUAGE="javascript">

```

```
var h = history.length;
document.write("Шунгача сиз кирган сахифа " +h+ " сахифа.")
</SCRIPT>
```

history объекти жуда оммавий объект хисобланади. Фойдаланувчи одатда бир икки варақ юқорига ёки пастга силжишни хохлаб колади, бу ишни улар браузернинг "Олдинга" ва "Орқага" кнопкаларидан фойдаланиб амалга оширади. Бу ишни **history** объекти ёрдамида амалга ошириш мумкин.

Объект сифатида **history** кириш журнали хисобланади. Бу браузер ишлаш вақтида кирилган сахифалар рўйхати. Рўйхатда "Орқага" кнопкаси ёрдамида олдинги сахифаларни яна бир марта кўриб чиқиш мумкин. Объектнинг хусусияти сифатида **length** (узунлиги) ишлатилади. Шунингдек яна **go()** (бориш) усули ҳам ишлатилади, у рўйхат буйича берилган қадам билан харакатланишни таъминлайди.

location объектининг иккита хусусияти

```
<SCRIPT LANGUAGE="javascript">
var hst = location.host
document.write("Сахифа жойлашган жой : <B>" + hst + "</B>.")
</SCRIPT>
<SCRIPT LANGUAGE="javascript">
var hstn = location.hostname
document.write("Сахифа жойлашган жой : <B>" + hstn + "</B>.")
</SCRIPT>
```

Бу ерда объект сифатида **location** ишлатилмоқда. Бу JavaScript даги сахифа URL манзилдир. Мисолда **location** объектининг иккита хусусияти келтирилган: **host** ва **hostname**. Бу буйруklar тенг кучлидир, чунки улар бир хил масалани ечишади — яъни улар URL манзилни матнли форматда беради ёки серверга боғлиқ холда IP манзилни беради.

Топшириқ

Юқоридаги буйруklarнинг биридан фойдаланиб сценарий ёзинг. JavaScript объектлари иерархияси билан танишиб чиқайлик:

- Window
 - Parent
 - Self
 - Location
 - Href
 - Document
 - Image
 - Src
 - Form
 - Text
 - Submit
 - Checkbox
 - Top
 - Frames

Иерархия харакати натижаси

Хамма фаворалар энг юқорги объектдан бошланади - **window** (браузер ойнаси), сўнгра пастки поғонали объектлардан бошланади. Ойна ва рамкалар (frames) **window** объектига тегишлидир. Улар сценарийда биттадан қатнашса уларга фовора бериш шарт эмас. **Top**, **self**, **parent** ва **frames** —ойналар учун "ички(встроенные)" номлар ҳисобланади. Бир нечта мисоллар кўрайлик. Уларнинг иерархиясига эътибор беринг.

document.mypic.src = "pic1.gif"

Бошланишида **window** керак бўлмайди. Чунки усиз ҳам хамма нарса ойна ичида ҳисобланади. Қуйидаги буйруқ - **document.mypic.src** **mypic** номи тасвирга қаратади ва унинг мазмунини **"pic1.gif"** га ўзгартиради. Бизнинг ҳолда **document** (ҳужжат) — **mypic** элементи жойлашган саҳифа, **SRC** — эса (**"pic1.gif"**) элементи манбаси.

document.write(location.href)

write() — бу **document**. **Location.href** объекти усулидир. Бу ерда **location** ва **document** битта босқичга жойлашган. Бу дегани биз шу босқичдаги ҳужжат манзилени оламиз.

JavaScript объектлари иерархияси тахлили

- *Window*
 - Parent
 - Self
 - Location
 - Href
 - Document
 - Image
 - Src
 - Form
 - Text
 - Submit
 - Checkbox
 - Top
 - Frames
- Бу ерда энг чалкашлик шундаки, баъзи объектлар хусусият бўлишлари ҳам мумкин.
 - **Window** — бу фақат объект бўлиши мумкин.
 - **Document** ойна хусусияти ҳисобланади, аммо ўз навбатида ҳам объектдир.
 - **Form** — бу ҳужжат хусусиятидир, аммо у ҳам ўз хусусияти билан объект бўлади!
 - **Value** (қиймат) ва **SRC** (манбаа) — булар фақат хусусият бўлишади!
 - Бу ерда хамма объектлар ва хусусиятлар келтирилмаган. Аммо шулар ҳам уларнинг концепциясини тушуниш имконини беради. Хамма фаворалар юқоридан пастга қараб келади. Масалан, **window** дан бошланиб пастга қараб боради. Яъни **document.mytext.myform** ёки **mypic.src.document** шаклида ёзиш мумкин эмас. **Form** объекти **text** дан юқори поғонада.

Топшириқ

Сиз **window** дан бошлаб хамма фавораларни, агарда улар талаб қилинмасалар ҳам, ёзиб чиқинг.

1. Хамма формаларга фаворалар ёзинг.
2. **Iname** майдонларига фаворалар ёзинг.
3. **fname** майдонлари қийматларига фаворалар ёзинг.
4. Тасвирни **"marigold.gif"** га алмаштиринг.

Жавоблар

1. `window.document.myform`
2. `window.document.myform.lname.value`
3. `window.document.myform.fname.value`
4. `window.document.mypic.src=marigold.gif`

*7-амалий маишулот.**Мавзу: JavaScript функциялари. Уларни яратиши ва ишлатиши**Концепцияси*

JavaScript буйруқ ва ходиса натижалари бўйича ўзгарувчилар яратилишида улар номланади. Функцияларни яратишда ҳам шу ишлар қилинади, фақат ном энди бир нечта буйруқлар сериясига берилади. JavaScript нинг буйруқлар тўплами биттага бирлаштирилади.

Сценарийси

Функция сценарийси амалда икки қисмдан ташкил топади: функциянинг ўзи ва уни ишга туширувчи `onLoad` буйруғидан ташкил топади:

```
<SCRIPT LANGUAGE="javascript">
<!-- JavaScript ни кулламайдиган браузерлар бекитилсин
function dateinbar()
{
var d = new Date();
var y = d.getFullYear();
var da = d.getDate();
var m = d.getMonth() + 1;
var t = da + '/' + m + '/' + y;
defaultStatus = "Сиз қуйидаги саҳифага кирдингиз " + t + ".";
}
// ёпманг -->
</SCRIPT>
```

...ва `onLoad` в `<BODY>`:

```
<BODY BGCOLOR="xxxxxxx" onLoad="dateinbar()">
```

Сценарий иши натижаси

Сенарий иши натижаси ҳолат каторига чиқарилади.

Сенарийни таҳлил қилиши

Биз бу ерда янги `<!--` ва `-->` элементлари ишлатилди. Улар HTML танасидаги коментарияларда ишлатилади:

```
<!--Бу саҳифада кўринмас матн коментариялардир -->
```

Буларнинг сценарияларда ишлатишига сабаб халигача JavaScript ни қўлламовчи браузерларнинг мавжудлигидадир. Агар браузер JavaScript ни қўлламаса, у уни матн сифатида қабул қилади ва саҳифага чиқариб юборади.

Қуйидаги қоидаларга риоя қилиши лозим

- Коментария буйруқлари `<SCRIPT>` ва `</SCRIPT>` оралиғига жойлашиши лозим.
- `<!--` дан кейин ёзилган матн битта қаторга сиғиши лозим.
- `-->` буйруғидан олдин `//` белгиси ёзилади, акс ҳолда JavaScript уни матн қисми сифатида қабул қилади.

Таҳлилга қайтамиз

Бу ерда иккита иш бажарилади. Аввало сценариянинг биринчи қисмида функция яратилади. Шундан кейин HTML **<BODY>** ичига жойлаштирилган буйрук бу функцияни ишга туширади. Аввало функцияни таҳлил қиламиз.

```
function dateinbar()
{
var d = new Date();

var y = d.getFullYear();
var da = d.getDate();
var m = d.getMonth() + 1;
var t = da + '/' + m + '/' + y;
defaultStatus = " Сиз саҳифага кирдингиз " + t + ".";
}
```

Схема жуда тушунарли ёзилган. Биз аввало **function** калит сўзини ёзамиз, сўнгра исталган номни берамиз. Функцияларни яратишда функцияни ташкил этувчи буйруқлар **{ }** фигурали қавсларга олиниши лозим.

Топшириқ

Шундай функция яратингки, у иккита (**prompt**) сўровларини яратсин. Сўровларнинг биринчиси фойдаланувчига ўз номини киритсин, иккинчиси — фамилиясини киритсин. . Шундан кейин бу функция (**alert**) хабар ойнасини чақиради, бу ойнада куйидаги матн ифодаланади:

Салом, "исм фамилия", хуш келибсиз саҳифамга "саҳифа манзили", менинг ажойиб саҳифамга!

Саҳифа манзили учун ўзгарувчилар яратиш лозим.

Ечилиш варианты:

```
<SCRIPT LANGUAGE="javascript">
function hello()
{
var first = prompt(" Ўз исмингизни киритинг", "Исм")
var second = prompt(" Ўз фамилиянгизни киритинг", "Фамилия")
var page = document.location;
var ttl = document.title;
alert("Привет, " +first+ " " +second+ ". Саҳифага хуш келибсиз " +ttl+ " манзил: "
+page+".")
} </SCRIPT>
```

8-амалий машғулот.

Мавзу: Функциялар ёрдамида ойналар очииш

Концепцияси

Биз шу вақтгача биз янги ойнани **window.open** буйруғи билан очган эдик. Ойнага қавслар ичида келтирилган HTML хужжат чиқарилар эди. Биз энди янги ойна очувчи ва янги ойна ва ундагиларни HTML хужжатида сақловчи функция яратамиз.

Сценарийси

```
<SCRIPT type="text/javascript">
function openindex()
```

```

{
var OpenWindow=window.open("", "newwin", "height=300,width=300");
OpenWindow.document.write("<HTML>")
OpenWindow.document.write("<TITLE>Янги ойна</TITLE>")
OpenWindow.document.write("<BODY BGCOLOR='00ffff'>")
OpenWindow.document.write("<CENTER>")

OpenWindow.document.write("<font size=+1> Янги ойна </font><P>")
OpenWindow.document.write("<a href='http://www.intuit.ru' target='main'>
Бу фавора <BR> асосий ойнада очилади </a><p>")
OpenWindow.document.write("<P><HR WIDTH='60%'><P>")
OpenWindow.document.write("<a href=" onClick='self.close()'>
Эта ссылка закроет окно</a><p>")
OpenWindow.document.write("</CENTER>")
OpenWindow.document.write("</HTML>")
}
</SCRIPT>
... ва куйидаги буйруқда BODY:
onLoad="openindex()"

```

Сценарий иши натижаси

Иккита фаворали ойна очилади. Аммо хаммаси битта сахифада бўлади.

Сценарийни тахлил қилиш

Функцияни сақловчи сценарийнинг асосий қисми **<HEAD>** ва **</HEAD>** теглари орасига олинади. Оддий схемада функцияга **openindex()** номи берилади. Шундан кейин фигурали кавслар ёзилади. Сўнгра **OpenWindow** ўзгарувчиси яратилади ва унда **window.open()** буйруғи киймати сақланади. У шундай кўринишга эга:

```
var OpenWindow=window.open("", "newwin", "height=300,width=300");
```

Бу формат таниш. Фарқи шундаки, бу ерда URL кўрсатилмаган. Унинг ўрнига бўш иккилик қўштирноқлар келтирилган. Улар браузерга янги ойна хақида ахборот излашини кўрсатади. Энди янги ойнада HTML сахифасини яратамиз. Матннинг биринчи қатори куйидагича бўлади:

```
OpenWindow.document.write("<HTML>")
```

Бу буйруқ матнинг қатори хужжатга **OpenWindow** (янги ойна) ўзгарувчиси орқали ёзилган бўлиши керак.

Топшириқ

Ойна очувчи функция яратилсин. Ойнада пайдо бўлувчи хужжат яшил фонга эга бўлиши лозим ва **TITLE** сарлавҳасига эга бўлиши лозим: "Салом, "фойдаланувчи исми", мана сизинг ойнангиз!" Фойдаланувчи номини (**prompt**) сўрови орқали билиш мумкин. Яна ойнани ёпувчи фавора хам қўшишимиз лозим.

Ечиш варианты:

```

<SCRIPT type="text/javascript">
function openindex()
{
var name=prompt("Сизнинг исмингиз?","Шу ерга ёзинг")

```

```

var OpenWindow=window.open("", "newwin", "height=300,width=300,status=yes");
OpenWindow.document.write("<HTML>")
OpenWindow.document.write("<TITLE>")
OpenWindow.document.write("Салом, " +name+ "! Мана сизнинг ойнангиз!")
OpenWindow.document.write("</TITLE>")
OpenWindow.document.write("<BODY BGCOLOR='green'>")

OpenWindow.document.write("<CENTER>")
OpenWindow.document.write("<h2>Янги ойна</h2>")
OpenWindow.document.write("<a href=" onClicк='self.close()'>бу фавора ойнани ёпади</a>")
OpenWindow.document.write("</CENTER>")
OpenWindow.document.write("</BODY>")
OpenWindow.document.write("</HTML>")
}
</SCRIPT>
<body bgcolor="xxxxxx" onLoad="openindex()">

```

9-амалий машғулот.

Мавзу: PHP нинг асосий синтаксиси

Биз биламизки, PHP Web дастурлашда жуда кенг ишлатилади. Биз PHP синтаксиси билан қисман танишамиз. PHP файлларни таҳлил қилишида файл матнини узатади, аммо PHP нинг махсус тэги учраса, файл матни энди PHP коди шаклида интерпретация қилинади. PHP ёпувчи тэги учраши билан шу вақтдан бошлаб оддий текст узатилади. Бу механизм PHP кодларни HTML га қиритиш имкониятини яратади. Махсус PHP тэгларидан ташқаридагиларнинг ҳаммаси узғаришсиз қолади. PHP кодлари блокларини белгилаш учун туртта тэглари терилмаси ишлатилади. Аммо улардан фақат иккитаси (<?php. . .?> ва <script language="php">. . .</script>) ҳамма вақт ишлатилиши мумкин. Қолганларини php.ini файлларига улаш ва уларни ўқириш мумкин.

PHP тэглари:

1. <?php echo("агар биз XHTML ёки XML ҳужжатлари билан ишласак , \n");?> лар ишлатилади.
2. <? echo ("бу энг сода SGML-инструкция ,\n");?>
<?= ифода ?> Бу аббревиатура , "<? echo учун ифода?>"
3. <script language="php"> echo </script>
4. <% echo ("Биз ASP стилидаги тэглари ишлатишимиз мумкин "); %>
<%= \$variable; # Бу "< echo . . ." %> учун аббревиатура .

PHP қуйидаги таркибларни ишлатиши мумкин:

```

<?php
if ($expression) {
    ?>
    <strong>Бу тўғри.</strong>
    <?php
} else {
    ?>
    <strong>Бу нотўғри.</strong>
    <?php
}
?>

```

PHP да хар бир оператор нуктали вергул билан тугалланади. Ёпувчи тэг (?>) ҳам операторни ёпувчи хисобланади. Шу сабабли куйидаги вариантлар эквивалентдирлар:

```
<?php
    echo "This is a test";
?>
```

ва <?php echo "This is a test"?>

Комментариялар

PHP 'C', 'C++' ва и Unix кобикларини кўллайди. Масалан:

```
<?php
    echo "This is a test"; // Эбу C++ стилидаги бир каторли коментарийдир
    /* Бу куп каторли коментарий,
       */
    echo "This is yet another test";
    echo "One Final Test";
```

Концепцияси

Ушбу **confirm** (кўллаб қувватлаш) буйруғи **alert** усулига ўхшаш тасвир қилади, аммо мулоқат ойнага "Бекор қилиш" (Cancel) кнопкасини қўшимча қилади. Уларнинг иккаласи ҳам усуллардир.

Битта буйруқ ўз - ўзидан кўп нарса бермайди. Агар **IF** (агар) ва **ELSE** (акс холда) қўшимча қилинса, жуда чиройли эффектлар хосил қилиниши мумкин.

Сценарийси

Аввало базавий форматга қарайлик:

```
<SCRIPT type="text/javascript">
confirm("Киришга хохишингиз бориға ишончингиз комилми?")
</SCRIPT>
```

Бу сценарийга бироз қўшимчалар киритамиз:

```
<SCRIPT type="text/javascript">
if (confirm("INTUIT га киришга хохишингиз бориға ишончингиз комилми?") )
{parent.location='http://www.intuit.ru/';
alert("Ок йўл");}
else
{alert("Унда қолинг");}
</SCRIPT>
```

Лаборатория машғулотлари

1-лаборатория иши

Мавзу : HTML ва JavaScript ёрдамида шахсий web –саҳифалар яратиши

1.1.HTML ёрдамида шахсий web саҳифалар яратиши

Фойдаланувчи форма майдонида киритган ахборотларини функцияга узатамиз. Сўнгра бу ахборотлар Yahoo да излашларда ишлатилади .

Сценарийси

```
<SCRIPT type="text/javascript">
```

```
function Gofindit(){
  var searchfor = document.formsearch.findthis.value;
  {
  var FullSearchUrl =

"http://av.yahoo.com/bin/query?p=" + searchfor ;
location.href = FullSearchUrl;
}}
</SCRIPT>
<FORM NAME="formsearch" action="">
Yahoo да топинг:
<INPUT NAME="findthis" SIZE="40" TYPE="text">
<INPUT TYPE="button" VALUE="Излаш"
onClick="Gofindit()">
</FORM>
```

Сценарий ишлаши натижаси

Сахифага куйидаги қатор чиқарилади " Yahoo дан топинг:", киритиш майдони ва "Излаш" ёзувли кнопка.

HTML>

Yahooдан топинг

Сценарий тахлили

Бу сценарий объектлар иерархиясини жуда мукамал билиш лозимлигини эслатади.

1. Биринчидан, **findthis** (топиш) элементи ичида **formsearch** номли **searchfor**(излаш) ўзгарувчили функция яратамиз, у **value** (қиймати) хусусиятига эга. Бу **document** объектида нимадир бўлиши натижаси бўлади.
2. Иккинчи функцияни биринчисининг ичига жойлаштирамиз ва иккинчи жуфт фигурали қавсларни киритамиз.
3. Иккинчи функция учун яна битта **FullSearchUrl** ўзгарувчисини киритамиз, у Yahoo излаш машинаси манзилни ифодалайди.
4. Нахоят, **location.href** қиймати ўзгарувчи **FullSearchUrl** нинг қийматига **тенглаштирилади**.
5. Энди форма буйруқларига ўтамиз. Улар иккита: (**ТЕХТ**) матн майдони, у ерга фойдаланувчи ўз сўровларини киритади ва функцияни ишга туширувчи кнопка.
6. Эсимиздаки умуман олганда форма **formsearch** деб аталади.
7. Энди матн майдонига **findthis** деб ном берамиз.
8. Сўнгра кнопкани **onClick** билан боғлаймиз, у функцияни ишга туширади.

Топшириқ

Сценарийни шундай ўзгартирингки у бошқа излаш машинасини чақирсин. Бундан ташқари сўров жўнатилишида "Хозир излаймиз..." деган ойна пайдо бўлсин.

Ечилиши варианти

Yandex да излаш. Агар излаш машинаси манзили маълум бўлса, у холда ҳеч қанақа қийинчилик бўлмайди. Шу сабабли биз олдин бу манзилни топиб олишимиз лозим.

Сценарийси:

```
<SCRIPT type="text/javascript">
function Gofindit()
{

alert("Хозир излаймиз...");
var search = document.formsearch.find.value;
{
var searchUrl = "http://www.yandex.ru/yandsearch?ctgl=11657&text=" + search;
location.href = searchUrl;
}}
</script>
<form name="formsearch">
Излаш : Яндех:
<input type="button" value="Найти" onClick="Gofindit()">
</form>
```



1.2. Web саҳифалари учун CSS файлини яратиш

CSS технологияси

CSS сузи инглизча Cascading Style Sheets "Каскадли стиллар жадвали" сузидан келиб чиккан. Дастлабки стиллар HTML 4.0 да пайдо булди. Стиллар одатда стиллар жадвалларида сакланади. Улар HTML-хужжат ичида ва махсус css кенгайтмали файлларда сакланишлари мумкин. Стилларни жадвалларда саклаб иш хажмини жиддий камайтириш мумкин.

CSS дан фойдаланишга мисоллар

CSS дан фойдаланиб HTML хужжатларига хар хил чикариш стилларини бериш мумкин.

Бир нечта стиллар жадвалларини олайлик:

Style1

```
p
{color: blue; font-style: italic;
margin-left: 20pt;}
```

```
ol,ul,li
{list-style: circle;
font-family: arial; }
a:link {color:black}
```

```

a:visited {color:black}
body
{background-image: url("http://www.intuit.ru/departament/picture.gif")}
Style2
h1,h3,h6
{margin: 20pt;}
h1,h2,h3,h4,h5,h6,p,ol,li
{font-family: arial, "sans serif";}
p,table,td,ol,ul
{font-style: oblique;}
:link {COLOR: blue;}
:visited {COLOR: blue;}
a:active {COLOR: black;}
a:hover {COLOR: black;}
a.myln:link {COLOR: olive;}
a.myln:visited {COLOR: olive;}
a.myln:active {COLOR: teal;}
a.myln:hover {COLOR: teal;}
Style3
p,ol,ul
{font-style: italic;
margin-right: 20pt;}
p,h1,h6
{font-family: "comic sans ms", arial, "sans serif"; }
p,th
{font-size: 20%}
table {background-color: blue}
body {background-color:#FFF000}
p,ol,ul
{color:green; }
:link {color:blue}
:visited {color:blue}
:active {color:green}
:hover {color:green}
Style4
:link { color:red }
:visited { color:red }
:active { color:yellow }
:hover { color:yellow }
body
{background-color: #FAD123;}
h1,h2,h6
{margin-left: 20pt;
color: blue;}
h3,h4,h5
{margin-left: 10pt;
color: green;}

p,h1,h3
{font-size: 150%;}
p,h1,h2,h3,h4,h5,h6,table
{font-family: arial;}

```

```

p
{margin-left: 30pt;}
Style5
p,h1,h2,h3
{font-size: 75%;}
table,h4,h5,h6,li
{font-family: arial, "sans serif";
margin-left: 20pt;}
body
{background-color: #F2FF2F;}
ol,ul {color: 12a45d;}
:link {color:brown}
:visited {color:brown}
:active {color:blue}
:hover {color:blue}

```

Бу стилларни битта хужжатга кетма - кет куллаб, хар хил хужжат ифодаланишини хосил килиш мумкин.

Стиллер жадвали иш хажмини жиддий камайтириши мумкин

Стиллер жадвали HTML элементларини чикаришни аниклайди (масалан, **font** тегининг **color** атрибутини). Одатда стиллар жадваллари css файлларда сакланади. Бу усул web даги ташки курунишларни узгартиришда жуда кул келади. CSS web дизайндаги жуда катта ютук булиб колди. Бу технология бир вақтда стиллар ва компановкаларни бирданига бошкаришни таъминлайди.

Web сахифа яратувчилар стилларни куйидагича аниклашади :

- HTML даги ягона элементи ичида;
- HTML сахифаси **<head>** элементи ичида;
- CSS ташки файлларида .

Синтаксиси

CSS синтаксиси уч кисмдан иборатдир: селекторлар, хусусият ва киймат:

селектор {хусусият: киймат}

Селектор — бу аникланиши лозим булган HTML элемент/тег. Хусусият —узгартирилиши лозим булган атрибут. Хар бир хусусият киймат кабул килади.

1. Хусусият ва киймат икки нукта билан ажратилади ва фигурали кавс ичига жойлаштирилади:

```
p {font-size: 75%}
```

Мисол:

CSS технологияси ташки жадвал ва стиллар билан богланишни таъминлайди.

Бунинг учун HTML файлида **<link>** теги ишлатилади:

```

<html>
<head>
<link rel="stylesheet" type="text/css"
style="color:#666666;"
href="/external/?popup=0&url=http%3A%2F%2Fwww.intuit.ru%2Fimage%2Fstyle.css"
onMouseOver="menuSetHelpText('intuit');return false;"
onMouseOut="menuClearHelpText()" title="ссылка на INTUIT.ru">
</head>
<body>
<ol><!--Рўйхат 10 пунктдан иборат ташки чегарага эга -->

```

```

<li>рўйхат элементи учун "arial"</li> шрифти берилади
<li>рўйхат элементи учун "arial"</li> шрифти берилади
</ol>
<ul><!-- Рўйхат 20 пунктдан иборат ташки чегарага эга -->
<li> рўйхат элементи учун "arial"</li> шрифти берилади

```

1. рўйхат элементи учун "arial" шрифти берилади
2.
- 3.

```

</body>
</html>

```

Ташки файл куйидаги стилларни тавсифлайди (style.css):

```

ol {margin-left: 10pt}
ul {margin-right: 20pt}
li {font-family: arial}

```

2-лаборатория иши

Мавзу: Интерактив Web саҳифалар яратиш

2.1. JavaScript ёрдамида интерактив Web саҳифаларини яратиш

Бу лаборатория ишида саҳифа билан мулоқатли сценарийлар яратиш масаласи ўрганилади. Мисол сифатида тасодифий сонлардан фойдаланиш масаласи дастурланади. Тасодифий сонларни генерация қилиш учун JavaScript сана ва вақтни ишлатади. Биз бу имкониятлардан фойдаланиб сценарийлар яратишни ўрганамиз. Лаборатория ишининг мақсади ҳам шундан иборат.

Лаборатория иши топшириги.

- 1) 1 ва 10 оралиғидаги тасодифий сонларни танлаш сценарийсини яратинг.
- 2) Сценарийда тасодифий сонларни ҳосил қилиш функциясини яратинг. Ҳосил қилинган сценарийни таҳлил қилинг.
- 3) Функцияни ишга тушурувчи элементни сценарийга киритинг

Лаборатория ишини бажаришига услубий курсатмалар

Қуйидаги сценарийда 1 ва 10 оралиғидаги тасодифий сонлар танланади.

Сценарий

```

<HTML>
<HEAD>
<SCRIPT type="text/javascript">
function rand()
{
    var now=new Date()
    var num=(now.getSeconds())%10
    var num=num+1
    alert(num)
}

```

```

}
</SCRIPT>
</HEAD>
<BODY>
<h1>Тасодифий сонлар</h1>

<form>
  <INPUT TYPE="button"
    VALUE="1 дан 10 гача тасодифий сонлар "
    onClick="rand()">
</FORM>
</BODY>
</HTML>

```



Сценарийнинг ишлаш натижаси

Саҳифага "1 дан 10 га тасодифий сонлар" ёзуви билан кнопка чиқарилади, уни босиш билан тасодифий сон билан ойна ҳосил бўлади.

Таҳлили

Таҳлилни функциядан бошлаймиз.

```

function rand()
{
var now=new Date()
var num=(now.getSeconds())%10
var num=num+1
alert(num)
}

```

Тасодифий сонни танлаш бир нечта қадамларда амалга оширилади.

1. Биринчидан функция яратилади ва уни **rand()** деб атаймиз.
2. Сўнгра **new Date()** усулига ўзгарувчи яратамиз.
3. Яна битта ўзгарувчи яратамиз ва уни **num** деб номлаймиз. У **getSeconds()** усулини сақлайди, чунки биз бу ҳолда тасодифий сонни танлашда секундларни ишлатамиз.

JavaScript, ҳамма бошқа компьютер тиллари сингари ҳисобни нолдан бошлайди. Шу сабабли **%10** элементи JavaScript га 0 дан 9 гача бўлган сонлар орасидан тасодифий сонни танлаш кераклигини кўрсатади.

Бу ерда **%** оператори бўлиш қолдигини қайтаради. Айтайлик, **getSeconds()** функцияси 20 секунд деган қийматни қайтарган бўлсин. Уни 10 га бўлсак қолдиқ 0 га тенг бўлади. Демак сценарий 0 ни қайтаради. Айтайлик секунд сони 12 га тенг бўлсин. 10 га бўлишдан қолган қолдиқ 2 га тенг бўлади.

Тасодифий сонга бирни қўшиб (**num=num+1**), биз 1 дан 10 гача бўлган тасодифий сонларни оламиз.

Наҳоят **alert** сони чиқаради.

Энди функцияни ишга туширувчи элементни кўрайлик:

```
<form action="">
  <INPUT TYPE="button"
  VALUE="1 дан 10 гача тасодифий сонлар"
  onClick="rand()">
```

```
</form>
```

Бу оддий кнопка бўлиб, юқорида келтирилган функцияни ишга туширади.

Топшириқ

Шундай JavaScript дастурини ёзингки, унда фойдаланувчи формадаги кнопкани боссин , дастур эса 0 дан 4 гача бўлган тасодифий сонларни қуйидаги сўзлар билан чиқарсин: "Сизнинг тасодифий сонингиз: "x".

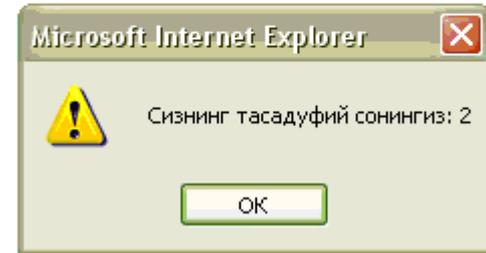
Ечилиш варианты

```
<html>
  <head>
    <SCRIPT type="text/javascript">
      function rand()
      {
        now=new Date()
        num=(now.getSeconds())%5
        alert("Сизнинг тасодифий сонингиз: " + num)
      }
    </script>
  </head>
  <body>
    <h1>Тасодифий сонлар</h1>
    <form>
      <input type="button" value="0 дан 4 гача булган тасодифий сонларни чиқариш "
      onClick="rand()">
    </form>
  </body>
</html>
```

Сценарийнинг натижаси:

ТАСОДИФИЙ СОНЛАР

0 дан 4 гача булган тасадуфий сонларни ч



2.2. Web саҳифалари учун кидириш дастурини яратиш.

Бу лаборатория ишида руйхатлардан излаш дастурини яратиш талаб килинади.

Лаборатория иши топшириги:

1. Вариант масалалари учун излаш дастурини яратинг.
2. Дастурни саҳифа объектига юкланг.
3. Натижаларни фреймга чиқаринг.

Лаборатория ишини бажариш учун услубий курсатмалар.

Куйида берилган картиналар рўйхатидан излаш масаласини дастурлаш келтирилади.

```
<!-- элементы заголовка -->
```

```
...
```

```
<SCRIPT>
```

```
<!--Скрипт матнини эски браузерлардан химояладингиз
```

```
pictures = new Array()
```

```
for(i=0;i<8;i++)
```

```
{
```

```
  pictures[i] = new Image()
```

```
  if(i==0) pictures[i].src = "freeze.gif"
```

```
  if(i==1) pictures[i].src = "dust.gif"
```

```
  if(i==2) pictures[i].src = "toster.gif"
```

```
  if(i==3) pictures[i].src = "cook.gif"
```

```
  if(i==4) pictures[i].src = "oven.gif"
```

```
  if(i==5) pictures[i].src = "wash.gif"
```

```
  if(i==6) pictures[i].src = "dishwash.gif"
```

```
}
```

```
function l_image()
```

```
{document.images[1].src = pictures [document.form1.item.selectedIndex].src}
```

```
// -->
```

```
</SCRIPT>
```

Суратни ўзгартиришни чакириш функцияси билан HTML-бўлак фрагменти:

```
<center>
```

```

<TABLE COLS=2 WIDTH="100%"  >

<CAPTION>
  <FONT COLOR="#000099" SIZE=+2>
  Маиший техника намуналари</FONT>
</CAPTION>
<TR>
  <th>
    <form name=form1>
    <select name=item onChange=l_image()>
      <option>Музлатгич
      <option selected>Чангютгич
      <option>Тостер
      <option>Стол
      <option>Духовка
      <option>Стирал машина
      <option>Идиш ювувчи машина
    </select>

    </form>
  </th>
</tr>
<tr>
  <th ALIGN=CENTER VALIGN=CENTER><
    IMG SRC="dust.gif" NAME="tool" > </th>
</TR>
</TABLE>
[<a href=#top>На начало страницы</a> ]
<HR>
</center>

```

Бу дастурда картина руйхат элементи сифатида танланади. Бу холда onChange, ходисаси рўй беради, унинг натижасига кўра қандай элемент танланилгани аникланади.

1. Храпцов П.Б., Брик С.А., Русак А.М., Сурин А.И.
Основы web-технологий
Интернет-университет информационных технологий - ИНТУИТ.ру, 2003
2. Капустин М.А., Капустин П.А., Копылова А.Г.
Flash MX для профессиональных программистов
Интернет-университет информационных технологий - ИНТУИТ.ру, 2006
3. Савельева Н.В. **Основы программирования на PHP**
Интернет-университет информационных технологий - ИНТУИТ.ру, 2005
4. Шохирев М.В.
Язык программирования Perl 5
БИНОМ. Лаборатория знаний, Интернет-университет информационных технологий - ИНТУИТ.ру, 2006
5. Сузи Р.А.
Язык программирования Python
БИНОМ. Лаборатория знаний, Интернет-университет информационных технологий - ИНТУИТ.ру, 2006

