

12 лабораториялық жұмыс. Формалар

Программалық басқару принципі

Браузер гипермәтіндік парақ ішіндегі әрбір тәг үшін объект жасай алады. Ал сол объектімен программалаушы name атрибутында көрсетілетін оның аты арқылы қатынас құра алады. Сонымен, тәгтерді программалық басқару мүмкіндігі пайда болады. Мысалы, HTML-кодта экранға сурет шығаратын мынадай тәг берілген болсын делік:

```
<IMG src=./pic/ris1.gif  
border=0 alt="" name=pic>
```

Осы тәг арқылы **pic** атты стандартты **image** объектісінің экземплярын мынадай эквивалентті конструкциялар арқылы пайдалануға болады:

window.document.images["pic"] **window.document.images.pic** немесе **ағымдағы терезе ішінде болатын болса:**

document.images["pic"] **document.images.pic** Мысалы, **image** объектісінің суреті бар файлға нұсқайтын **src** қасиеті бар екені белгілі. Программалауда осы қасиетті төмендегі жолды жазып оқуға болады: **document.images.pic.src** Тек қана оқып қана қоймай, бұл қасиеттің мәнін де өзгерте аламыз: **document.images.pic.src = "./pic/ris2.gif"** сонда сурет сәйкесінше басқа суретке ауысады.

2. <Form> блогы

Гипермәтіндік құжаттармен жұмыс істей отырып, мәтіндерді енгізу өрісіне жазып, меню пункттерін тандап, жалауша арқылы бір жолды тандап, батырмаларды шертуге болатынын көргенбіз. Мұндайда әрекетке байланысты құжаттың соған жауап беретінін байқаймыз.

Осы интерфейстік элементтердің бәрі браузер экранына контейнер-команда **<form>...</form>** арқылы шығарылады, ал батырмалар, енгізу өрісі, жалаушалар және меню осы блок ішіндегі әр түрлі тәгтер арқылы берілетін.

<form>...</form> блогы форма деп аталады. Оның ашылу тәгінде мынадай атрибуттар жазылады.

action - форманы желі бойынша жөнелту адресін береді. Форманы оны өңдейтін программасы (CGI-скрипт) бар серверге жіберуге болады немесе оның мәні ретінде электрондық пошта адресі көрсетілсе, форма **e-mail** арқылы жөнелтіледі. Мысалы, былай жазуға болады: **<FORM action="mailto:myaddres@mail.ru">** Егер ол көрсетілмесе форма сайттағы адресі бойынша жөнелтіледі. **enctype** – желі арқылы берілетін мәліметтерді кодтауды береді. Мұнда біз формадан алынған мәліметтерді өңдейтін программа-ларды қарастырмаймыз. Бірақ браузер тұтынушы толтырған форманы сіздің электрондық адресіңізге жібереді. Сол мәліметтерді өздерің қарап шығуға немесе басқа программа арқылы оны өңдеуге болады. Көбінесе төмендегідей толтырылған мәндер дұрыс жұмыс істейді:

<FORM action="mailto:myaddres@mail.ru" enctype="text/plain">

method – форманы тасымалдау тәсілін береді. Атрибуттың екі мәні бар: **get** және **post**. **post** тәсілі әмбебап болып саналады (форма URL-ден бөлек жіберіледі). **get** мәні форманың URL-мен бірге жіберілетінін көрсетеді. **e-mail** арқылы былай жіберуге болады:

<FORM action="mailto:myaddres@mail.ru" enctype="text/plain"

method="post"> **name** – форма атын береді. Мұны скриптерде форма өрістерін пайдалану үшін береді. Егер форманы ешқайда жібермеу керек болса, онда тек **name** көрсетіледі: **<FORM name=anketa>** Енді JavaScript-кодтарында осы форманы былай пайдалануға болады:

document.anketa. Егер форманың **work** атты жолдық өрісі болса, сол жолды айнұмалыға меншіктеуге болады:

var str = document.anketa.work.value; Осы өріс мәнін өзгерте аламыз:

document.anketa.work.value = "веб-мастер"; Мұнан кейін бұл өріс мәні өзгереді.

1.1. E-mail арқылы форманы жіберу

Төмендегі мысалды форманы электрондық пошта арқылы жіберудің үлгісі ретінде қарауға болады.

<FORM action = mailto:myaddres@mail.ru

enctype = "text/plain" method="post"> **Нажмите “сброс”и**
**заполните заказ.
** **<INPUT type=reset value=сброс>
** **<P>** **<TABLE border=0**
cellspacing=0 cellpadding=5> **<TR>** **<TD>Ваше имя:</TD>** **<TD>** **<INPUT**
type=text size=33 name=name **value="Горбунков Семен**
Семенович"> **</TD>** **</TR>** **<TR>** **<TD>Ваш адрес:</TD>** **<TD>**
<INPUT **type=text** **size=33** **name=email**
value="gorbun@mail.ru"> **</TD>** **</TR>** **<TR>** **<TD>Прислать:</TD>**
<TD> **<INPUT** **type=text** **size=33** **name=obj**

```
value="Программный соковыжиматель"> </TD> </TR> </TABLE> <INPUT
type=submit value="послать заказ"></FORM>
```

Егер осы кодты HTML-программаға кірістірсе, онда гипермәтіндік бетте төмендегі суреттегідей бейне пайда болады. Егер **сброс**, батырмасын басып келісім бойынша мәндерді өшіріп, жаңа мәндерді енгізуге болады. Форма толтырылған соң, **послать заказ** батырмасын шертіп оны мына адреске **myaddres@mail.ru** жөнелтеміз. Әдетте браузер мәлімет-ті өзі жөнелтпейді, оны компьютердің операциялық жүйесіндегі пошта программасы атқарады (мысалы, Outlook). Мәлімет мынадай түрде жөнелтіледі:

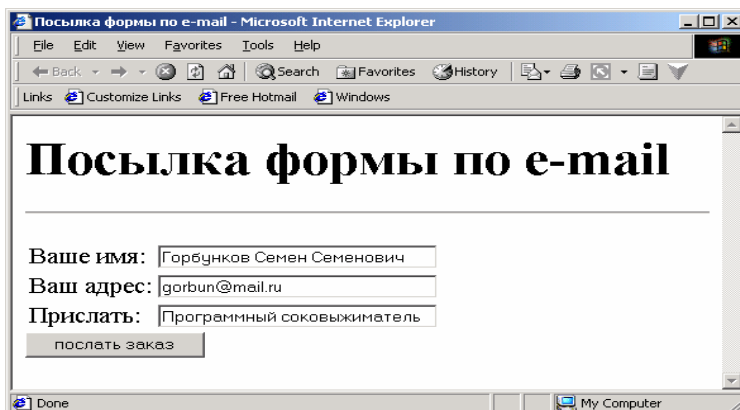
Subject: Форма отправлена из Microsoft Internet Explorer
name = Горбунков Семен Семенович
email = gorbun@mail.ru
obj = Программный соковыжиматель

3. Батырмалар. <INPUT> элементі

<input> тәгі арқылы формада әр түрлі егізу және басқару элементтерін орната аламыз. **Name атрибуты** Әрбір **<input>** тәгінде **name** атрибуты болады. Оның мәні браузер ашатын объект атауы (типi) болып табылады.

Мысалы, бір ғана мәтін енгізу жолы бар форма берілсін делік (оның экрандағы бейнесі суретте):

```
<FORM name=anketa>                   <INPUT
type=text value=крошкосшиватель
name=obj> </FORM>
```



JavaScript программасындағы мынадай конструкция **document.anketa.obj.value** объектінің **<input>** тәгі үшін тұрғызылған **value** қасиетін пайдаланады (өзгертеді), Мұндай жазуды былайша түсіну керек:

document – document объектісі (атасы (тегі) - window объектісі); **document.anketa** – anketa объектісі (атасы - document объектісі); **document.anketa.obj** – obj объектісі (атасы - anketa объектісі); **document.anketa.obj.value** – obj объектісінің **value** қасиеті, ол **<input>** тәгінің **obj** атты **value** атрибутына сәйкес келеді.

value қасиетінің мәнін былай меншіктеп оқуға болады: **var str = document.anketa.obj.value;** Бұл қасиеттің мәнін өзгерте аламыз: **document.anketa.obj.value = "пылеистребитель"** Мұндай меншіктеу экрандағы мәнді бірден өзгертеді. Енгізу өрісіндегі «крошкосшиватель» мәні «пылеистребитель» мәніне ауысады.

Name атрибуты мәні айнымалы аты тәрізді, сондықтан орыс/қазақ әріптерін пайдаланбау керек және бірінші символ цифр болмауы тиіс.

3.1. Экрандық батырмалар

<input> тәгіндегі интерфейстік элементтің нақты бейнесі **type** атрибутымен беріледі. Осы атрибуттың экрандық батырмалар тұрғызатын мәндерін қарастырайық:

reset - шерткен кезде форманы автоматты түрде бастапқы қалыпқа келтіретін батырма жасайды (форманың барлық енгізу өрістері келісім бойынша HTML-кодта орнатылған алғашқы мәндерді қабылдайды). Ал **value** атрибуты батырмадағы жазуды көрсетеді: **<INPUT type=reset value=сброс>** submit - шерткен кезде форманы желі бойынша жөнелтетін батырма тұрғызады. **value** атрибуты батырмадағы жазуды көрсетеді: **<INPUT type=submit value=послать>** button – қарапайым батырма тұрғызады: **<INPUT type=button value="нажми меня">**

Программалық басқару Төмендегі тәг үшін браузер тұрғызған объектің

<input type=button> (не мынадай тәгтер үшін **<INPUT type=submit>**, **<INPUT type=reset>**) **value** қасиеті болады, ол батырмадағы жазуды анықтайды. Төменде осы қасиетті пайдаланатын мысал келтіріліген.

Браузерге осы мысалды енгізіп, ондағы батырмалар жұмысын тексеруге немесе өзгертуге, яғни алгоритмін бақылауға болады.

```

<HTML> <HEAD> <TITLE>Умная кнопка</TITLE> <SCRIPT language=JavaScript>
<!-- var num=0; var name0="число нажатий:0"; //--> </SCRIPT> </HEAD>
<BODY bgcolor=white text=black> <H2>Умная кнопка</H2> <HR> <FORM
name=count> <INPUT type=button value=сброс
onclick="document.count.key.value=name0;
num=0;"> <INPUT type=button
name=key value="число
нажатий: 0"
onclick="document.count.key.value='число
нажатий: '+(++num);"> </FORM> </BODY> </HTML>

```

Форма өрістерін пайдалануды жеңілдету үшін, this нұсқауын - объектінің ағымдағы экземплярларын көрсетуді және this.form нұсқауын - ағымдағы форма үшін тұрғызылған объектіге нұсқауды пайдаланамыз:

```

<FORM> <INPUT type=button value=сброс
onclick="this.form.key.value=name0;
num=0;"> <INPUT type=button name=key value="число
нажатий: 0" onclick="this.value='число
нажатий: '+(++num);"> </FORM>

```

Бұл мысалда форма атын көрсету қажет болмады. Форма ішіндегі оның синонимі ретінде this.form конструкциясы қолданылды. Ал екінші <input> тегінде де оның атын жазбай-ақ, орнына оның синонимі this жазылды.

3.2. Енгізу жолы

<input type=... > тегіндегі type атрибутының text және password мәндері енгізу жолын экранға шығарады: **text** – қарапайым енгізу қатары;

password – пароль енгізу қатары.

Қосымша атрибуттар: · value атрибуты келісім бойынша (по умолчанию) орнатылған енгізу өрісін береді (осы мән reset батырмасын шерткенде орнатылады: <INPUT type=reset>); · size атрибуты өріс енін символмен береді. Егер атрибут көрсетілмесе, ол оның (value) мәні ретінде берілген сөздің ұзындығымен анықталады; · maxlength атрибуты осы өріске енгізілетін сөздің макси-малды енін символмен береді. Егер ол атрибут көрсетілмесе, ұзындығы шектелмейді; · readonly атрибуты «тек оқылатын» сөз тіркесін анықтайды. Мұндай жол түзетілмейді. <INPUT type=text value="Варвара" size=10 maxlength=30>

password – «құпия сөз» енгізу жолы. Ол енгізу кезінде көрінбей, жұлдызшалармен бейнеленеді. Пароль сөз буферге де көшірілмейді.

```

<INPUT type=password value="" size=10
maxlength=10 >

```

Оны программалық басқару Браузердегі <input type=text> (немесе <input type=password>) тегі үшін тұрғызылған объектінің мынадай қасиеттері болады:

value – енгізу жолының ағымдағы мәні;

defaultValue – value атрибутының соған сәйкес <INPUT> тегінде көрсетілген мәні.

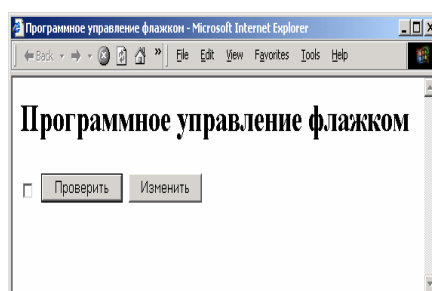
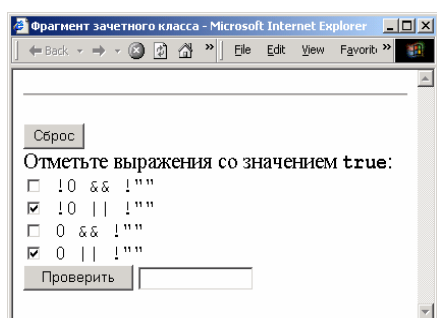
Бұл екеуінің айырмашылығын білу үшін браузерге лаборатория-лық жұмыстарда көрсетілген мысалдарды орындап шығу керек (Студент бумасындағы “Испытатель” программасы).

4. Формалар (жалаушалар, батырмалар)

Енді интерактивті парақтарда жиі қолданылатын екі интерфейстік элементті қарастырайық, олар: жалаушалар мен радиобатырмалар.

4.1. Жалаушалар Жалаушалар немесе тәуелсіз басылатын батырмалар – басқа жалаушаларға байланысты емес, тәуелсіз жұмыс атқарады. Олардың алдындағы қанат белгісін қоюға және алып тастауға болады.

Жалауша жасау тегі: <input type=checkbox>.



<INPUT> тәгінің жалаушаларға арналған қосымша атрибуттары: **name** - <input> тәгі арқылы жасалған объектіні скриптер ішінде пайдалану үшін қолданылатын атау; **checked** - логикалық атрибут, мәні болмайды. Ол тек жалаушаның алдына алдын ала келісім бойынша қанат белгі қойылатынын көрсетеді.

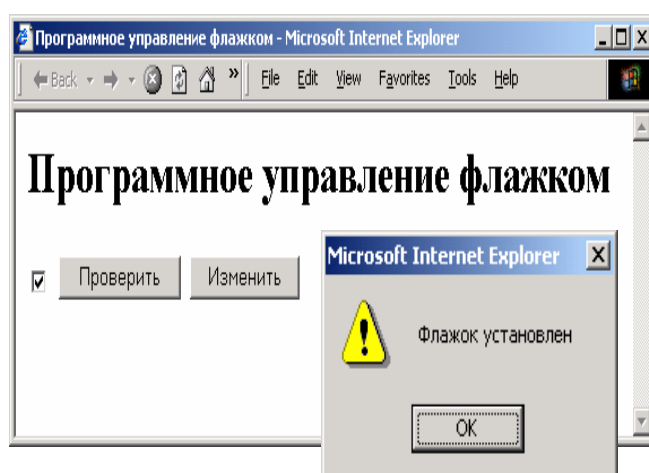
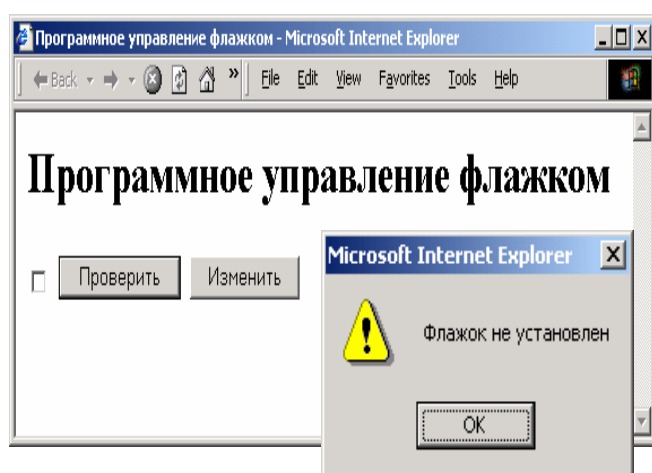
Оны программалық басқару Браузерде мынадай тәгпен тұрғызылған объектінің

<input type=checkbox>checked атты қасиеті болады. Жалауша іске қосылып тұрса, оның мәні – true, әйтпесе мәні – false болады.

Осыларды пайдаланатын төмендегі программаны пайдаланып, оның нәтижелерін де қарастырып шығайық.

```
<HTML> <HEAD> <TITLE>Программное управление флажком</TITLE> </HEAD> <BODY
bgcolor=white text=black> <H2>Программное управление флажком</H2> <FORM>
<INPUT type=checkbox name=flag> <INPUT type=button value=Проверить
onclick="if (this.form.flag.checked) alert ('Флажок
установлен'); else alert ('Флажок не установлен');">
<INPUT type=button value=Изменить
onclick="this.form.flag.checked
!this.form.flag.checked;">
</FORM>
</BODY>
</HTML>
```

Енді мынадай тәжірибелер жасауға болады.



1. Проверить батырмасын шертсек, «Флажок не установлен» мәліметі шығады. Өйткені мұнда checked атрибуты іске қосылмаған:

<INPUT type=checkbox name=flag> <input> тәгі объектісінің checked қасиеті мәні – false, оны this.form.flag.checked өрнегімен де беруге болады (if командасында).

Изменить батырмасын шертсек, мына код жұмыс істейді: this.form.flag.checked=!this.form.flag.checked; Checked қасиеті мәнін өзгертеді (false ® true. Экрандағы жалауша алдында белгі пайда болады, (оны алдын ала қоймасақ та, белгі программа арқылы орнатылды).

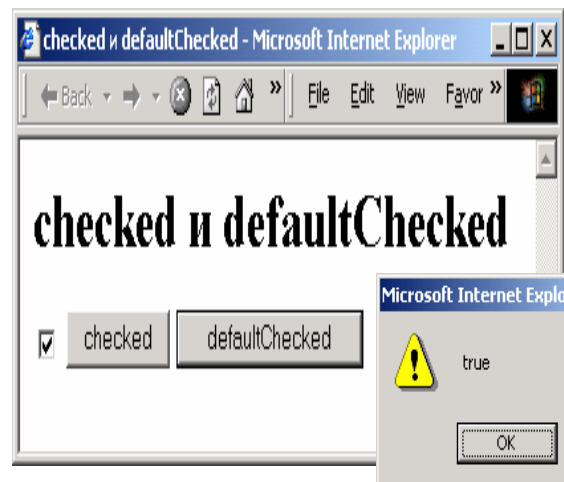
3. Проверить батырмасын шертсек, экранға «Флажок установлен» мәліметі шығады. Енді this.form.flag.checked мәні true болады. Сонымен, checked қасиетін тек оқуға емес, оны өзгертіп мәнін қарама қарсыға ауыстыру программалық жолмен де жүргізіледі екен.

Мұндағы объектінің checked қасиеті <input> тәгіндегі атрибут мәнінен басқаша, онда ол келісім бойынша іске қосылса, мұнда код бойынша іске қосылады. Егер программада <input> тәгінің checked атрибутының мәнін білгіміз келсе, онда сол объектінің defaultChecked мәнін білуіміз керек.

Енді браузерге келесі кодты енгізіп, оның нәтижелерін қарап шығайық.

```
<HTML> <HEAD> <TITLE>checked и defaultChecked</TITLE> </HEAD>
<BODY bgcolor=white text=black> <H2>checked и defaultChecked</H2>
```

```
<FORM>      <INPUT type=checkbox checked name=flag>      <INPUT type=button
value=checked      onclick="alert(this.form.flag.checked) ;">      <INPUT
type=button value=defaultChecked
onclick="alert(this.form.flag.defaultChecked) ;"> </FORM>      </BODY>
</HTML>
```



Келесі тәжірибелерді жасау керек:

1. **checked и defaultChecked** батырмаларын біртіндеп басайық. Сонда «true» және «true» мәліметтерін аламыз. Бұл жалаушаның да, белгінің де қатар орнатылғанын көрсетеді. 2. Жалауша алдындағы белгіні алып тастап, қайтадан **checked** және **defaultChecked** батырмаларын шертеміз. Сонда «false» және «true» мәліметтерін аламыз. Бұл жалаушаның орнатылмағанын, бірақ белгінің алдын ала қойылғанын көрсетеді. 3. `<INPUT type=checkbox checked name=flag>` тәгінен **checked** атрибутын алып тастайық, сонан соң **checked** және **defaultChecked** батырмаларын қайта басайық. Сонда «false» және «false» мәліметтерін аламыз. Бұл жалаушаның орнатылмағанын және белгінің алдын ала да қойылмағанын көрсетеді. 4. Жалауша алдына белгі қойып, **checked и defaultChecked** батырмаларын шертеміз. Сонда «true» және «false» мәліметтерін аламыз. Бұл жалаушаның орнатылғанын және белгінің алдын ала да қойылмағанын көрсетеді.

4.2. Радиобатырмалар

Радиобатырмалар (селекторлық батырмалар, ауыстырғыш) арқылы бірнеше мүмкіндіктің бірін таңдай аламыз. Батырманың бірін белгілесек, қалғандарындағы белгі автоматты түрде жоғалады. (сурет).

Радиобатырма мына тәгпен беріледі: `<input type=radio>` Қосымша атрибуттары келесі Бетте көрсетілген.

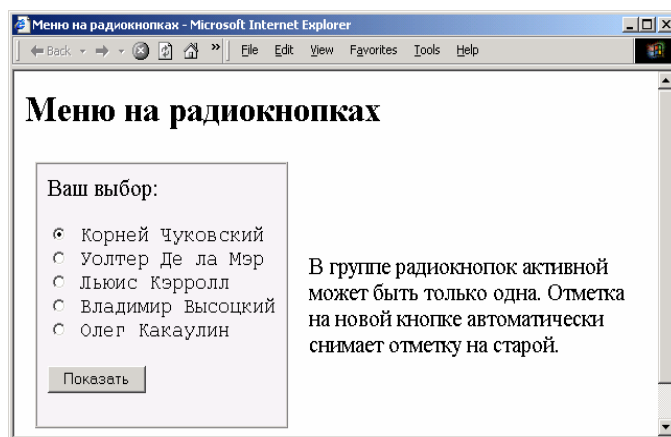
name – атауы; `<input>` тәгі арқылы жасалған объектіні скриптер ішінде пайдалану үшін қолданылатын атау. Бір топқа кіретін батырмалар аты бірдей болуы тиіс. Браузер осы атау бойынша ауыстырғыштарды анықтап, батырмаларды белгілейді немесе басқа батырма таңдалса, белгіле-мейді. Топтағы жеке батырманы анықтау массив элементтерін анықтау сияқты индекс арқылы орындалады. Топтағы батырмалар орналасу кезегі бойынша нөмірленеді, нөмірлеу нөлден басталады.

checked – логикалық атрибут, мәні болмайды. Ол тек батыр-маның алдына алдын ала келісім бойынша таңдалып алынғанын көрсетеді.

Программалық басқару. Браузердің `<INPUT type=radio>` тәгі арқылы тұрғызған объектісінің **checked** қасиеті болады. Ол таңдалып алынса, мәні – **true**, ал таңдалмаса мәні – **false** болады.

Келесі программаны қарастырайық.

```
<HTML>      <HEAD>      <TITLE>Программное      управление      радиокнопкой</TITLE>
</HEAD>      <BODY bgcolor=white text=black>      <H2>Программное      управление
радиокнопкой</H2>      <FORM>      <INPUT type=radio name=tst>      <INPUT type=button
value=Проверить      onclick="if(this.form.tst[0].checked)
alert('Отметка на 1-ой кнопке установлена')      else      alert('Отметка на
1-ой кнопке      не установлена') ;">
      <INPUT      type=button      value=Изменить      onclick="var      x      =
this.form.tst[0].checked;this.form.tst[0].checked=this.form.tst[1].checked;
```

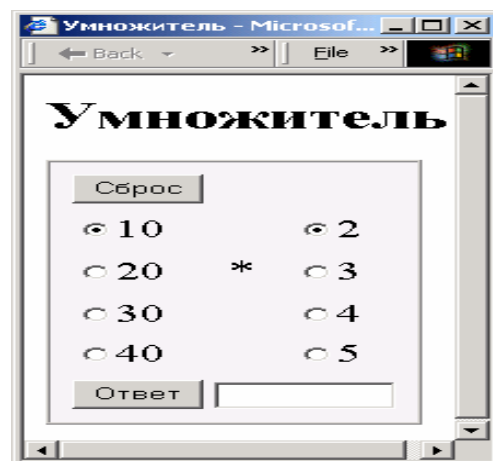
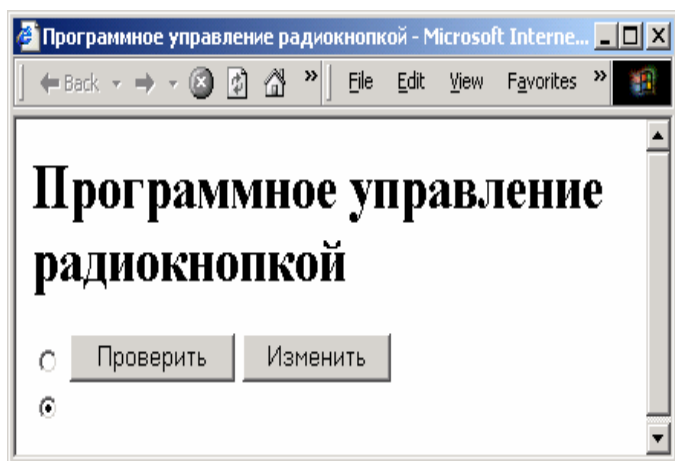


```
this.form.tst[1].checked=x;"><BR> <INPUT type=radio checked name = tst>
</FORM> </BODY> </HTML>
```

Келесі әрекеттерді орындап шығайық.

1. **Проверить** батырмасын шертсек, экранға «Отметка на 1-й кнопке не установлена» мәліметі шығады. Негізінде батырма іске қосылмаған, өйткені оның коды `<INPUT type=radio name=tst>`, атрибуттарына **checked** енгізілмеген. Бұл осы батырма тұрғызған объектінің **checked** қасиеті іске қосылмағанын көрсетеді, сондықтан оның мәні **false**. Егер **if** командасы арқылы **this.form.tst[0].checked** өрнегін пайдалансақ та, нәтижесі осындай болады. 2. **Изменить** батырмасын шертсек, мынадай код орындалады: `onclick="var x = this.form.tst[0].checked;this.form.tst[0].checked=this.form.tst[1].checked;this.form.tst[1].checked=x;"` Бұл код белгіні екінші батырмадан біріншісіне ауыстырады.

3. **Проверить** батырмасын шертсек, «Отметка на 1-й кнопке установлена» мәліметі шығады. Бұл мынадай өрнектің `this.form.tst[0].checked` мәні **true** болатынын білдіреді.



Объектінің **checked** қасиеті `<input>` тәгінің осындай атрибутымен бірдей емес. **Checked** атрибуты батырмаға алдын ала белгі қояды, ал свойство **checked** қасиеті батырманың қазіргі орнатылған мәнін береді. Жалаушалардағыдай **checked** атрибутының **defaultChecked** қасиеті бар.

Бір батырманы іске қосу арқылы қалғандарының алдындағы **checked** арқылы қойылған белгі алынатыны программалаушы есінде болуы тиіс (тек бір батырма белгіленеді).

Мысал. Программасы келесі бетте көрсетілген **Көбейткіш Умножитель** суреттегідей нәтиже береді

```
<HTML> <HEAD> <TITLE>Умножитель</TITLE> </HEAD> <BODY bgcolor=white
text=black> <H2>Умножитель</H2> <FORM> <TABLE border=1
bgcolor=#F0F0F0 cellspacing=0 cellpadding=5> <TR><TD>
<TABLE border=0 cellspacing=0 cellpadding=5> <TR>
<TD colspan=3><INPUT type=reset value=Сброс></TD>
</TR> <TR>
<TD><INPUT type=radio checked name=op1>10</TD> <TD>&nbsp;</TD> <TD><INPUT type=radio
checked name=op2>2</TD> </TR>
<TR> <TD><INPUT type=radio name=op1>20</TD> <TD><BIG>*</BIG></TD>
<TD><INPUT type=radio name=op2>3</TD></TR><TR> <TD><INPUT
name=op1>30</TD> <TD>&nbsp;</TD> <TD><INPUT type=radio
name=op2>4</TD></TR><TR> <TD><INPUT type=radio name=op1>40</TD>
<TD>&nbsp;</TD> <TD><INPUT type=radio name=op2>5</TD></TR><TR> <TD
colspan=3>
<INPUT type=button value=Ответ onclick="var op1=new
Array(10,20,30,40); var op2 = new Array(2,3,4,5); var len =
4; var ind1; var ind2; for(var i=len; --i>=0;)
{ if(this.form.op1[i].checked) ind1=i;
```

```

if(this.form.op2[i].checked) ind2=i;          }
this.form.result.value=
                                op1[ind1]*op2[ind2];">      <INPUT type=text size=10
name=result
                                value="" readonly>      </TD>      </TR>      </TABLE>      </TD></TR>
</TABLE> </BODY></HTML>

```