

**O'ZBEKISTON RESPUBLIKASI ALOQA, AXBOROTLASHTIRISH VA
TELEKOMMUNIKATSIYA TEXNOLOGIYALARI DAVLAT QUMITASI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
NUKUS FILIALI**

Axborot texnologiyalari kafedrası

**Komp`yuter injiniringi fakulteti <<5521900-Informatika va axborot
texnologiyalari>> yo`nalishining 4-kurs talabasi Yuldashev Xolmurodning**

BITIRUV MALAKAVIY ISHI

MAVZU: Flash texnologiyalardan foydalanib ilovalar yaratish

Ilmiy rahbar : _____

Artikov B.

**Qoraqolpog`iston
Respublikasi Elektromagnit
moslashuv xizmati boshlig'i**

Ilmiy maslahatchi: _____

t.f.n. B.Aytmuratov

Kafedra mudiri: _____

t.f.n. T.Arzimbetov

NUKUS -2014

MUNDARIJA

Kirish.....	3
1 bob Macromedia Flash dasturi haqida.	
1.1. Flash texnologiyasi.....	5
1.2. Macromedia Flashda animatsiya yaratish.....	10
1.3 Action Script da obyektlarni yaratish va boshqarish.....	22
Birinchi bob bo'yicha xulosa.....	45
2 bob Flash dasturi yordamida geometrik masalalarni tayyorlash.	
2.1. Planimetriya bo'limiga doir masalalarni tayyorlash.....	46
2.2. Stereometriya bo'limiga doir masalalarni tayyorlash.....	56
Ikkinchi bob bo'yicha xulosa.....	61
Xulosa.....	62
Foydalanilgan adabiyotlar ro'yxati.....	63

Kirish

Respublikamiz Prezidenti I.A.Karimovning 2001 yil Oliy Majlisning 5sessiyasida soʻzlagan nutqida axborot texnologiyalari va kompyuterlarni jamiyat hayotida, kishilarning turmush tarzida, maktab va oila oʻquv yurtlariga jadalik bilan olib kirish gʻoyasi ilgari surilgan edi. Prezidentimiz tashabbusi bilan Oʻzbekiston Respublikasi Vazirlar Mahkamasining 2001 yil 25 mayda qabul qilingan 230-sonli 2001-2005 yillarda kompyuter va axborot texnologiyalarini rivojlantirish „Internetning xalqaro tizimlariga keng kirib borishni taʼminlash“ dasturini ishlab chiqishni tashkil etish chora tadbirlar toʻgʻrisidagi qarori qabul qilindi. Oradan bir yil oʻtgach yaʼni 2002 yil 30 mayda Oʻzbekiston Respublikasi Prezidentining „Kompyuterlashtirishni yanada rivojlantirish va axborot kommunikatsiya texnologiyalarni joriy etish toʻgʻrisida“gi farmonini amalga oshirish yuzasidagi Vazirlar Mahkamasining 2002 yil 6-iyundagi 2002-2010 yillarda kompyuterlashtirish va axborot kommunikatsiya texnologiyalarni rivojlantirish dasturi“ toʻgʻrisidagi qarori eʼlon qilindi. Bulardan koʻrinadi hozirgi kunning dolzarb muammosiga aylangan. Endigi asosiy vazifa axborot texnologiyalari va kompyuterlash boʻyicha toritilgan natijalar umumlashib ularni albatta keng joriy qilishdan iborat. Kompyuterning jamiyat, xalq xoʻjaligi taʼlim-tarbiya sohasidagi ahamiyati beqiyosdir. U xalq taʼlim tizimini maʼmuriy boshqarishdan tortib, alohida maktab, kasb hunar kollejlari, oila taʼlim muassasalari faoliyatini tashkil etish, boshqarish, nazorat qilishgacha, sinf jamoalarida oʻquv predmetlarini oʻrganishni tashkil qilishdan tortib, oʻquvchilarning individual mashgʻulotlarini tashkil etishga boʻlgan muammolarini tashkil etishgacha boʻlgan muammolarni qamrab oladi. Bitiruv malakaviy ishi „Flash texnologiyalardan foydalanib ilovalar yaratish“ga bagʻishlangan.

Bitiruv malakaviy ishning mavzusining dolzarbligi: Zamonaviy oʻquv qoʻllanmalarda koʻrgazmali qurollar yordamida dars oʻtishning samarasi katta. Kompyuter texnologiyalar yordamida darslarni tashkil etishda Macromedia Flash dasturini imkoniyatlarini yoqari.

Tadqiqot obyekti va predmeti: Ushbu bitiruv malakaviy ishda tadqiqot obyekti Macromedia Flash dasturi va geometriya fani hisoblanadi.

Ishning maqsad va vazifalari: Geometriya fanini o'tishda o'quvchilarga tushinarli bo'lishi uchun ko'rgazmali qurollar o'rniga animatsion virtual stend yaratish.

Tadqiqot usuli va uslibiyoti: Bitiruv malakaviy ishda qiziqarli geometrik masalalarni tayyorlashda Macromedia Flash dasturini imkoniyatlarini ko'satish.

Olingan asosiy natijalar: Bitiruv malakaviy ishimda geometriya fanini o'tishda o'quvchilar uchun ko'rgazmali qurollar yaratishda Macromedia Flash dasturining imkoniyatlaridan foydalanib animatsion ko'rinishda multimedia imkoniyatlarini qo'llab virtual stendlar yaratildi.

Ushbu bitiruv malakaviy ishimning birinchi bobida Macromedia Flash dasturida grafik xususiyatlari dasturlarni yaratish, Macromedia Flashda animatsiya yaratish, Montaj usuli va uning, ususiyatlar inspektori, fodalanuvchi interfeysni sozlash haqida ma'lumotlar keltirilgan.

Bitiruv malakaviy ishimning ikkinchi bobida Flash dasturi yordamida geometrik masalalarni tayyorlash usullari keltirilgan.

Mazkur bitiruv malakaviy ishi kirish, ikkita bob, xulosa va foydalanilgan adabiyotlar ro'yxatidan iborat.

1 BOB. MACROMEDIA FLASH DASTURI HAQIDA

Flash texnologiyasi

Flash texnologiyasi Shockwave Flash (SWF) formatidagi vektor grafikadan foydalanishga asoslangan. Albatta bu format eng kuchli formatlardan bulmasada, SWF yaratuvchilariga grafikani imkoniyatlari, grafika bilan ishlovchi vositalar va natijani Web-sahifalarga kushish mexanizmlarini birlashtirish urtasida eng kulay yechimni topishgan . SWF ni kushimcha imkoniyatlardan yana biri bu uning moslashuvchanligidir, ya'ni bu format barcha platformalarda (MacOS sistemali Macintosh kompyuterlari yoki Windows sistemali IBM kompyuterlarida) ishlatilishi mumkin. SWF ning yana bir qulay imkoniyati uning yordamida yaratilgan tasvirlar nafaqat animatsiyali bulishi, balki interaktiv elementlar va tovush bilan boyitilishi mumkin.

Mosalashuvchanlik va interaktiv multimediya dasturlar yaratish imkoniyati SWF formatini Web-dizaynerlar urtasidagi mashxurligini oshishiga imkon berdi. Shuning uchun bu format yaratilishi bilan bir vaqtda Macromedia firmasi tomonidan ikki asosiy tarmoq brouzerlari, Internet Explorer va Netscape Communicatorlar uchun komponentlar (Plug-In) yaratildi. Bu esa, uz navbatida SWF ni Internetda yana ham keng tarkalishiga olib keldi. Natijada ushbu brouzerlar yaratuvchilari SWF formatini o'z dasturlarini asosiy formatlar bazasiga kiritishdi. Bu yulni boshqa yirik dasturiy ta'minot yaratuvchilar (masalan, Adobe firmasi) ham tutishdi. Yana bir bor aytish kerakki: agar Macromedia SWF formatini juda oddiy va kulay instrumentlar bilan ta'minlamaganda bu format shunchalik kup muxlis orttirmagan bulardi. Shuni aytish kerakki, hozirgi vaqtda ushbu instrumentlarni bir qancha pulik tuplamlari mavjud.

Ushbu instrumentlarni bir turi (Macromedia Director Shockwave Studio) multimediya prezentatsiyalar yaratish, boshqalari (Macromedia FreeHand i Macromedia Fireworks) grafik tasvirlar, uchinchilari esa (Macromedia Authorware i Macromedia CourseBuilder) interaktiv urgatuvchi kurslarni yaratish imkoniyatini beradi. Lekin Web-yaratuvchilar orasida eng kup ishlatiladigani bu Macromedia Flash dir, chunki ushbu dastur xar qanday saytga mashxurlik olib keluvchi Web-sahifalar yaratish imkonini beardi. Balki shuning uchundir

SWF formatini oddiy qilib Flash deb atalish odatga kirib qolgandir. Shunday qilib Flash-texnologiyalar quyidagilardan iborat:

- vektorli grafika;
- animatsiyani bir qancha turlarida ishlash imkoniyati;
- interfeysni interaktiv elementlarini yaratish imkoniyati;
- sinxron ovoz qushish imkoniyati;
- HTML formatli va boshqa Internetda ishlatiluvchi barcha formatlarga eksportini ta'minlash;
- platformali mustaqillik;
- Flash-filmlarni avtonom rejimda ham ,Web-brouzer yordamida ham ko'rishni imkoniyati mavjud;
- vizual yaratish instrumentlari mavjudligi Flash-film yaratuvchilarini ko'plab murakkab operatsiyalardan halos etadi, shuningdek Flash-texnologiyalarning texnik aspektlarni o'rganish zarur bo'lmaydi.

Asosiy tushunchalar

Xar bir Flash-filmni yaratishda ob'ektga yo'naltirilgan yo'ndashuv qo'llaniladi. Bu, filmni barcha elementlari u yoki bu tipdagi ob'ektlar sifatida interpretatsiya qilinishini bildiradi. Bu ob'ektlarni har biri uchun xususiyatlari va ruxsat etilgan operatsiyalar tuplami berilgandir. Masalan, «Matn» ob'ekti uchun simvollar razmeri, rangi va boshqa parametrlar o'rnatilgan bo'lishi kerak. Matnni usulda taxrirlash, kirkish, nusxasini olish, matnni asosida giperssilklar yaratish va boshq. Bu gapni tovush va grafik tasvirlar xususida ham aytish mumkin. Shunday bo'lsa ham Flashda ishlanganda «ob'ekt» tushunchasi o'rniga simvol (Symbol) terminii ishlatiladi. Ularning asosiy farqi quyidagicha:

Simvol- bu ob'ektni o'ziga xos xususiyatlarini tuplamiga ega shablonidir. Simvollar maxsus kutubxona (Library) da saqlanadi. Bu kutubxonadagi simvollar ko'p marta, ham bir filmda ham bir necha filmlarda bir vaqtni o'zida ishlatilishi mumkin. Filmga joylashtirilgan simvolni xar bir yangi nusxasi simolni ekzempliyari (Instance) deb ataladi. Ekzempliyar simvolning o'zini barcha xususiyatlarini meros qilib oladi va ular orasida boglanish xosil bo'ladi: simvolni xususiyatlarini o'zgartirilganda avtomatik ravishda barcha ekzempliyalar xususiyatlari ham o'zgaradi. Bunday yo'ndashuv filmlar yaratishni

vaqtini kamaytiradi. Bundan tashkari, simvollar qollash mexanizmi filmni razmerini kamayishiga ham olib keladi: filmda simvolni bir necha ekzemplari ishlatilsa, o'ning har bir ekzemplari xususiyati tugrisidagi ma'lumot inobatga olinmaydi. Shu bilan birga, siz konkret ekzemplarni xususiyatlarini o'zgartirishingiz mumkin, bu simvol- originalga ta'sir ko'rsatmaydi. Masalan, ekzemplarni razmeri va rangini o'zgartirish mumkin, agar tovush haqida gap borsa-u yoki bu effektni qo'shish.

Flash-filmlardagi dinamika ekzemplarni u yoki bu xususiyatlari (masalan, koordinatalari, rangi , razmeri va boshq.) ma'lum bir vaqt oraligida o'zgarishi hisobiga ta'minlanadi, ya'ni ekzemplar holati o'zgaradi. Ekzemplarning har bir holati bilan filmni alohida bir kadri (Frame) bog'liqdir. Ekzemplarni holatini o'zgarishiga tug'ri keluvchi kadr asosiy (klyuchevoy) kadrdir (Keyframe). Asosiy kadrning o'zi ham alohida bir ob'ekt sifatida ko'riladi. Asosiy kadr uchun maxsus funktsiya va buyruqlar nazarda tutilgan.

Film kadrlari almashish dinamikasi vaqt diagrammasi (Timeline) yordamida ta'riflanadi. Vaqt diagrammasini parametrlari sifatida kadrlar almashish chastotasi, ob'ekt xarakatini boshlanish va yakunlashish va boshqa parametrlarni ko'rsatish mumkin. Filmda bir necha turli xil ob'ektlardan foydalanish mumkin. Har birining xolati boshqasidan mustaqil ravishda o'zgarishii yoki umuman o'zgarmay qolishi (masalan, biror-bir ob'ekt fon sifatida ishlatilsa) mumkin. Filmni turli elementlarini xarakatini ta'riflashni osonlashtirish uchun, ularni har birini alohida qavat (Layer-sloy) da joylashtiriladi. Flash yaratuvchilari filmdagi qavatlar rolini tushuntirish uchun ularni yupka kalka bilan solishtirishadi. Bunday varaqlarni bir nechasini bir to'plamga yig'lsa ma'lum bir saxna namoyon bo'ladi. Saxna (Scene)- bu Flashni yana bir terminlaridan biridir. Xar bir saxna kavatlarini ma'lum bir to'plamidan iboratdir. Sodda filmlar uchun bir qavatdan iborat saxnani yaratish va ta'riflash kifoyadir. Murakkab filmlar uchun bir necha turli ko'rinishdagi saxnalar yaratish kerak bo'ladi. Bir saxnadan ikkinchisiga o'tish vaqt diagrammasi emas balki boshqa bir mexanizmlardan foydalanib amalga oshiriladi. Oddiy holda, film saxnalari ma'lum ketma- ketlikda, ularni tartib raqami bo'yicha bajariladi. Filmni murakkabrok usulda tuzishda ActionScript tilidan foydalaniladi. Murakkab filmlarni yaratishda muhim rolni yana bir tushuncha- klipdir (Clip, yoki Movie clip). Klip- bu simvolni maxsus turidir. U o'ziga xos mini- filmdir, bu film uchun xususiy vaqt diagrammasi va xususiy parametrlar (masalan, kadrlar almashshish

chastotasi) o'rnatiladi. Klipdan ko'p marotaba foydalanish uchun filmni boshqa bir elementlari singari kutubxonaga joylashtirish mumkin. Klipni har bir ekzemplari uchun o'ziga xos xususiy nom beriladi. Filmni har bir elementi klipni ichida ham ishlatilishi mumkin. Ichma- ich joylashgan kliplar ham yaratish mumkin. Agar film ichidagi klipni aktivlashtirishni ba'zi quchimcha shartlarini berish uchun ActionScript tili vositalaridan foydalanish mumkin. Klipga interaktiv elementlarni(masalan, knopka) ham qushish mumkin. Flash-filmlar yaratish ketma- ketligi Windows-dasturlarni ko'pchiligini ishga tushirganda ushbu dastur bilan ishlash uchun «zagotovka» (bush xujjat, blank) avtomatik ravishda yaratiladi. Masalan Word muxarririni oynasini ochilganda unda bush xujjat ochiladi, bu xujjatga siz matn kirtishingiz, grafika kushish , shrift kurinishi o'zgartirish va boshk. Amalga oshirishingiz mumkin. Yangi xujjat uchun urnatilgan parametrlar(shrift razmeri va stili, maydon kengligi va boshq.) qo'llaniladi. Xujjatingiz sizga ma'qul bo'lgan ko'rinishda bo'lishi uchun parametrlarni kerakli ko'rsatkichlariin o'rnatib qo'yish talab qilinadi.

Flash muharririda ham ish shu usulda tashkil etilgan.Flash ishga tushirilganda avtomatik ravishda bo'lajak film uchun «zagotovka»(aniqrok, uning birinchi saxnasi) yaratiladi. Agar lozim bo'lsa, muallif filmning umumiy xossalariin o'ziga ma'qul bo'lgan kursatkichlarini o'rnatish mumkin, bu kursatkichlarga masalan, kadrlar almashish chastotasi, oynalar razmeri, fon rangi va boshqalar. Filmning umumiy parametrlarini o'rnatilgandan so'ng filmni birinchi saxnasini yaratishga o'tsa ham bo'ladi. Ob'ektlarni kay biri saxnada ishtirok etishni xal qilgandan so'ng, siz ketma-ketlikda xar bir ob'ektni yangi kavatga joylashtirasiz. Qavat parametrlarini o'rnatish-bu filmni yaratishni ma'suliyatli bosqichidir. Filmdagi qavatlar(yoki ularda joylashtirilgan ob'ektlar) bir- biri bilan aloqa qilganda, navbatdagi qavat yaratilgandan so'ng avval yaratilganini tahrirlash lozim bo'lib qoladi. Bu unchalik ham qiyin ish emas, Flash muallifga avvalgi xarakatlarini o'zgartirish imkonini beradi. Filmga kiritiladigan navbatdagi ob'ekt yo simvollar kutubxonasidan olinishi mumkin yoki yangitdan yaratilishi mumkin. Agar ob'ekt hozircha faqat hayolingizda yoki qo'gozda bo'lsa, Flash chizish uchun muljallangan barcha instrumentlarga ega bo'lgan grafik muxarrirga aylanadi. Shu bilan birga, yaratiladigan ob'ektni alohida komponentlari yoki tugatilgan tasvirlar boshqa bir grafik formatlardan import qilinishi mumkin. Agar biror bir ob'ekt filmda bir necha marotaba ishlatilsa, u holda ularni yangi simvol sifatida yaratish maqsadga muvofikdir. Flash juda

qulay interfeysga ega bo'lsa ham, Flash-filmlarni yaratish juda murakab ishdir. Shuning uchun film yaratishni xar bir kadamidan sung natijani tekshirish lozim. Flash filmni aloqida saxnasini ham butun filmni tekshirish imkoniyatini beradi. Agar tekshirish muvafakkiyatli utgan bo'lsa, natijani diskda saqlashni unutmang, chunki Flash da avtomatik saqlash funktsiyasi mavjud emas. Filmni ustida ishlab bulingandan sung, u Web-sahifani bir qismiga aylanishini esdan chiqarmasalik kerak. Har bir sahifani eng asosiy kamchiligi nimada? Tug'ri, yuklanishni juda ham ko'p vaqt mobaynida utishidadir. Bundan kelib chiqib, Flash yaratuvchilari siz filmni yuklanish vaqtini baholashingiz uchun imkoniyat yaratgandirlar. Ushbu ma'lumot filmni xar bir kadri uchun ham son ham grafik ko'rinishda olinish mumkin.

Grafik dasturlarni yaratish

Hozirgi kunda Web sahifalarni yaratishda oldingi o'rinlardan biri rastrlı grafika hisoblanadi. Rastrlı formatlardan GIF (graphics interchange format ma'lumotlarini almashinuvi uchun grafik format). JPEG (Join Photografic Experts Graup Tasvirlar bo'yicha mutaxassislarni birlashgan guruhi) va PNG (Portable network Graphics ko'chirma grafik format) va boshqa formatlarni keltirish mumkin.

Rastrlı grafikani ishlatishda tasvir nuqtalar majmuasi dan iborat bo'ladi. Bu nuqtalar biri biri bilan bog'liq bo'lmaganligi uchun ushbu nuqtalar har biriga rangini va koordinatasi berilishi kerak. Oddiy holda agar ikki xil rangli tasvir ishlatilsa u holda har bir pikselni ta'riflash uchun har pikselga bunday razryadlardan 8ta kerak bo'ladi. Juda ham murakkab fotorealistik rangli tasvirlar 1 pikseliga 24 razryad talab qiladi. Rastrlı tasvirlarni yana bir kamchiligi shundan iboratki ,tasvir sifati piksel o'lchamiga bo'liq, u esa o'z navbatida monitorni imkoniyati bilan belgilanadi. Shuning uchun bir xil rasm turli monitorda har xil ko'rinishga ega bo'lishi mumkin. Bu ham hali hammasi emas. Rastrlı tasvirni o'lchamini o'zgartirish juda ham murakkab. Chunki bunday tasvirni kattalashtirish pikseller sonini o'sishiga olib keladi. Kompyuter grafikasi sohasidagi mutassislar tomonidan juda murakkab rastrlı tasvirlar pikseller "ko'paytirushy" yoki o'chirish algaritmlari ishlab chiqilgan, lekin ular doim ham ushbu masala bajara olmaydi.

Grafikaning yana bir turi bu vektorli grafikadir. Vektorli grafika bu tasvirni rasmda joylashuvi matematik formulalar bilan berilgan egri chizialar majmuasi yordamida namoyish etish usulidir. Masalan istalgan doirani tasvirlash uchun uch-to'rt raqam kerak bo'ladi. Radius markazi koordinatalari va chiziq qalinligi. Shuning uchun vektorli grafika rastrli grafikaga nisbatan bir qancha qulayliklarga ega:

- Vektorli tasvirlarni belgilovchi matematik formulalar kompyutr xotirasida rastrli tasvir pixsellarga qaraganda kamroq joy egallaydi.

- tasvir (yoki uning ayrim qismlarini) sifatini yo'qotmasdan chegaralanmagankattalashtirish imkoniyati mavjudligi.

- tasvirni bir platformadan ikkinchisigako'chirishning qulayligi .

Albatta vektorli tasvirni o'z kamchiligi ham mavjud. Masalasn, fotorelastik tasvirni vektorli formatdan namoyish qilish murakkabroq. Flash yaratuvchilari bunga ham yechim topishgan. Flash dasturi yordamida Web sahifalar tuzishda siz nafaqat vektorli balki rastrli tasvirlarni ishlatishingiz ham mumkin.

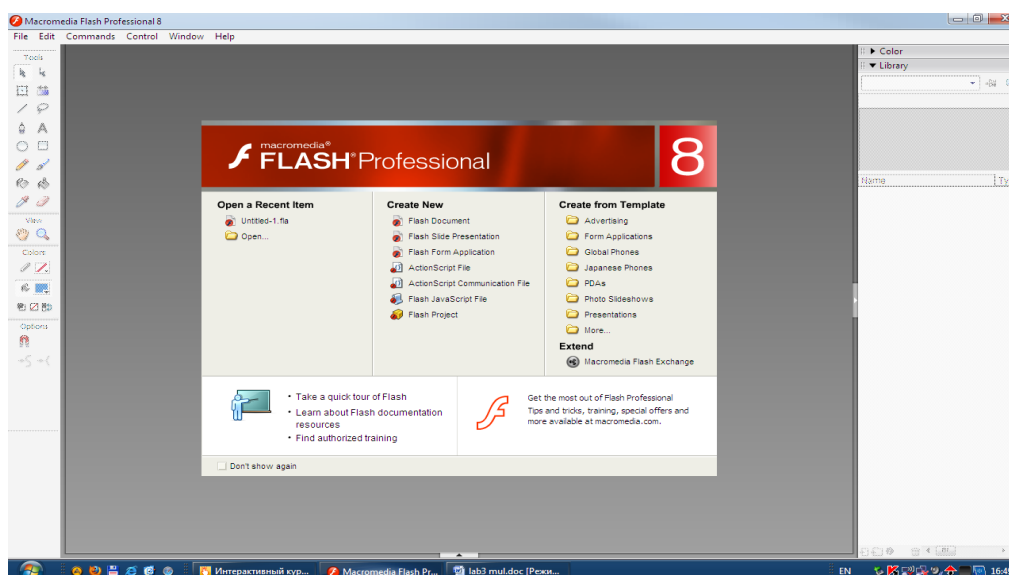
1.2 Macromedia Flashda animatsiya yaratish.

Flash dasturida animatsiya ikki xil bo'ladi: kadrli va avtomatik. Avtomatik animatsiya shakllar geometriyasini o'zgarishi yoki boshqaruv kadrlar o'zgarishi asosidagi animatsiya turlariga bo'linadi. Bu turdagi animatsiyani yaratish uchun biz bitta boshqaruv kadri yaratamiz va unga belgi qo'yamiz. Masalan boshqaruv kadrda aylana chiziladi. Va u grafik tasvir belgi turiga F8 yoki Vstavka menyusida Preobrazavot v simvol buyrug' yordamida o'tkaziladi. Yoki ctrl+F8 buyrug'ini tanlab yangi belgi yaratamiz va belgilar kutubxonasi yordamida uni boshqaruv kadrda qo'shamiz.

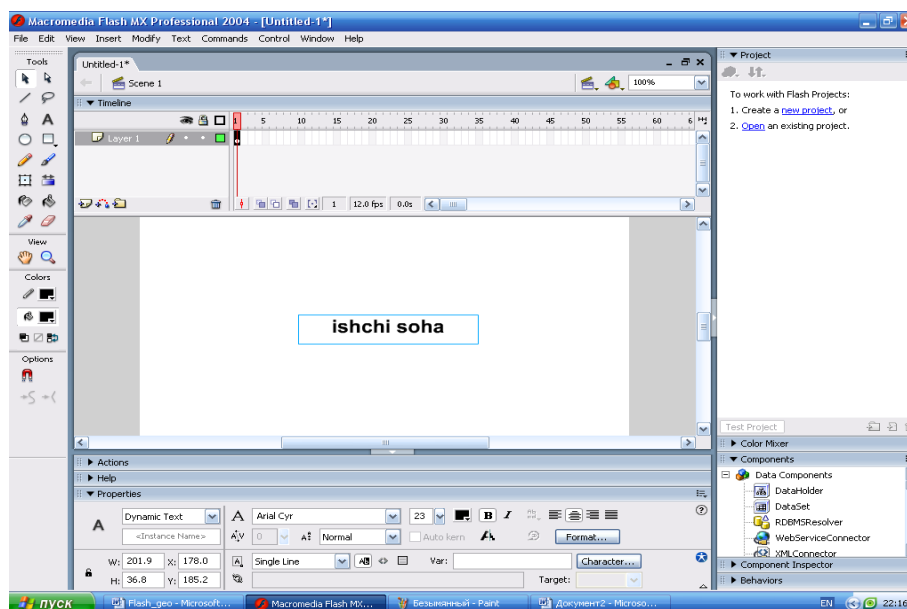
Macromedia Flashda ishni osonlashtirish va rolik o'lchamlari yaxshilash uchun simvol tushinchasi kiritilgan. Simvol quyidagicha bolishi mumkin. Grafika (Graphic), tugma (button), animatsion rolik (movie clip), tovush(sound). Bu imkoniyat yaratilgan obyektlar bir necha marta foydalanish imkonini beradi.

Endi belgi joylashgan boshqaruv kadri sichqonchani o'ng tomonidagi tugma yordamida tanlab Great motion twening yoki vstavit menyusining shu nomli buyrug'ini

tanlaymiz. Shu harakatlar natijasida boshqaruv kadr rangi ko'k rangga o'zgaradi. Endi sichqoncha bilan yangi kadrni tanlaymiz va F6 ni buyrug'ini tanlaymiz. Natijada kadrda ko'k rangli boshqaruv kadr hosil qilinadi. Birinchi boshqaruv kadrda ikkinchi boshqaruv kadr gacha kadrlar ko'k rangda avtomatik hosil qilinadi. Oxirgi harakatimiz bu ikkinchi boshqaruv kadrda belgini o'zgartirish (cho'zish, aylantirish, kattalashtirish yoki kadrda joylashishini o'zgartirish). Endi klaviaturadagi Enter tugmasini bosamiz va biz yaratgan animatsiyani ko'rishimiz mumkin. Bu turdagi animatsiyani yaratish uchun biz boshqaruv kadrlar o'zgarishi asosidagi yaratilgan animatsiya hosil qilamiz. Faqat endi oxirida ikkinchi qatlamdagi belgini butunlay o'chirib uning o'rniga kvadrat chizamiz. Shu harakatimizdan keyin kadrlar rangi normal rangga qaytadi. Keyin birinchi va ikkinchi boshqaruv kadrlaridagi grafik tasvir belgisini ctrl+F8 yoki Izmenet menyusidagi Razidelet oldelno buyrug'I yordamida aloda shakllarga bo'lib chiqamiz. (2-rasm).



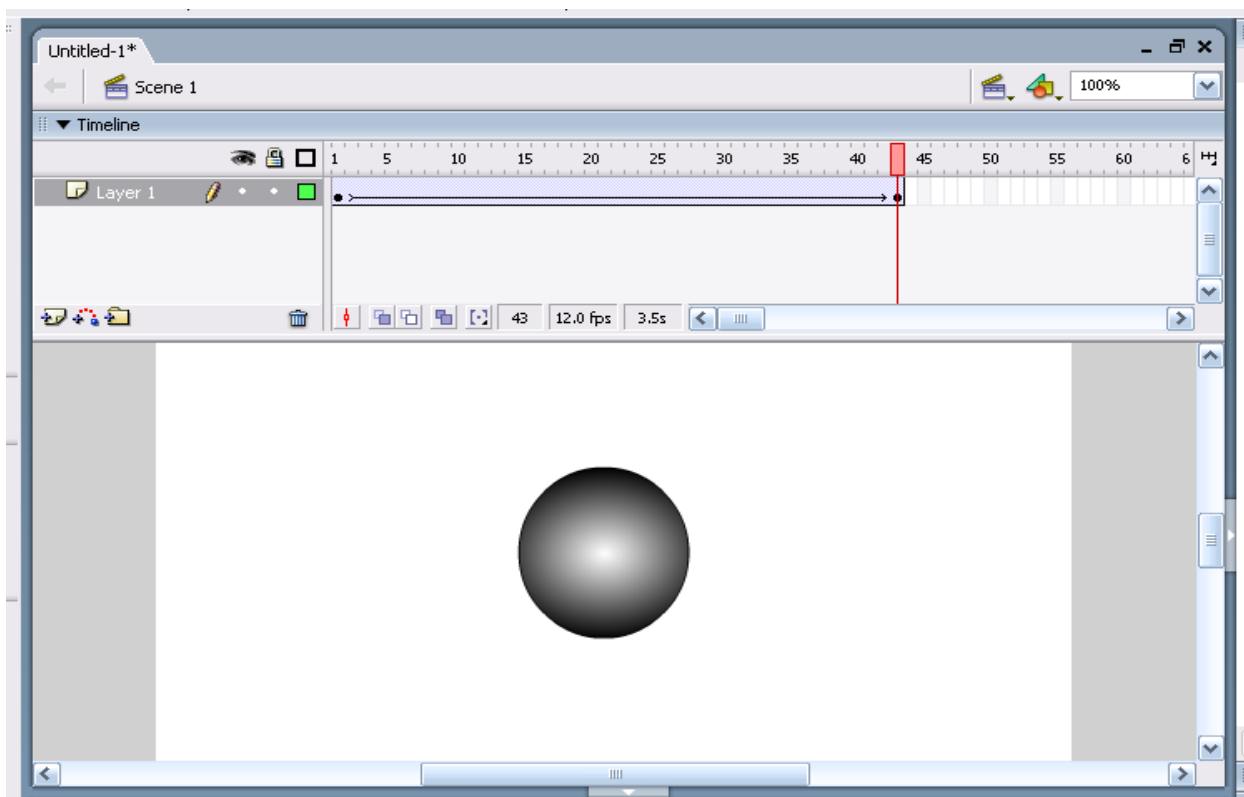
1-rasm. Macromedia Flash dasturi.



2-rasm. Ishchi soha

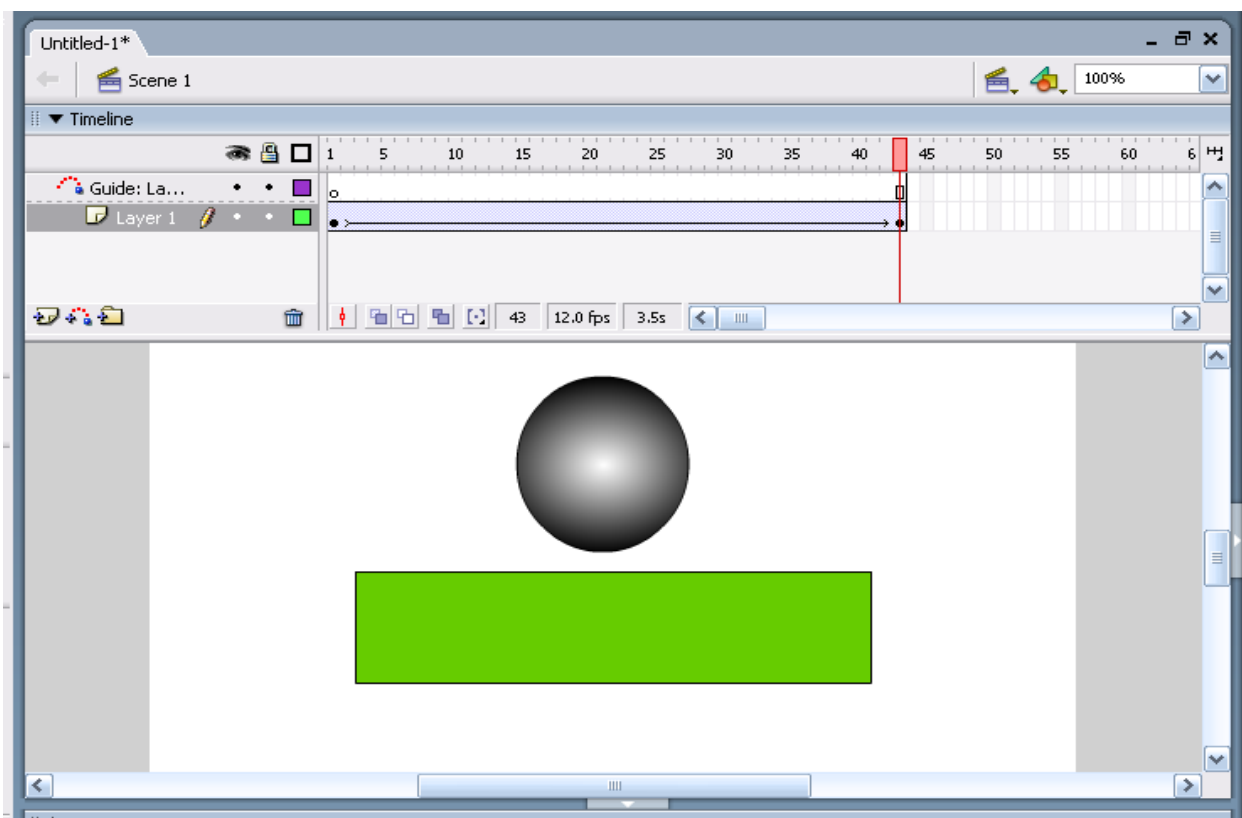
Endi avval birinchi boshqaruv sichqoncha chap tugmasi bilan tanlab stoyanka(Propertill)yoki ctrl+F3 yoki Okno menyusini shu nomli buyrug'ini tanlaymiz.Natijada muloqot oynasi hosil qilinadi va unda Tweening sohasida Motions o'rniga Shape holatini tanlaymiz.Natijada boshqaruv kadr rangi yashil ranga o'garadi.Endi ikkinchi boshqaruv kadrni ham sichqoncha bilan tanlab animatsiya turini Motiondan Shapega o'zgartiramiz va oxirida Enter tugmasini bosamiz.Va qarayotgan animatsiyani ko'rishimiz mumkin.Bunda yopiq doirani pleerda harakatga kelish animatsiyasini ko'ramiz.

Ko'rishimiz mumkin ikkila animatsiyalarda ham ikkita boshqarish kadrlar o'rtasidagi masofa kadrlarni kompyutr o'zi avtomatik yaratgan va animatsiya to'g'ri chiziq bo'yicha harakatlanadi.Ammo bizga harakat traektoriyasi bo'yicha bajarilishi kerak bo'lsa,u hold anima qilish kerak?Bunday animatsiyalarni hosil qilish uchun bizning qatlamimiz ustida maxsus harakat traektoriya qatlamini yaratish kerak bo'ladi.Va shu qatlamda qalam bilan qiyshiq traektoriya chizig'ini chizamiz.Natijani ko'rish uchun Enter tugmasini bosamiz.(3-rasm).



3-rasm. kadr hosil qilish.

Insert Layer buyrug'i yordamida yangi qatlam yaratamiz va Pencil instrumentini faollashtiramiz. Yaratilgan chiziqni sakrayotga koptok uchun yo'l sifatida foydalanamiz. Chiziqni tanlaymiz va Edit copy (ctrl+c) buyrug'i yordamida buferga olamiz. Shu bilan birga avtomatik animatsiyalarda foydalanish mumkin bo'lgan yana bir effekt mavjud bu-mask qatlamdir. Mask qatlami uchun asosiy qatlam ustida yangi bo'sh qatlam yaratmiz. Shu qatlamning sichqonchani chap tugmasi bilan bosib Mask mask buyrug'ini tanlaymiz. Natijada qatlam ko'k rangga o'zgaradi. Vaikkila qatamlar o'zgarishlardan himoyalaniadi. Mask qatlamdan shu himoyani o'chirib boshqaruv kadrda bir nechta to'rtburchaklar Enter tugmasi bosamiz va natijani ko'ramiz. (4-rasm)



4-rasm. animatsiyani hosil qilish.

Usul obyekt xususiyatlarni boshqarish uchun xizmat qiladi. Eng oddiy foydalanish sifatida koordinata o'zgarishlarini keltirishimiz mumkin. Yangi fayl yaratamiz vaaylana chizamiz. Paint Bucket(u) eksterumenti yordamida oq-qora gradient bilan uning ichini bo'yaymiz. Maska qatlamdagi boshqaruv kadrni Motion tweening animatsiyalahtirilsa harakatlantirish effekti yanada chiroyli bo'ladi. Flashdagi animatsiya "multik"da ishlatiladigan obektlarni xossalarini o'zgartirishga asoslangan. Masalan obekt yo'qolishi yoki paydo bo'lishi ,joylashuvini rangini va boshqalarni ko'rishimiz mumkin. Flashda obyektlarni animatsiyalashni 3 ta turli mexanizmi ko'zda tutilgan:

Kadrma kadrli ("klassik") animatsiya bu usulda muallif bo'lg'usi "multik"ni har bir kadrni o'zi yaratadi yoki boshqa joydan import qiladi. Namoyish ketma ketligini o'zi yaratadi. Avtomayikanimatsiya (tweened animatsiya)bu usulda muallif multifatsiya kadrlarini faqat birinchi va oxirgisini yaratadi. Otraliq kadrlarni esa Flash avtomatik tarzda o'zi yaratadi. Tweened animatsiyani ikki xil turi mavjud :obyektni ko'chirishga asoslangan animatsiya va transformatsiyasiga asoslangan animatsiya. Snariyga asoslangan animatsiya ,Snariy Flashning dasturlash tilida(bu til Action Script)deb nomlanadi. Obyektni hatti-

harakatini berilishidir. Bu tilni sintaksini Web hujjatlarda ishlatiladigan boshqa senariy tillariga (Java Skript va VB Script) o'xshab ketadi. Ushbu mexanizmlarni har biri o'zining afzalliklari va kamchiliklariga ega. Masalan tweened animatsiya ikki afzalikka ega:

- birinchidan muallif har bir kadrni alohida yaratish zarurati kutilgan.

- ikkinchidan bunday "multik"ni tashkil etish uchun Flashda faqat birinchi va so'ngi kadrni saqlash kifoya. Bu esa o'z navbatida bunday filmni hajmini kam bo'lishini ta'minlaydi.

Shu bilan birga tweened-animatsiyani faqat kadrlar interaktiv bo'lishi mumkin. Shu bilan birga ko'pgina standart amallar filmini biror bir elementiga Action Scriptda dasturlasgandan foydalanmay berilishi mumkin. Standart amallarga masalan filmni namoyishini boshlash va to'xtatish ovozi yoqish yoki o'chirish va boshqalar. Hammasi bo'lib standart amallarni o'nga yaqin turi mavjud.

Agar zarurat tug'ilsa ham murakkab snariylar tuzilishi mumkin. Masalan u yoki bu holat yuz berishi, so'ngra kerakli amal bajarilishiga o'tish. Lekin bunday snariylarni yoqish yoki o'chirish yaratish dasturlashdagi bilimlarini boshlash va to'xtatish kerak bo'ladi. Flash filmlarni ishlatilish bir necha usullarni tadim etadi. Siz filmni syujeti va foydalanuvchi bajargan amaldan qat'iy nazar to'xtamaydigan amalni yaratishingiz mumkin. Alternativ variant animatsiyava tovushni sinxronlash. Bundan tashqari Flash tugmachalariga tovush ishlatishingiz mumkin. Bu esa tugmalarni interaktiv qiladi. Tovush bilan ishlashni yana bor varianti - Action Script snariylardan foydalanib tovushni boshqarish. Flashda tovushni qo'shishni ikki xil mavjud. Xodisaviy boshqariladigan (event sound) va oqimli (stream sound). Flash tovush bilan ishlashda yaratuvchilar uchun quyidagi imkoniyatlarni yaratadi:

- bo'linuvchan tovush kutubxonalari yaratish yoki bir kutubxonadagi tovushni bir necha filmlarda foydalanish imkoniyatidir.

- tovush sifati va eksporti qilinayotgan filmni optimal birlashuvini tanlash maqsadida tovushli ma'lumotlarni qo'shish ko'rsatgichlarini boshqarish muallif alohida tovush yoki filmdagi barcha tovushlar uchun qo'shish ko'rsatgichini tanlashi mumkin. Bundan tashqari siz tovushni o'zgartirish uchun turli qo'shimcha effektlardan foydalanishingiz mumkin.

Flash filmlarni ishlatish variantlari.

Yuqorida takidlab o'tilgandek Flashning asosiy vazifasi grafika va animatsiya bo'lgan interaktiv Web-sahifalarni yaratishdan iboratdir. Flash yaratuvchilari uning yordamida olingan filmlarni ishlatilishini yana ikki boshqa variantlarini ham ishlab chiqishgan. Birinchidan Flash filmlarini mustaqil (Web browserda bog'liq bo'lmagan holda) namoyish etilishi, ikkinchisi esa Flash filmlarini boshqa formatlariga o'tkazilishidir. Filmlarni flash yordamida yaratish va taxrirlash vaqtida muallif FLA formatiga ega fayllar bilan ishlaydi. Bu Flash muxarririni ichki (shaxsiy) formatidir va u faqat Flashda tushinarlidir. Web-browser yordamida namoyish etish uchun mo'jallangan filmlar SWF formatiga o'tkazib olish zarur. Yuqorida ta'kidlab o'tilgandek, ushbu format browserlari (Internet Explorer, Netcape va opera) so'ngi versiyalarida namoyish etilishi mumkin. Vayomimaxsus Flash pleer yordamida ham namoyish etilishi mumkin. Ushbu Flash pleer Flash muxariri tarkibiga mustaqil dastur sifatida kiradi va SWF fayllar muxaririsiz ham Flash pleerda namoyish etilishi mumkin. Shu bilan birga filmni barcha funksional va interaktiv imkoniyatlar saqlanib qoladi.

Flash filmlarini "avtonom" namoyish etilishi ikkinchi variant bu-Universal Media pleerdan foydalanishdir. Buning uchun Flash film AVI formatida eksport qilinishi kekrak. Lekin bu holda filmni hajmi ayrim hollarda yuz barobar ortib ketishi mumkin.

Macromedia Flashda ishni osonlashtirish va rolik o'lchamlarini yaxshilash uchun simvol tushinchasi kiritilgan. Simvol quyidagicha bo'lishi mumkin. Grafika (Graphic), tugma (button), animatsion rolik (movie clip), tovush (sound). Bu imkoniyat yatarilgan obyektlar bir necha marta foydalanish imkonini beradi. Flash filmlarini AVI formatiga o'tkazikishi ularning interaktivligini yo'qolishiga olib keladi. Shuning uchun ushbu variant keng tarqalmagandir. Flash filmlarini namoyish etishni yana bir turi mavjuddir. Bu variant Flash filmni animatsiyalashgan GFI formati ga o'tkazishdir. Bu holda eksportni ikki turi mavjud : har bir foydali animatsiyani bir kadr ga mos keluvchi GFI fayllar to'plami ko'rinishida yoki bir animatsiyalangan fayl ko'rinishidadir. Ekportning ikkinchi varianti afzalroqdir. Cuunki boshqa bir dastur yordamida kadrni "yig'ishga" qaraganda bu variantga fayl hajmi kichikroq bo'ladi.

Dasturlash tillarida obyekt faqat shu obyektga qo'llanishi mumkin bo'lgan atributlar (xususiyatlar) to'plami va usullar ro'yxati bilan ifodalanadi. Har bir sinf obyektlari uchun o'z

atributlar va usullar to'plami belgilangan. Masalan "tugma" obyekti uchun atributlar sifatida ishlatilishi mumkin.

Flash filmlarni grafik formatlarga o'tkazish.

Yuqorida sanab o'tilgan Flash filmlarni eksportiularning dinamikasini (u yoki bu darajada) saqlanishini taminlaydi. Lekin zarur bo'lganda Flashdan oddiy grafik muxariri sifatida ham firdalanish mumkin. Uning yordamida yaratilgan tasvir so'ng biror bir grafik format (vektorli yoki rastrli)ga o'tkazilishi mumkin. Shu bilan Flash filmga boshqa grafik formatlardan berilgan tasvirlarni qo'shish(import qilish) imkoniyatiga ega.

Action Script da Button va Movie clip sinflari obyektlari orasida bo'ysinish munosabatlarini aniqlash juda qiyin: ular bir filmdan ikkinchisiga o'tganda va hatto saxnadan saxnaga o'tganda ham o'zgarishi mumkin. Masalan bir vsaxnada tugma klip tugmani "jonlantirish" uchun ishlatiladi Movee sinfini boshqa obyektlarini (text field) dan boshqa button va Movee clip obyektlariga nisbatan bo'ysingan deyish mumkin. Chunki ular Button va Movee clip obyektlari tarkibiga kirishi mumkin lekin aksincha emas. E'tibor bering Action Scriptda kadrlar uchun ham, film saxnasi uchun ham mos obyektlari sinfi yaratilmagan. Gap shundaki Action Script filmining ushbu elementlari xususiyatlarini o'zgartirishga yo'l qo'ymaydi. Siz snayriy yordamida faqatgina kadrqa nomi yoki raqami orqali murojaat qilishingiz, shuningdek uning yuklangan va yuklanmaganligi.

Web-sahifalar yaratish.

Flash filmlarni namoyish etilishini turli usullari majjudligiga qaramasdan bugungi kunda bu texnologitani qo'llanishining asosiy yo'nalishi bu chiroyli va dinamik Web sahifalar yaratishdir. Flash filmni brouzerda-namoyish etilishini ikki usuli mavjud:

- uni SWF formatida saqlash so'ng sichqoncha yordamida fayl znachogini bu fayl saqlanadigan katologidan brouzer oynasiga o'tkazish.

- flash filmni HTML formatiga eksport qilish so'ng oddiy usul bilan brouzerda ochish.

Flash filmlarni Web sahifa elementlari sifatida ishlatish turli tumandir. Bulardan ayrimlari qyyida keltirilgan:

- sahifani estetik ko'rinishi oshirish uchun mo'jallangan "oddiy" animatsiya;

-foydalanuvchi harakati (sichqonchani siljitish tugmani bosishga bog'liq bo'lgan animatsiyalashgan knopka, bunday knopka gipersilka yordamida ishlatilishi mumkin yoki biror bir vazifani bajarishi mumkin).

-Web-sahifa foydalanuvchilardan biror-bir ma'lumotni qabul qilishga mo'ljallangan forma ko'rinishidir.

-reklama banneri .

Flash filmlarda tovushni ishlatishni bir necha usullarini taqdim etadi. Siz filmni syujeti va foydalanuvchi bajargan amaldan qat'iy nazar to'xtamaydigan amalni yaratishingiz mumkin. Alternativ variant animatsiyava tovushni sinxromlash. Bundan tashqari Flash tugmachalariga tovush ishlatishingiz mumkin. Bu esa tugmalarni interaktiv qiladi.

Flash filmni yaratish asosiy sxemasi.

Birinchi o'rinda Flash Web-sahifalarni yaratish uchu mo'ljallanganligiga qaramay u bilan ishlash vaqtida teg, atribut, sahifa va shunga o'xshash terminlar bilan siz kamdan kam uchratasiz. Shuning uchun Flashni o'rganishda avval uning terminalogiyasini tushinib olish kerak. Flashda tovushni qo'shishni ikki xili mavjud. Xodisaviy boshqariladigan (event sound) va oqimli (stream sound). Flash tovush bilan ishlashda yaratuvchilar uchun quyidagi imkoniyatlarni yaratadi:

-bo'linuvchan tovush kutubxonalari yaratish yani bir kutubxonadagi tovushni bir necha filmlarda foydalanish imkoniyatidir.

-tovush sifati va eksporti qilinayotgan filmni optimal birlashuvini tanlash maqsadida tovushli ma'lumotlarni qo'shish ko'rsatgichlarini boshqarish muallif alohida tovush yoki filmdagi barcha tovushlar uchun qo'shish ko'rsatgichini tanlashi mumkin. Bundan tashqari siz tovushni o'zgartirish uchun turli qo'shimcha effektlardan foydalanishingiz mumkin.

Film kadrlarini almashish dinamikasi vaqt diagramasi (Timelene) yordamida ta'riflanadi. Vaqt diagramasini parametrlari sifatida kadrlar almashinish chastotasi obyekt xarakatini boshlash va yakunlash va boshqa parametrlarni ko'rsatish mumkin. Filmda bir necha xil turli obyektlardan foydalanish mumkin. Har birining holati boshqasidan mustaqil ravishda o'zgarishi yoki umuman o'zgarmay qolishi mumkin. Filmni turli elementlarini xarakatini ta'riflashni osonlashtirish uchun ularni yupqa halqa bilan solishtirishadi. Bunday varaqlarni bir nechasini bir to'plamga yig'lsa ma'lum saxna namoyon bo'ladi Saxna

(Scene)-bu Flashni yana bir terminlaridan biridir.Har bir saxna qavatlarini ma'lum bir to'plamidan iboratdir.Sodda filmlar uchun bir qavatdan iborat saxnani yaratish va ta'riflash kifoyadir.Murakkab filmlar uchun bir necha turli ko'rinishdagi saxnalar yaratish kerak bo'ladi.Bir saxnadan ikkinchisiga o'tish vaqti diagrammasi balki boshqa bir mexonizmlardan foydqlanib amalgam oshiriladi.Oddiy holda ,film saxnalari ma'lum ketma-ketlikda, ularni tartib raqami bo'yicha bajariladi.Filmni murakkabroq usulda tuzishda Action Script tilidan foydalaniladi.Murakkar filmni yaratishda muhim rolni yana bir tushincha klip (clip yoki Movie clip).Klip bu-simvolni maxsus turidir.U o'ziga xos mini filmdir.Bu film uchun xususiy vaqt diagrammasi va xususiy parametrlar(masalan,kadrlar chastotasi)o'natilgan.Klipdan ko'p marotoba foydalanish uchun filmni bioshqa bir elementlari singari kutubxonasiga joylashtirish mumkin.Filmni har bir elementi klipni ichida ham ishlatilishi mumkin.Agar film ichidagi aktivlashtirishni ba'zi qo'shimcha shartlarini berish uchun Action Script tili vositalaridan foydalanish mumkin.Klipda interaktiv elementlarni(masalan,knopka) ham qo'shish mumkin.

Flash filmlarni yaratish ketma-ketligi.

Windows dasturlarini ko'pchiligini ishga tushirganda ushbu dasturbilan ishlash uchun "zagatovka"(bo'sh hujjat)avtomatik ravishda yaratiladi.Masalan Word muxaririni oynasi ochilganda unda bo'sh hujjat ochiladi.Bu hujjatga siz matn kiritishingiz ,grafika qo'shish shrift ko'rinishini o'zgartirish va boshqalarni amalgam oshirishingiz mumkin.Yangi hujjat uchun o'rnatilgan parametrlar qo'llaniladi.Hujjatingiz sizga ma'qul bo'lgan ko'rinishda bo'lishi uchun parametrlarni kerakli ko'rsatgichlarini o'rnarib qo'yish talab qilinadi.Flash juda qulay interfesga ega bo'lsa ham ,Flash filmlarni yaratish juda murakkab ishdir.Shuning uchun film yaratishning har bir qadamidan so'ng natijani tekshirish lozim.Flash filmnialohida saxnasini ham butun filmni tekshirish imkoniyatini beradi.Agar tekshirish muvafaqiyatli o'tgan bo'lsa natijani diskda saqlashni unutmang.Chunki Flashda avtomatik saqlash funksiyasi mavjud emas.

Ish oblasti insterimentlari paneli.

Ish oblasti xoxlagan sichqonchani o'ng tugmasi bosilganda kontekst menyusi ochiladi Bu menyu ish oblasti vafilmni parametrlarini o'zgartirish uchun asosiy buyruqlardan iborat.Bunday buyruqlar 20 tadan ortiqroqdir.Hozircha ulardan bittasi SCENE (saxna)ga

to'xtab o'tamiz. Bu buyruqni tanlaganda nomi Scene nomli panel paydo bo'ladi. Bu panel film saxnalari bilan ishlash uchun mo'ljallangan. Uning yordamida siz quyidagilarni amalga oshirishingiz mumkin.

- filmning saxnalari ro'yxatini ko'rish, saxnalarni ro'yxatdagi joylashuvini filmda paydo bo'lish ketma ketligi mosdir.

- filmni xoxlagan saxnasiga (bu saxnani ro'yxatdan tanlab) o'tish;

- tanlangan saxnani nusxasini olish, buning panelni pastki qismidagi knopka bosiladi, nusxa ro'yxatga saxna originalidan keyin joylashtiriladi;

- yangi (bo'sh) saxna qo'shish, saxna ro'yxatga tanlangan saxnadan keyin joylashadi;

- tanlangan saxna o'chiriladi.

Montaj stoli va uning xususiyatlari.

Ish oblastini markaziy qismida montaj stoli joylashgan. Uni keyinchalik qisqartirish uchun faqat stol deb ataymiz. Stol obyektlari xususiyatlarini paneli va boshqalar. Masalan aksariyat Windows-dasturlarida (ofis paketlar va boshqalar) instrument paneli tarkibini boshqarish buyruqlari View(vid) menyusiga kiradi. Flashda esa bu buyruqlar Window menyusida joylashtirilgan. Taxrirlanayotgan obyektlar xususiyatlar paneli ham Boshqacharoq ko'rinishga ega. Lekin bu turdagi yangiliklar unchalik ham qiyinchilik tug'dirmaydi. Va bir nechta ish seanslardan so'ng ular tabiiydek tuyiladi. Shunday qilib, Flash MX muxaririning ish joyi bilan tanishib chiqamiz.

Flash dasturida animatsiyalar yaratish dasturi Shape Tweening usuli.

Usul asosan obyekt formasi bir turdan ikkinchi turga o'zgarishga asoslangan.

1. Birinchi marta Oval instrumenti yordamida aylana chizib olamiz. Ellips emas balki aylana hosil bo'lishi uchun Shift tugmasi birbalikda bosib tortiladi.

2. Arrow elementini ishlatib va timeline bo'limidan 30-kadrga tanlang. Bosh menyusidan Insert Keyframe(F6) buyrug'ini bajaramiz. Bu harakat bilan siz 30-kadrga kalit kadrni aylantirdik. E'tibor bering leneyka ustida kulrang rangda bo'yalib qoladi.

3. 30-kadrga Rectangli(R) instrumenti yordamida aylananing chap qismida to'rtburchak chizing. Arrow instrumentini tanlaymiz va aylanani belgilangan holda DELETE tugmasi yordamida o'chirib tashlaymiz.

4.Sichqoncha tugmasini 1-30 kadrlar orasida ikki marta ishlatamiz va hosil bo'lgan paneldan Tweening bo'limini tanlaymiz.Shape holatini o'rnatamiz.

5.Kadrda kursorni o'rnatgan holda ENTER tugmasini bosamiz va hosil bo'lgan animatsiyani ko'ramiz.Oxirgi natijani ko'rish uchun esaControl>Test Movie(CTRL+ENTER)buyrug'ini bajaramiz.

Motion Tweening usulini yaratish uchun quyidagi usul obyekt xususiyatlarini boshqarish uchun xizmat qiladi.Eng oddiy foydalanish sifatida koordinata o'zgarishini keltirishimiz mumkin.

Animatsion simvol yaratish.

Macromedia Flashda ishni osonlashtirish va rolik o'lchamlarini yaxshilash uchun simvol tushinchasi kiritilgan. Simvol quyidagicha bo'lishi mumkin. Grafika (Graphic), tugma(button), animatsion rolik (movie clip), tovush(sound). Bu imkoniyat yatarilgan obyektlar bir necha marta foydalanish imkonini beradi. Flash filmlarini AVI formatiga o'tkazikishi ularning interaktivligini yo'qolishiga olib keladi.Shuning uchun ushbu variant keng tarqalmagandir.Flash filmlarini namoyish etishni yana bir turi mavjuddir.Bu variant Flash filmni animatsiyalashgan GFI formati ga o'tkazishdir.Bu holda eksportni ikki turi mavjud :har bir foydali animatsiyani bir kadrda mos keluvchi GFI fayllar to'plami ko'rinishida yoki bir animatsiyalangan fayl ko'rinishidadir.

1)movie klip tayyor animatsiyadan fodalnamiz.Buning uchun Shift tugmasini bosib turgan holda birinchi va oxirgi kadrni tanlaymiz.So'ngra Edit>Copy Frames yordamida kadrlar nusxasini olamiz.

2)File New buyrug'i yordamida yangi simvol yaratamiz.Hosil bo'lgan muloqot oynasida Movie Clip holatini tanlaymiz va Name maydoniga unikal nom kiritamiz: ball clip

3)Yaratilgan Movie clip holatini tanlaymiz,birinchi kadrda kursorni o'rnatamiz va oldin nusxalangan kadrlarni Edit>Paste>Frames(CTRL+ALT+V)buyrug'ida qo'yamiz.Yuqoridagi Scene 1 belgisini tanlang.

4)Window>Library (CTRL+L)buyrug'I yordamida Library panelini ishlatamiz.E'tibor bergan bo'lsangiz,panelda ikkita obyekt bo'lib ,birinchi obyekt ball clip –Movie clip va ikkinchi obyekt –ball –Graphik.

5)Shuni bilish lozimki ball –clip animatsiyadan biz uni obyekt sifatida bir necha foydalanishimiz mumkin.Paneldan obyektни tanlab,sichqoncha yordamida ishchi oynaga olib tashlaymiz va CTRL+ENTER ni bosamiz.

1.3 Action Script da obyektlarni yaratish va boshqarish.

Dasturlash tillarida obyekt faqat shu obyektga qo'llanishi mumkin bo'lgan atributlar (xususiyatlar)to'plami va usullar ro'yxati bilan ifodalanadi.Har bir sinf obyektlari uchun o'z atributlar va usullar to'plami belgilangan.Masalan “tugma” obyektі uchun atributlar sifatida ishlatilishi mumkin.Ish oblastidan pastki qismida xususiyatlar inspektori paneli joylashgan,agar ish oblastida biror bir obtekt tanlanmagan bo'lsa yoki obyektlarumuman bo'lmasa u holda xususiyatlar inspektori formati avtomatik ravishda o'zgaradi.Butun hujjat uchun xususiyatlar inspektori formati ko'rsatilgan. Aniq obyekt-bu mos sinfnıng exzemplyaridir.Bir sinfnıng ikki exzemplyari bir biridan atributlar qiymatlari bilan farq qilishi mumkin.Obyekt ustida harakatlarni ifodalash shuningdek obyektning bo'ysinganligini ko'rsatish uchun odatda “nuqtali usul”qo'llaniladi.

Obyekt ustidagi harakatlarni ifodalash shuningdek obyektning bo'ysinganligini ko'satish uchun odatda “nuqtali usul”qo'llaniladi.Masalan button 1 tugmasiga On Prees usulini qo'llash kerakligini ko'rsatish uchun Button 1 on Prees konstruksiyasi ishlatiladi.Agar Button 1 tugmaning Clip 2 ga tegishligini ko'rsatish kerak bo'sa unda yozuv quyidagicha ko'rinishga ega bo'ladi; Clip 2 Button 1.

Action Script da qo'llaniladigan obyekt modeli boshqa senariylar tillari obyekt modellariga ko'p jihatdan o'xshaydi (masalan:Java Script).Shu bilan birga ba'zi faxqlar ham mavjud .Ulardan asosiysi Action Script da obyektlar iyerarxiyasi Web brouzer oynasida tasvurlangan HTML hujjatiga nisbatan emas,balki Flash pleerga nisbatan quriladi.

Yuqorida aytib o'tilgandek,istalgandek snariyda mavjud Flash pleer asos obyekt hisoblanadi.Keyingi darajada obyektlarning 4 sinfi joylashgan.

- Movie (Film obyektlari);
- Core (Yadro obyektlari);
- Cleent /Server (Bajarishning klient-server muhiti obyektlari);
- Authoring (Ishlab chiqish muhiti obyektlari);

Movie sinfi obyektlari senariyda film elementlari orasidagi elementlari orasidagi munosabatlarni (ya'ni film tuzilishini) tasavur qilish va shuningdek, film elementlari parametrlari vazirakatini boshqarish imkonini beradi. Ushbu sinfga xususan quyidagi obyektlar kiradi:

- button(tugma)-senariyda ifodalovchi obyekt, bunday obyekt uchun, masalan kiritish fokusini o'rnatish tartibi dinamik tarzda o'zgarishi mumkin;

- movie clip (klip)- senariyda klipni ifodalovchi obyekt. Bunday obyektни ifodalash uchun, masalan ichki kliplar soni dinamik tarzda o'zgarishi mumkin;

- mouse(sichqoncha)-senariyda sichqoncha ko'rsatgichini ifodalovchi obyekt, u ko'rsatgich ko'rinishi va uslubini boshqarish imkonini beradi.

- key(klaviatura)-senariyda filmni boshqarish uchun ishlatiladigan klavishlarni ifodalovchi obyekt.

- color(rang)-senariyda klip palitrasini va mos holda ushbu palitra o'zgarishini imkonini beruvchi obyekt.

- sound(tovush)-senariyda tugma yoki klip bog'langan tovushli simvolni ifodalovchi obyekt.

- stage(stol)-ishchi stoli ba'zi parametrlarini boshqarish uchun mo'ljallangan obyekt.

- text Field(matn maydoni)-senariydagi dinamik Action Skript da Button va Movie clip sinflari obyektlari orasida bo'lanish munosabatlari juda qiyin, ular bir filmdan ikkinchisiga oladi va hatto saxnadan saxnaga o'tganda ham o'zgarishi mumkin. Masalan, bir saxnada tugma klip tarkibida kirsa, ikkinchisiga klip tugmani "jonlashtirish" uchun ishlashyirilishi mumkin. Movie sinfini obyektlarini (Text Fielddan boshqa) Button va Movie klip obyektlariga nisbatan bo'ysingan deyish mumkin. Chunki ular Button va Movie klip obyektlari tarkibiga kirishi mumkin. Lekin aksincha emas.

Etibor bering, Action Skript kadrlar uchun ham film saxnasi uchun ham mos obyektlar sinfi yaratilmagan. Gap shundaki Action Skript da filmning ushbu elementlarini xususiyatlarini o'gartirishga yo'l qo'ymaydi. Siz senariy yordamida faqatgina kadrda uning nomi yoki raqami orqali murojaat qilishingiz shuningdek, uning yuklangan yoki yuklanmaganligini tekshirishingiz mumkin. Qoida ko'ra kadr yoki saxna harakatlar va usullar parametrlari sifatida ishlatiladi.

Cori sinfi obyektlari film obyektlari bilan ishlash uchun mo'ljallangan va yordamchi rolini bajaradi. Agar film obyektlari boshqarishda (xususiyatlarni o'zgartirish) standart usullar etmasa, ular qo'llash mumkin. Cori obyektiga xususan quyidagi obyektlar kiradi:

-matn (matematika)-senariyda sonli kattaliklar bilan ishlash qo'llaniladigan obyekt. Action Skript tili boshqa sinflardan farqli ravishda Matn obyektida qo'llanishi cos usuli yordamida cosinus bevositiya matn usuli yordamida qiymatini olish mumkin.

-number (raqam)-raqamli qiymatli ba'zi maxsus operatsiyalarni bajarish uchun mo'ljallangan obyekt, masalan uning yordamida ba'zi diapozonlarda ko'proq sonlarni aniqlash mumkin.

-string(satr)senariyda satrlar bilan ishlash uchun qo'llaniladigan obyekt.

-date (sana)senariyda calendar sanalari va vaqt bilan ishlash uchun qo'llaniladigan obyekt. Client/server sinfi obyektlari XML tilida yozilgan hujjatlar bilan ishlash uchun mo'ljallangan. Ushbu obyektlarning bazilari XML hujjatlarini yuklanishi qayta ishlanishi va yuborilishini taminlasa boshqalari- socketlar orqali bog'lanishga xizmat qiladi. Action Script shu orqali Flash filmlari saqlaydigan XML hujjatlarini serverli qayta ishlashni qo'llashni amalga oshiriladi.

Klip holatlarini qayta ishlash haqida umumiy ma'lumotlar

Klip holatini qayta ishlash uchun quyidagi operator ishlatiladi:

```
onClipEvent () { }
```

Qavs ichiga esa har bir buyruqqa mos skript yoziladi. Ularni qisqacha izohlab o'tamiz. Holatlarni ikkiga bo'lishimiz mumkin. Bular Klip namoyish qilinganda bajariladigan holatlar:

load - klip ekranga birinchi marta ko'ringanda, yoki klipning yangi nusxasini saqlovchi assosiy kadr ga o'tganda bajariladigan holat.

unload - loadga teskari bo'lib, klip ekranda ko'rinmaganda bajariladigan holat.

enterFrame - bu holat klipning har bir kadriga o'tganda bajariladigan holat. Agar klip faqat bitta kadr dan iborat bo'lsa ham u rolikning namoyish qilish vaqti bilan bir xil ravishda ya'ni 1 sekundda 12 marta tezlik bilan takrorlanadi.

data - holati klipga tashqi ma'lumot yuklanganda bajariladi. U matnli faylning ma'lumotlarini biror o'zgaruvchiga olganda, yoki tashqi swf fayllar yuklanganda bajariladigan holatni ifodalaydi.

Foydalanuvchi harakatiga ko'ra bajariladigan holatlar:

mouseDown - sichqonchaning chap tugmasi bosilganda bajariladigan holat. Bu holat sichqoncha ishchi stolning qaysi obyekt ustida bo'lmasin bajariladi.

mouseUp - sichqoncha ko'rsatgichi qaysi obyekt ustida bo'lmasin, uni tugmasini bosib, so'ng qo'yib yuborilganda bajariladigan holat.

mouseMove - sichqoncha tugmasi qaysi obyekt ustida bo'lmasin, ozgina siljiganda bajariladigan holat.

keyDown - klaviaturadagi ixtiyoriy tugma bosilganda bajariladigan holat.

keyUp - klaviaturadagi bosilgan tugmani qo'yib yuborilganda bajariladigan holat.

Bu holatlarning har biriga keyinchalik batafsil to'xtalib, misollar asosida tushunirib o'tamiz. Hozir esa **load** va **unload** holatlari bilan tanishib o'tamiz.

Yonib o'huvchi chiroqni **load** va **unload** holatlaridan foydalanib tasvirlang.

1-asosiy kadrqa uzib ulagichning yoqilgan holdagi rasmini keltirib qo'yamiz, 9-kadrgacha oddiy kadrlarni joylashtiramiz, 10-kadrga asosiy kadr deb olib, unga uzib ulagichni o'chirilgan holdagi tasvirini hosil qilamiz. 19-kadrgacha oddiy kadrlarni qo'yamiz. Xullas, 1-kadrdan 9-kadrgacha ulangan uzib-ulagich, 10-kadrdan 19-kadrgacha o'chirilgan uzib ulagich tasvirlangan. Yana bir qatlam hosil qilib unga **movie clip** va **movie clip**dan tarqalayotgan yorug'lik (nur deb nomlangan) **movie clip**larni joylashtiramiz. Ular quyidagi rasmda tasvirlangan:

Bizda yorug'lik va **movie clip** alohida-alohida klip bo'lgani uchun ularni boshqarish oddiy. 10-kadrga o'tib uzib-ulagichni belgilaymiz va unga quyidagi skriptni yozamiz:

```
onClipEvent (load) {  
    _root.nur._visible=0;  
}  
onClipEvent (unload) {  
    _root.nur._visible=1;  
}
```

Demak 10-kadrga kelganda `_root.nur._visible=0`; buyrug'I bajariladi, ya'ni nur ko'rinmaydi. 19-kadrga kelganda `_root.nur._visible=1`; buyrug'I yordamida yana qaytib ko'rinadi. Dasturni ishga tushirsak, uzib ulagich ulanadi, **movie clip** yoqiladi, uzib-ulagich o'chiriladi va **movie clip** o'chadi, shu rolik to'xtovsiz namoyish qilinib boriladi.

EnterFrame holati

O'ngdan chapga tomon harkatlanuvchi sharni dasturlash orqali hosil qiling. Dasturni tuzish uchun ekranning chap tomoniga birorta shar rasmi tasvirlangan moie klipni joylashtiramiz. So'ng unga quyidagi skriptni yozamiz::

```
onClipEvent (enterFrame) { _x+=10;
}
```

Shar to'xtovsiz chapdan o'ngga harakatlanadi.

To'xtovsiz ravishda avval kattalashib so'ng kichiklashuvchi shar tasvirini **interFrame** holati yordamida hosil qiling.

Oynaga 1 ta shar rasmi chizilgn **movie clip**ni joylashtiramiz. O'ylangan g'oyani amalga oshirish uchun bizga 1 ta qo'shimcha o'zgaruvchi kerak bo'ladi. Uning qiymati faqat bir marta klip namoyishi boshlanganda aniqlanishi kerak. Buning uchun uni load holatida joylashtiramiz va uning qiymati sifatida **true** (chin)ni yozamiz:

```
onClipEvent (load) { a=true;}
```

Keyingi qadam shart operatori yordamida tekshiramiz, agar sharning hajmi 400 pikseldan katta bo'lmasa va a=true qiymat qabul qilgan bo'lsa, u holda shar vertikalva gorizontal 10 rikselga kattalashadi. Agar qo'yilgan shart bajarilmasa, a=false qiymatni qabul qiladi. Bunda shar 400 pikselgacha kattalashadi. Bu quyidagicha ifodalaniladi:

```
onClipEvent (enterFrame) {
    if (_xscale<400 && a) {
        _xscale+=10; _yscale+=10;
    }else {
        a=false;
    } }
}
```

Sharni qayta kichiklashtirish uchun quyidagi skriptni kiritamiz:

```
onClipEvent (enterFrame) {
    if (_xscale>10 && a==false) { _xscale-
        =10; _yscale-=10; }else {
        a=true;
    } }
}
```

Shu skriptlarni hammasini **movie clip**ga kiritsak, Shar pulsiv tarzda kattalashib, kichiklashadi.

getTimer () funksiyasi

getTimer() o'tgan millisekundni qaytaradi. Ushbu funksiya yordamida vaqtni ifodalovchi **taymer**lar yaratishimiz, rolikning biror kadrini bir necha soniyaga to'xtatib qo'yishimiz mumkin. Uning qo'llanilishini misol asosida ko'rib o'tamiz. O'tilgan vaqtni ifodalovchi taymer hosil qlamiz.

Vaqt degan **movie clip** hosil qilamiz. Unda 1 satrli "soniya" nomli dinamik maydonni keltirib qo'yamiz. **Movie clip**ni oynaga keltirib qo'yamiz va unga quyidagi skriptni kiritamiz:

```
onClipEvent (enterFrame) {  
    a=Math.floor(getTimer()/1000);  
    soniya='o'tgan vaqt:'+a; }  
getTimer()
```

funksiyasi millisekundni qaytaradi bizga esa sekundlar ko'rinishi kerak, shuning uchun uning qiymatini 1000 ga bo'lib olamiz va Math.floor() funksiyasi yordamida uning qiymatini eng kichik butun songacha yaxlitlab olamiz. "Soniya" nomli obyektga uning qiymatini izoh bilan chiqaramiz.

Agar vaqt teskari tartibda kamayib borishini xohlasak, unda **movie clip**ga quyidagi skriptni kiritamiz:

```
onClipEvent (enterFrame) {  
    a=Math.floor(getTimer()/1000); if (a<10){  
soniya=(10-a)+' soniya qoldi';} else {soniya='vaqt tugadi';}}  
Bu dasturda o'tilgan vaqt 10 soni bilan tekshiriladi. Agar o'tgan vaqt 10 dan kam bo'lsa unda o'tgan soniyalar chiqariladi. Agar 10 ga teng yoki 10 dan katta bo'lsa, vaqt tugadi degan yozuv ko'rinadi.
```

Misol sifatida, elektron kitob yaratamiz. Har 5 sekunda uning yangi sahifasi ko'rinsin.

Oynaga 1 ta dinamik maydondan iborat bo'lgan **movie clip**ni oynaning 1-qatlamiga olib qo'yamiz. Buning uchun avval yangi **movie clip**ni hosil qilamiz va unga 1 ta dinamik maydonni keltirib qo'yamiz. Dinamik maydonni "soniya" deb nomlaymiz. Oynhaga o'tib F11 tugmasini bosamiz va kutubxonadan dinamik maydondan iborat **movie clip**ni oynaga olib qo'yamiz. So'ng ikkinchi qatlamni hosil qilamiz. Unda 1-kadrga **static text** o'rnatib kitobning birinchi betidasgi ma'lumotlarni kiritamiz. 2-kadriga ham static text qo'yib, unga kitobning 2-betidagi ma'lumotlarni kiritamiz^ 1-qatlamning 1-kadriga joylashgan **movie clip**ga quyidagi skriptni kiritamiz:

```
onClipEvent (load) {  
    q=0;}  
onClipEvent (enterFrame) {  
    a = Math.floor((getTimer()-q)/1000); if (a<5) {  
        soniya = (5-a)+' soniya qoldi'; } else {  
        _root.nextFrame();  
        q=getTimer(); } }
```

Avval q o'zgaruvchisining qiymatini o'tilgan vaqtga teng deb olamiz va a o'zgaruvchisiga o'tgan vaqtdan q ni ayirib, so'ng uni soniyaga aylantiramiz. Agar aning qiymati beshdan kichik bo'lsa, o'tgan vaqt kamayib borish tartibida izoh bilan e'lon qilinadi.

Agar 5 ga teng bo'lsa, keyingi kadr namayish qilinadi. **nextFrame()** - keyingi kadrda o'tish buyrug'idir. Biz unga o'zgaruvchi sifatida murojaat qilamiz, shuning uchun uning oldiga root so'zini qo'shib qo'yamiz. Q ga o'tgan vaqtni olamiz. Bu keyingi galda a ning qiymatini yana 5 dan boshlash uchun kerak. Xullas dastur ishga tushgach bizga birinchi sahifa ko'rinadi, 5 soniyadan so'ng ikkinchi , yana 5 soniyadan so'ng uchinchi,... oxirgi sahifaga kelganda **timer** 5 dan 1 gacha teskari tartibda sanaydi va yana hechqanday o'zgarishsiz yana 5 dan 1 gacha teskari tartibda sanaydi. Bizga **timerning** to'xtovsiz sanashi kerak emas. Shuning uchun 1-qatlamning 5-kadriga asosiy kadrni hosil qilib, **movie clip**ni tanlaymiz va unga quyidagi skriptni kiritamiz: onClipEvent (load) {

```

    q=getTimer(); } onClipEvent
    (enterFrame) {
        a = Math.floor((getTimer()-q)/1000);
        if (a<5) {
            soniya = (5-a)+' soniya qoldi';
        } else {
            soniya='Bu kitobning oxirgi beti';
        }
    }

```

Bunda 5-sahifa yangilanganda 5 soniyadan so'ng bizga "bu kitobning oxirgi beti" degan yozuv ko'rinadi.

Ektranda namoyish qilinadigan rolikni bir necha soniyaga to'xtatib turing va rolikni namoyishini davom ettirish.

Buning uchun tayyor rolikning to'xtashi lozim bo'lgan kadriga quyidagi skriptni kiriting.

```

    q=getTimer(); while (getTimer()-
    q<200){ stop();} play();

```

q ga joriy vaqtni olamiz. Uning qiymati bizni qiziqitirmaydi. So'ng dastlabki vaqtning qiymati joriy vaqtning qiymatidan 200 ms farq qilgunga qadar stop(); buyrug'i amal qiladi, ya'ni rolik namoyishi to'xtaydi. 200 ms so'ng play(); buyrug'I amal qiladi, ya'ni rolik namoyishi davom ettiriladi.

mouseMovie vasifasi

mouseMovie holati sichqoncha ishchi maydon ustiga siljigandagi holat bo'lib u sichqoncha siljiganda bir necha marta protsessorning holatni qayta ishlash tezligiga bog'liq tezlikda ishlaydi.

Shaxsiy kursorini yaratish masalasini qaraymiz.

Shaxsiy kursorini yaratish uchun avvalambor shu kursorning rasmini chizish, yoki uni **import** qilish lozim. Biz kursor sifatida archani tanlaymiz. Biror tugma ustiga kelganda

kursorning ko'rinishi o'zgarganligi tufayli 2-rasm sifatida strelkani tanlaymiz. 1 ta **movie clip** hosil qilib, uni 1-kadriga archaning 2-kadriga chiroqning rasmini keltirib qo'yamiz. Har ikkala kadrda ham stop()buyrug'ini yozib, **Movie clip**ni kursor deb nomlaymiz va uni oynaga keltirib qo'yamiz. Endi kursorni ko'rinmas holatga keltiramiz. Uni faqat bir marta ko'rinmas holga keltirish kifoya, shuning uchun uni **load** holatiga joylashtiramiz:

```
onClipEvent (load) {  
    Mouse.hide(); }  
}
```

Hosil qilgan kursorimiz, ko'rinmas kursor ortidan harakatlanishi uchun, yangi kursorga ko'rinmas kursorning koordinatalarini o'zlashtirishimiz lozim. Sichqoncha faqat siljiganda uning koordinatalari o'zgaradi. Shuning uchun uni **mouseMove** holatiga joylashtiramiz. Ushbu skriptni enterFrame holatiga joylashtirsak ham bo'ladi, lekin bu holat sichqoncha siljimaganida ham har doim bajarilib turganligi tufayli u protsessorga ortiqcha vazifa yuklaydi. Shuning uchun skriptni mouseMove holatiga yozamiz.

```
onClipEvent (mouseMove) {  
    _x=_root._xmouse;  
    _y=_root._ymouse;  
    updateAfterEvent(); }  
}
```

Bu yerda xmouse Ko'rinmas kursorning koordinatalari va _x joriy kursorning koordinatasi. updateAfterEvent() - rolikning namoyishini tezlashtirish uchun ishlatiladi. Agar ushbu skriptni yozmasak sichqoncha harakatlanganda hosil qilgan yangi kursorimiz to'xtab-to'xtab harakatlanadi, chunki uning harakati filning tazligiga bog'liq bo'ladi. Bu buyruq faqat **mouseMove** holatining ichidagina ishlaydi. Dasturni testdan o'tkazib, uni tekshirib ko'ramiz.

Navbatdagi masala sichqoncha tugma ustiga kelganda uning ko'rinishi o'zgarishi kerak. Buning uchun ekranga tugmani joylashtiramiz. Bunda tugma 2-qatlamda, kursor joylashgan qatlam ostida joylashgan bo'lishi lozim, aks holda kursor tugma tagiga qoladi. Tugmaga quyidagi skriptni ko'ramiz:

```
on (rollOver) {  
    kursor.gotoAndStop(2);} on  
(rollout) {  
    kursor.gotoAndStop(1); }  
}
```

Bunda tugma ustiga sichqoncha ko'rsatkichi kelganda kursorning 2-ko'rinishi, ya'ni sichqonchanning kursori, tugma ustidan ketganda 1-ko'rinishi ya'ni archa ko'rinadi.

Bulardan tashqari sichqoncha tugmasi bosilganda uning bosilganligini bildiruvchi biror o'zgarish bo'lishi kerak bo'lsa, masalan, ko'rsatkich uchida yorqin rang bo'lishini xohlasak, quyidagilarni amalgam oshirish lozim. Avval yorqin rangni hosil qilamiz

Men quyidagini tanladim: **X**. Uni oynadan tashqariga ko'rinmas holda joylashtiramiz va unga quyidagi skriptni kiritamiz:

```
onClipEvent (mouseDown) {  
    _x = _root._xmouse;  
    _y = _root._ymouse;  
    this._visible = 1;  
    updateAfterEvent();}  
onClipEvent (mouseUp) {  
    this._visible = 0;  
    updateAfterEvent();}
```

Ushbu skript quyidagicha o'qiladi: Sichqoncha tugmasi bosilganda, uning koordinatalarini ushbu **movie clip**ga o'zlashtir va **movie clip**ni ko'rinadigan holatga keltir, hodisani tezlashtir. Sichqoncha tugmasi qo'yib yuborilganda, **movie clip**ni ko'rinmas holga keltir va hodisani tezlashtir. Dasturni ishga tushirsak bizga tayyor foydalanuvchi kursori hosil bo'ladi. Quyida hosil qilingan kursorning tugma ustuda bosilgandagi va tugma ustida bo'lmagan holatlarining tasviri keltirilgan.



10-rasm. Ko'rinishi o'zgartirilgan sichqoncha ko'rinishi va tugma.

Linza yordamida kichkina xaritani kattalashtirish masalasini ko'ramiz.

Xarita rasmi tasvirlangan **movie clip**ni oyna markaziga joylashtiramiz. Linza nomli **movie clip** hosil qilamiz va unga xarita rasmini joylashtiramiz, xaritaga "mc" deb nom qo'yamiz. **Alligin** oynasidan foydalanib, uni markazga joylashtiramiz. Ikkinchi qatlamni hosil qilib, unga linzaning rasmini keltirib qo'yamiz va markazga joylashtiramiz. 1-va 2-qatlamlar orasida yangi 3-qatlamni hosil qilamiz va unga "xxx" deb nom beramiz. Bu qatlamda pastki qatlamda jolashgan rasmning ko'rinmasligi kerak bo'lgan qismi yashiriladi.

Linza aylanasi mos keluvchi aylanani rasmini chizamiz, uni linza ichiga choylashtiramiz. U bizga tuynuk vazifasini bajaradi. So'ng, "xxx" qatlam ustiga sichqonchaning o'ng tugmasini bosamiz va maska bo'limini tanlaymiz.

Maskada joylashgan grafika rolikda namoyon qilinmaydi va undan pastda joylashgan, niqoblangan qatlamlardagi rasmlarni namoyon qiluvchi oyna (darcha) vazifasini bajaradi. **Statik text**, grafik simvollar va rasmlarni maska sifatida ishlatish mumkin, agar ular 1 qatlamda joylashgan bo'lsa. **Dinamik text** va chiziqlar maska bo'lolmaydi.

xxx va 1-qavat orasiga yana bir qavatni hosil qilamiz va uni eng pastga ko'chiramiz. Hosil qilingan qatlamga xxx qatlamdagi aylanani nusxasini keltirib qo'yamiz. U ham maska qilingan qatlam ichiga kirib, fon vazifasini bajaradi. Linza orqali qaralganda rasmning asl nusxasi ko'rinmasligi uchun ishlatiladi, aylananing rangini oyna rangi bilan bir xil qilish kerak.

Oynaga hosil qilingan linza **movie clip**ini keltitib qo'yamiz. Linzaga quyidagi skriptni kiritamiz:

```
onClipEvent (load) { n = 2;  
}
```

n o'zgaruvchisi linzaning necha marta kattalashishini belgilaydi.

```
onClipEvent (enterFrame) { _xscale  
= 100*n; _yscale = 100*n; _x =  
_root._xmouse; _y =  
_root._ymouse; mc._x = -  
_x+275; mc._y = -_y+200;}
```

Xaritaning o'lchamini x va y o'qlari bo'yicha 200% (n=2 bo'lganda) ga oshiradi, linzaning koordinatalarini sichqonchaning koordinatalari bilan bir xil qilib boriladi. Bunda linza sichqoncha ko'rsatkichi bilan harakatlana boshlaydi. mc._x = -_x+275 xaritaning gorizonta koordinatasiga linzaning gorizonta koordinatasi qarama-qarshi ishora bilan o'zlastilyapti, bunda xarita linza harakatlangan tomonga qarama-qarshi harakatlanadi. Xaritaning mc nomli nusxasi joriy kadr markazida joylashgan, oynadagi xaritaning rasmi esa, oyna markaziga joylashgan. Ular orasidagi farq 275, 200 bo'lganligi uchun x va y koordinatalarga ular mos ravishda qo'shiladi. Shu skript yozilgandan so'ng linza markaziga xaritamizning joriy qismi tasvirlanadi.

```
onClipEvent (mouseDown) { n = 4;}  
onClipEvent (mouseUp) { n = 2;}
```

Sichqoncha bosilganda xaritaning o'lchami 4 marta oshadi, qo'yib yuborilganda xarita yana 2 marta kattalashadi. Dasturimiz quyidagi ko'rinishga ega:



11-rasm. Xaritani kattalashtirish.

Key obyektining asosiy metodlari

Bu bo'limda biz Flash obyektlarini klaviatura yordamida boshqarishni qarab chiqamiz. Hamda, **key** obyektining asosiy metodlari bilan tanishib chiqamiz. Bu obyekt bizga hozirgi vaqtda qanday tugma bosilgan va oxirgi bosilgan tugmaning qaysiligini aniqlash uchun ishlatiladi. Klaviaturadagi ixtiyoriy tugma bosilganda, 3 xil holat sodir bo'ladi. Bular **keyPress**, **keyDown** va **keyUp**. Ularning har birini imkoniyatlarini misollarda qarab o'tamiz.

Yo'nalish tugmalari yordamda **movie clip**ni harakatlantirish.

Buning uchun ekranga 1 ta **movie clip**ni keltirib qo'yamiz. Biz **movie clip** sifatida quyinni rasmi tasvirlangan **movie clip**ni tanladik. Bu **movie clip**ning 1-4 gacha quyon oyoqlarini qimirlatib yuradigan holdagi tasviri o'rin olgan 5-8 gacha quyonning orqa tomondagi tasviri tasvirlanib bunda ham quyon harakati tasvirlangan.

4- kadrda

`gotoAndPlay(1);` 8-kadrda

`gotoAndPlay(5);`

buyruqlari yozilgan. Dastur quyidagicha ishlashi kerak. Agar pastga qaralgan yo'nalish tugmasi bosilsa, quyon pastga 3 pikselga ekranga qaragan holatda harakatlanib to'xtashi kerak. Agar tepaga qaralgan yo'nalish tugmasi bosilsa, quyon tepaga ekranga orqa tomonini o'girib 3 pikselga harakatlanib to'xtashi kerak. Agar o'ng va chap tugmalar bosilsa, quyon bundan oldin qanday holatda bo'lsa, shu holatda (ekranga qarab o'tirgan bo'lsa, ekranga qaragan holda, ekranga o'girilgan bo'lsa, o'girilgan holda) mos ravishda o'ngga va chapga 3 pikselga harakatlanib to'xtashi kerak. Demak quyon rasmi tasvirlangan **movie clip**ni ekranga o'rnatib, uni quyon deb nomlaymiz va unga quyidagi skriptni yozamiz:

```
onClipEvent (keyDown) {  
    if (Key.isDown(38)) { _y -= 3;
```



```

        _root.quyon.gotoAndPlay(5);
    }
    //yuqoriga qaragan tugma bosilganda if (Key.isDown(40)) {
        _y += 3;
        _root.quyon.gotoAndPlay(1);
    }
    //pastga qaragan tugma bosilganda if (Key.isDown(37))
    {
        _x -= 3; } //o'ngga qaragan tugma bosilganda if (Key.isDown(39)) { _x
        += 3;
    }
    //chapga qaragan tugma bosilganda updateAfterEvent();
}

```

Bu yerda Key.isDown(38) klaviaturadan qaysi tugma bosilganini tekshiradi.

Qavs ichidagi son o'rniga har bir tugmaning ASCII kodini yozish kerak. Flashda ko'p ishlatiladigan klavishlarning nomi konstanta deb e'lon qilingan. Qavs ichidagi son o'rniga klavishning nomini yozsak ham bo'ladi. Masalan bo'sh joy tugmasining bosilganligini rekshirmoqchi bo'lsak shartga Key.isDown(SPACE) deb yozishimiz mumkin. Dastur ko'rinishini o'zgartirish uchun unga quyosh va osmon rasmini chizamiz. Dastur quyidagi ko'rinishga ega bo'ladi.

Klaviaturadagi bosilgan ixtiyoriy klavishning skriptini bilib olish uchun ekranga biror **movie clip**ni hosil qilib unga quyidagi skriptni o'zlashtiramiz:

```

onClipEvent (keyDown) {
    trace(Key.getCode());
}

```

Bu yerda **Key.getCode()** - oxirgi bosilgan tugmaning skriptini qaytaradi.

Biror tugma bosilganda tezlanish bilan harakatlanuvchi **movie clip**ni hosil qiling.

Ekranga - joyidan qimirlamagan holda oyoq-qo'llarini harakatlantiruvchi, toshbaqa **movie clip**ini keltirib qo'yamiz. Yo'nalish tugmalari bosilganda bu toshbaqa harakatlanishi, agar tugma bosib turilsa, u tezlanuvchan harakatlanishi kerak. Toshbaqa **movie clip**iga quyidagi skriptlarni kiritamiz:

```

onClipEvent (load) { a = 0; b = 0;
}

```

a,b o'zgaruvchilariga 0 qiymatini o'zlashtiramiz. Bu o'zgaruvchilar, **movie clip**ning x va y o'qlari bo'yicha nechi pikselga siljishini aniqlaydi.

```

onClipEvent (enterFrame) { if
    (Key.isDown(38)) {
        b--;
    }
}

```

/*agar yuqoriga qaragan yo'nalish tugmasi bosib turilsa, b ning qiymati 1 pikselga kamayib boraveradi.*/

```

if (Key.isDown(40)) {
    b++;
}
if (Key.isDown(37)) {
    a--;
}
if (Key.isDown(39)) {
    a++;
}
_y += b;
_x += a;

```

*/*x va y oqlariga a va b ning qiymatlari qo'shiladi. Masalan agar yuqoriga qaragan yo'nalish tugmasi bosilgan qiymati 12 ga teng bo'lgunga qadar bosib turilsa, toshbaqa 12 piksel tezlik bilan yuqoriga qarab harakatlana boshlaydi. Agar pastga qaralgan yo'nalish tugmasi bosib turilsa, b manfiy qiymat qabul qiladi, y ga manfiy qiymat qo'shiladi va toshbaqa pastga qarab harakatlanib boradi.*/*

Dastur quyidagi ko'rinishga ega:



13-rasm. Obyektga tezlanish berish

Quyosh atrofida aylanuvchi yer tasvirini hosil qiling. Yerning tezligini va yo'nalishini o'ng va chap yo'nalish tugmalari orqali boshqarish mumkin bo'lsin.

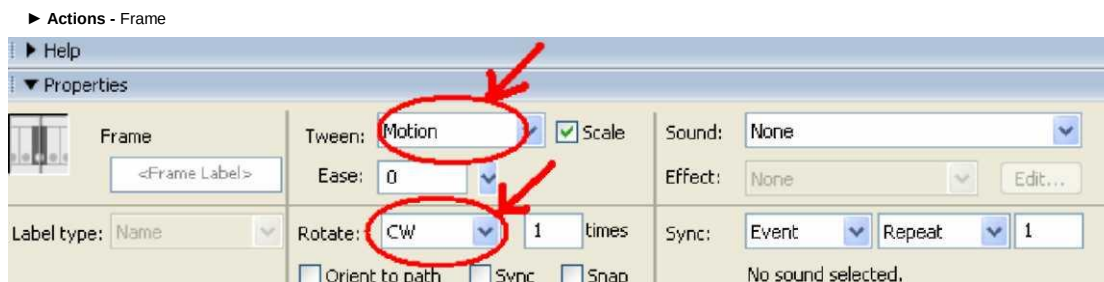
Yangi **movie clip**ni hosil qilamiz. Unga yerning rasmini keltirib qo'yamiz. Oynaga shu **movie clip**ni joylashtiramiz.. Uni berilgan nuqta atrofida aylanadigan qilib harakatlantiramiz. Buning uchun **movie clipni Tools** panelida joylashgan **H - free transform tool(q)** asbobi yordamida belgilab olamiz va uni markazida joylashgan nuqtani yerning tasviridan uzoqroq masofada joylashtiramiz:



o

14-rasm. Obyektni berilgan nuqta atrofida harakatlantirish.

Yuqoridagi rasmda ushbu nuqta yo'nalish tugmasi yordamida ko'rsatilgan. Yer va nuqta orasidagi masofa radius bo'lib yer shu radiusda aylana boshlaydi. 30-kadrni asosiy kadr deb ko'rsatamiz va qaytib 1-kadrga o'tamiz. **Properties** panelidagi "Tween" yozuvi oldidagi



15-rasm. Motion buyrug'ga muroiaat

Bunda yer markaziy nuqta atrofida aylana boshlaydi. ochiluvchi ro'yxatdan "Motion" yozuvini tanlaymiz. "Rotate" yozuvi oldidagi ochiluvchi ro'yxatdan "CW" yozuvini tanlaymiz.

Animatsiya o'z-o'zidan namoyish qilinmasligi uchun uni 1- kadrda to'xtatib qo'yamiz (buning uchun 1-kadrga stop() buyrug'ini kiritamiz). Oynaga yangi qatlam hosil qilib unga quyoshning rasmini ham keltirib qo' yamiz. Dasturni boshqarish uchun bizga bo'sh (hech narsani tasviri aks etmagan) **movie clip** kerak bo'ladi. Yangi **movie clip**ni hosil qilamiz va unga hech narsaning rasmini chizmay, uni ekranga, yangi qatlam hosil qilib, shu qatlamga olib qo'yamiz. **Actions** oynasini ochib, bo'sh **movie clip**ga quyidagi skriptni yozamiz:

```
onClipEvent (load) { k=1; n=1;
```

tf=_root._totalframes-1; } **k** va **n** ga boshlang'ich qiymat berib, tf o'zgaruvchisiga animatsiyadagi kadrlar sonidan 1 ta kam sonni o'zlashtiramiz. K o'zgaruvchisi rolikning tezligini boshqarish uchun, n esa namoyish qilinayotgan kadr raqamini saqlash uchun ishlatiladi.

```
onClipEvent (enterFrame) { n+=k;
    if (n>tf) {
        n-=tf }
    if (n<1) {
        n+=tf }
    _root.gotoAndStop(n);
    if (Key.isDown(Key.LEFT)) {
        k--;}
    if (Key.isDown(Key.RIGHT)) {
        k++;}
}
```

Dastur yordamida flash simvollari nusxasini hosil qilish

Biror obyektning nusxasini hosil qilish uchun, har bir obyektни oynaga 1 tadan keltirib qo'yish shart emas. Maxsus buyruqlar orqali biror obyektning bir necha nusxalarini hosil qilib bo'ladi. Masalan 1 dona tomchi tasviridan foydalanib, yomg'irni tasvirlash mumkin. Hosil qilingan nusxalarni o'chirish ham mumkin, lekin oynaga oldindan joylashgan obyektlarni o'chirishning imkoni yo'q. Oynada hosil qilingan har bir obyektning o'z nomi bo'ladi. Oynada joylashgan obyektning nusxasini hosil qilish uchun quyidagi buyruqdan foydalanamiz:

duplicateMovieClip("<a1>","<a2>",<a3>");

Bu yerda a1- asl nusxaning nomi, a2- hosil qilingan nusxaning nomi, a3-obyektning nusxasi joylashishi kerak bo'lgan stekning chuqurlik darajasi bo'lib son qiymat qabul qiladi. Ikkita klip ustma-ust tushganda stek darajasi past bo'lgan klip, darajasi yuqori bo'lgan klip ostiga qolib ketadi.

Oynada hosil qilingan nusxani yo'qotish uchun quyidagi funksiyadan foydalanamiz:

removeMovieClip("a1") Bu yerda a1-o'chirilishi kerak bo'lgan nusxaning nomi.

Yana bir funksiya bilan tanishib o'tamiz, bu funksiya bizlarga obyektning ixtiyoriy xususiyatini o'zgartirish uchun kerak bo'ladi. Bu funksiya quyidagi ko'rinishga ega:

setProperty("<a1>","<a2>",<a3>");

Bu yerda a1-obyekt nomi; a2- bu yerda o'zgartirilishi kerak bo'lgan xususiyat yoziladi. a3-xususiyatning qiymati; Endi Bu funksiyalarni amalda qo'llab ko'ramiz.

Ekranda yulduzli osmonni hosil qilish masalasini ko'ramiz.

Buning uchun dastlab **movie clip**ni hosil qilamiz. Unga yulduz rasmini chizamiz. Ishchi oynaga shu **movie clip**ni "yulduz" nomi bilan keltirib qo'yamiz. Avval 1 ta **movie clip**ni 2- nusxasini hosil qilib ko'ramiz. Buning uchun ekranga 2 ta tugma qo'yamiz. 1-tugma yashil rangda bo'lib, u bosilganda oynadagi yulduzning 2- nusxasi ko'rinsin, 2-tugma qizil rangda bo'lib, ushbu tugma bosilganda yulduzning nusxasi yo'qolsin. Avval 1 - tugmani belgilab **Actions** oynasini ochamiz va unga quyidagi skriptni yozamiz:

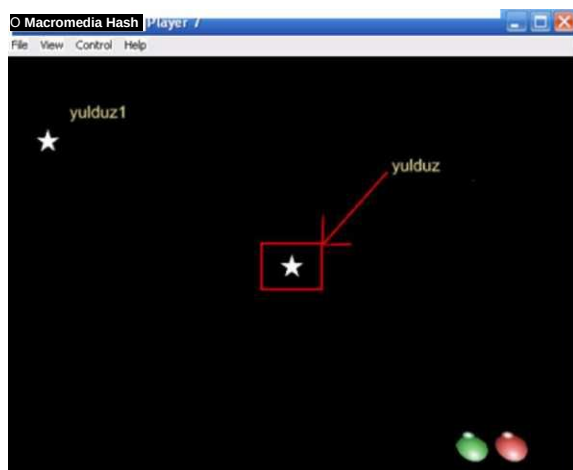
on (release) {

duplicateMovieClip("yulduz","yulduz1",1);} Lekin dasturni ishga tushirsak, hech qanday hodisa sodir bo'lmaydi. Aniqroq qilib aytadigan bo'lsak, sodir bo'ladi, lekin biz buni ko'rolmaymiz. Sababi biz hosil qilgan yuduz nomli **movie clip**imizning nusxasi "yulduz1"

nomi bilan yaratilyapti lekin **1-movie clip** bilan ustma-ust tushayapti. Nusxani hosil bo'lganiga aniq ishonch hosil qilish uchun uni avvalgi yulduzning koordinatalaridan biroz uzoqroqqa joylashtiramiz. Buning uchun skriptni quyidagicha yozamiz:

```
on (release) {  
  duplicateMovieClip("yulduz","yulduz1",1); setProperty("yulduz1",  
  _x, 40); setProperty("yulduz1", _y, 80);}
```

Dastur ishga tushgandan keyin, yashil tugma bosilganda, 1-yulduzning ikkinchi



16-rasm. Obyektning birta nusxasini hosil qilish

nusxasi hosil bo'ladi.

Endigi ishimiz qizil tugma bosilganda shu hosil qilingan nusxani yo'qotish. Buning uchun bu tugmaga quyidagi skriptni yozamiz:

```
on (release) {  
  removeMovieClip("yulduz1");}
```

Bunda yulduz1 nomli nusxalangan obyekt yo'qoladi.

Endi yashil tugma bosilganda 1 dona emas bir nechta nusxani hosil qilishni qarab o'tamiz. Buning uchun yashil tugmaga quyidagilarni kiritamiz:

```
on (release) {  
  for (i=1; i<=10; i++) {  
    duplicateMovieClip("yulduz","yulduz"+i,i);  
    setProperty("yulduz"+i, _x, 40*i); setProperty("yulduz"+i, _y,  
    80);}}
```

Biz for takrorlash operatori bilan oldin tanishib o'tganmiz. Uning dastlabki ikki qadamini qarab o'tamiz.

1-qadam: i o'zgaruvchisi 1-qadamda 1 qiymatini qabul qiladi. **duplicateMovieClip** "yulduz" nomli obyektning nusxasini yaratadi. Bu nusxani nomlayotgan vaqtda "yulduz" satrini oladi va unga i ning joriy qiymatini qo'shadi (i ning joriy qiymati 1 ga teng) "satr"+"son"="satr" bo'lgani uchun yulduz(satr)+1(son)=yulduz1(satr) bo'ladi. Demak hosil qilingan nusxaning nomi "yulduz1", Biz bu takrorlanish jarayoniga 10 ta yulduz hosil

qilamiz. Ularning steklari ustma-ust tushmasligi uchun har bir yulduzni alohida stekka qo'yib chiqamiz. 1-yulduzni birinchi, 2-yulduzni ikkinchi va hokazo. steklarga joylashtirib chiqamiz. "Yulduz1" nomli nusxamiz 1-stekka joylashgan. **duplicateMovieClip** ning vazifasi tugadi. `setProperty("yulduz" + i, _x, 40*i)`-bunda har bir nusxaning nomiga murojaat qilinyapti, 1-hol uchun yulduz satriga 1 sonini qo'shib yulduz1 obyektiga murojaat qiladi, so'ng uning x o'qi bo'yicha koordinatasining qiymati ($40*i$) ekanligi ko'rsatilyapti, ya'ni $40*1=40$ demak x bo'yicha koordinatasi 40 ga teng. y o'qi bo'yicha koordinatasani 80 deb olinadi. Hamma nusxalarning y o'qi bo'yicha koordinatasi bir xil bo'lganligi uchun, ular 1 qatorda joylashadi.

2-qadam: i o'zgaruvchining qiymati 2 ga teng bo'ladi. Yulduz obyektining yulduz2 nomli nusxasi 2-stekka yaratilib qo'yiladi. yulduz2 obyektning x o'qi bo'yicha koordinatasi 80 ($40*2=80$) ga teng bo'ladi. Uning y o'qi bo'yicha koordinatasi ham 80 ga teng.

' Qolgan qadamlar ham xuddi shu tartibda bajariladi va berilgan 1 dona yulduzning 10 ta nusxasi ekranga 1 qatorda joylashtirilib chiqiladi.

Qizil tugmaga quyidagi skriptni kiritamiz:

```
on (release) {  
    for (i=1;i<=10;i++) {  
        removeMovieClip("yulduz"+i);  
    }  
}
```

Bu bu takrorlanish jarayonida hosil qilingan 10 ta yulduzning hammasi o'chirilyapti. 1-qadamda i o'zgaruvchisining qiymati 1 ga teng bo'ladi va "yulduz1" nomli nusxa o'chiriladi, ikkinchi qadamda i ning qiymati 2 ga teng demak "yulduz2" nomli obyekt o'chiriladi va boshqalar. Dastur quyidagi ko'rinishga ega bo'ladi:



17-rasm. Obyektning n ta nusxasini hosil qilish

Biz tayyorlagan bu dastur yulduzli osmonga o'xshamaydi. Yulduzli osmonni tasvirlash uchun yulduzlar har xil koordinatada joylashishi kerak. Buning uchun oynaning bosh joyiga sichqonchani 1 marotaba cherib **actions** oynasini ochamiz va unga quyidagi skriptni yozamiz:

```
j=i;
while (j<=50) {
    yulduz.duplicateMovieClip("yulduz"+j,j);
    _root["yulduz"+j]._x=Math.random()*550;
    _root["yulduz"+j]._y=Math.random()*400;
```

Endi ushbu skriptni chuqurroq qarab chiqamiz. **While** opratori bilan oldindan tanishib chiqqanmiz. *j* o'zgaruvchisiga 1 qiymatni o'zlashtiramiz. Bu o'zgaruvchi takrorlash tanasini boshqarish uchun ishlatiladi. **While** operatoriga shart qoyilyapti. Agar *j* o'zgaruvchisining qiymati 50 dan kichik bo'lsa, takrorlash tanasi bajariladi. *j* sanagichning qiymati takrorlash oxirida 1 ta ga oshiriladi. duplicateMovieClip("yulduz"+jj) yulduz obyektining "yulduz+j" nomli nusxasini yaratadi. Bu funksiyaning ikkinchi ko'rinishi. Keyingi qadamda yaratilgan nusxaning koordinatalari aniqlanilayapti. Nusxalarning koordinatalarini setProperty funksiyasi yordamia aniqlasa ham bo'ladi, lekin biz shu usulni tanladik. _root["yulduz"+j] obyektga massiv sifatida murojaat qilishni taminlaydi _x esa x o'qi bo'yicha koordinatasini aniqlaydi. Math.random(); funksiyasi bilan biz oldin tanishib o'tganmiz. U bizga 0 va 1 oralig'idagi tasodifiy sonni qaytaradi va olingan natija mos ravishda 550 va 400 sonlariga ko'paytiriladi(ekranning o'lchami 550x400 bo'lgani uchun). Shunday tartibda ekranga berilgan 1 dona yulduzning 50 ta nusxasi ko'rinadi. Ekranda hosil qilingan 50 ta 1 xil nusxaning ko'rinishi unchalik chiroyli bo'lmaydi. Bizning vazifamiz betartib joylashgan ushbu yulduzlarning o'lchamlarini turli xil qilib ko'rsarishdir.

Buning uchun yuqoridagi skriptga takrorlash ichiga quyidagini qo'shib yozamiz:

```
aa=Math.random();
```

```

        _root["yulduz"+j]._xscale-=aa*15;
        _root["yulduz"+j]._yscale-=aa*15;

```

"aa" o'zgaruvchisiga [0;1] oraliqdagi tasodifiy son tanlanadi. Har yulduzning x va y bo'yicha koordinatalari aa*15 ga kamaytiriladi. **_xscale** va **_yscale** buyruqlari bilan oldin tanishib o'tganmiz. Xullas dastur har gal ishga tushganda yulduzlar oynaning turli tuqtalariga turli o'lchamlarda ko'rinib turadi. Yulduzlarning joylashish o'rni va o'lchamini yana boshqa kombinatsiyalarda joylashtirmoqchi bo'lsak, dasturni yana qayta yopib, ishga tushirish kerak. Bzga bu noqulay. Buni oldini olish uchun uni vaqtga bog'laymiz. Buning uchun ekrandagi ikkita tugmani olib tashlaymiz va "yulduz" nomli obyektimizni oyna chegarasidan tashqarida joylashtiramiz. Chunki keyinchalik har 2 soniyada yulduzlarning joylashish o'rni va o'lchami o'zgarib boradi, asl nusxa esa o'z joyidan jilmasdan turadi. Umumiy ko'rinishni buzmaslik uchun uni ko'rinmas holga keltirib qo'yamiz. Dasturni vaqtga bog'lash uchun Yulduzni belgilab unga quyidagi skriptni yozamiz:

```

onClipEvent (load) {
    q = getTimer(); } onClipEvent (enterFrame) { a =
    Math.floor((getTimer()-q)/1000); if (a>2) { j = 1;
    while (j<50) {
        _root.yulduz.duplicateMovieClip("yulduz"+j, j); _root["yulduz"+j]._x =
        Math.random()*550; _root["yulduz"+j]._y = Math.random()*400; aa =
        Math.random();
        _root["yulduz"+j]._xscale      -=      aa*15;
        _root["yulduz"+j]._yscale -= aa*15;
        j++;}
    q = getTimer();}

```

Sichqoncha ko'rsatgichi o'rniga sehirli tayoqcha hosil qilish

Dastlab tayoqchaning rasmi chizilgan **movie clip** hosil qilamiz. Dastur ishga tushganda sichqonch ko'rsatgichi ko'rinmas bo'lib, uning o'rniga shu tayoqchamiz ko'rinishi kerak. Buning uchun tayoqchaga quyidagi skriptni yozamiz:

```

onClipEvent (load) {
    Mouse.hide();
}
onClipEvent (mouseMove) {
    _x=_root._xmouse;
    _y=_root._ymouse;
    updateAfterEvent();
}

```

Bu skript bilan biz oldin tanishib o'tganmiz shuning uchun unga to'xtalib o'tirmaymiz. Biz sichqonchaning o'rniga harakatlanuvchi tayoqchani hosil qildik, lekin uning ko'rinishi sehirli tayoqchani eslatmaydi. Endigi qiladigan ishimiz tayoqcha harakatlanganda o'zidan

yurakchalar sachratib harakatlanishi kerak. Buning uchun o'ng tomonga 25 pikselga harakatlanuvchi yurakchani animatsiyasini hosil qilamiz, animatsiya o'z-o'zidan namoyon bo'lmasligi uchun animatsiyaning oxirgi kadriga stop(); buyrug'ini yozib qo'yamiz. Uni oyna chegaasidan tashqarida joylashtiramiz. Yuqoridagi skriptni quyidagicha o'zgartiramiz:

```
onClipEvent (load) { Mouse.hide();  
    i=0;}  
onClipEvent (mouseMove) {  
    _x=_root._xmouse; y=_root._ymouse;  
    _root.yurak.duplicateMovieClip("s"+i, i); _root["s"+i]._x=_root._xmouse;  
    _root["s"+i]._y=_root._ymouse; _root["s"+i]._rotation=Math.random()*360;  
    i++; updateAfterEvent(); }
```

Bu yerda:

_root.yurak.duplicateMovieClip("s"+i, i) - yurak obyektini <s1,s2,s3...> nomli nusxalarini hosil qiladi.

_root["s"+i]._x=_root._xmouse - hosil qilingan nusxalarga sichqonchani koordinatalarini o'zlashtiradi.

_root["s"+i]._rotation=Math.random()*360 - yurakning nusxasini tasodifiy burchakka buradi.

Agar nusxaga tasodifiy burilish burchagini bermasak harakatlanuvchi yurakchalar faqat 1 tomonga harakatlanadi. Shu buyruq tufayli sichqoncha harakatlanganda ko'rsatkich vazifasini bajaruvchi tayoqchadan yurakchalar ajralib chiqadi va ular har tomonga harakatlanib to'xtaydi. Lekin, dasturimizda katta xato mavjud sichqonchani biroz vaqt mobaynida harakatlantirib tursak, ekranda juda ko'p yurakchalar tasviri hosil bo'ladi va oyna yurakchalardan to'lib qoladi. Bu protsessorning normal ishlashiga to'sqinlik qiladi. Buning oldini olish uchun dastur oxiriga quyidagi skriptni qo'shib qo'yamiz:

```
if (i>=40) {i=0}
```

```
updateAfterEvent();
```

 Bunda ekrandagi yulduzchalarning soni 40 tadan oshmaydi.

41- nusxa yana qaytib s1 nomi bilan ekranda namoyon bo'ladi. Bunda agar sichqonchani harakatlantirsak va harakatlantirishdan to'xtatsak, tayoqchani uchidan sachrayotgan yurakchalarning oxirgi 40 donasi har xil tomonga harakatlana boshlaydi va harakat oxirida to'xtab qoladi. Chunki oxirgi kadrda stop() buyrug'ini yozib qo'yganmiz. Bizga oxirgi sachragan yurakchalar o'z harakat trayektoriyasini bosib o'tgandan so'ng ko'rinmas holga kelishi kerak. Buning uchun yurakcha **movie clip**ini ochamiz va uning

oxirgi kadridagi skriptni quyidagicha o'zgartiramiz:

```
stop();
```

`removeMovieClip(this);` Bunda yulduzcha harakatlanib bo'lgandan so'ng ko'rinmas bo'ladi. Ya'ni nusxa o'chiriladi. Tayoqchadan chiqayotgan yurakchalarning rangi har xil bo'lsa, uning ko'rinishi yanada chiroyli bo'ladi. Buning uchun oynadagi yurakchaga quyidagi skriptni yozamiz:

```
onClipEvent (load) { a=new Color (this);  
    a.setRGB(Math.random()*0xFFFFFF);  
}
```

Obyektni harakatlantirish texnologiyasi

Biz yuqorida Flash obyektlarini tugmalar, klaviaturadagi klavishlar orqali boshqarishni o'rgandik. Ammo foydalanuvchi uchun qulay yana bir texnologiya mavjud bo'lib, u **drag and drop** ya'ni ekrandagi obyektning sichqoncha orqali harakatlantirishdir. Bu bir qarashda oddiy bo'lib tuyilgan jarayon, Flashning kuchli vositalaridan biridir. Bu texnologiya: interfaol interfeyslarda, o'yinlarda va qiyin dinamik effektlar hosil qilishda qo'llaniladi. Bu jarayonni oddiy misollardan boshlab ko'rib chiqamiz. Oynada qizcha rasmi tasvirlangan **movie clip**, va uy tasvirlangan tugma berilgan. Sichqoncha orqali uyni harakatlantiring. Bizga ma'lumki, ekrandagi obyektning harakatlanishi sichqoncha tugmasi bosilgandan boshlanib, qo'yib yuborilganda to'xtaydi. Shuning uchun biz quyidagilardan foydalanamiz:

on (press) - obyekt ustida sichqoncha tugmasini bosgandagi holat.

on (release) - tugmani bosib qo'yib yuborilgandagi holat.

Harakatlanadigan obyektning esa tugma deb olish kerak. Uycha rasmi tasvirlangan tugmani va bitta **movie clip**ni kutubxonadan olamiz va ekranga joylashtiramiz. Dasturga chiroyli ko'rinish berish uchun, oynaga moviy osmon va quyoshni chizamiz. Yerni esa yashil rangga bo'yaymiz. Bu yerda quyosh, yer, osmon oynaga chizilgan rasm. Uycha rasmi tasvirlangan tugmani belgilab unga quyidagi skriptni yozamiz:

```
on (press) { startDrag("");  
    }  
on (release) { stopDrag(); }
```



18-rasm. Movie clipni sichqoncha orqali harakatlantirish

Bunda uycha ustiga sichqoncha tugmasini bosib turib harakatlantirsak, uycha bilan birga qizcha rasmi tasvirlangan **movie clip** ham harakatlanadi. Chunki, ularning ikkisi ham bitta vaqt shkalasida joylashgan.

Skriptning har bir qismiga to'xtalib o'tamiz.

```
on (press) {
    startDrag(""); }
```

Uycha rasmi tasvirlangan tugma ustida sichqoncha bosilganda uychni sichqoncha ko'rsatgichi ortidan harakatlantiradi. Tugmani harakatlantirish uchun uning ixtiyoriy nuqtasi ustida sichqoncha tugmasini bosib siljitsak u siljiydi. Uning faqat markazidan to'rtganda harakatlanishi uchun unga yana bir parametрни qo'shishimiz kerak. Yuqoridagi skriptga o'zgartirish kiritamiz:

```
on (press) {
    startDrag("",true);}
```

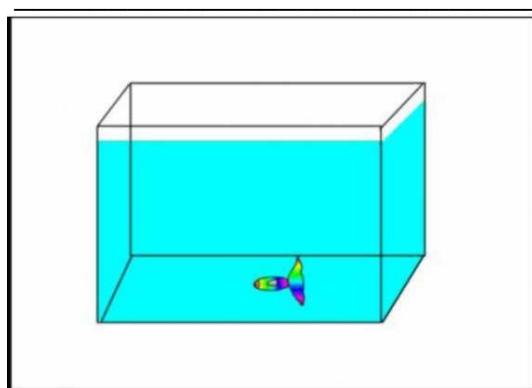
Dasturni ishlatib ko'rsak, qiziq holat ro'y beradi. Uycha ustida sichqoncha bosilganda, u bilan qizchaning rasmi tasvirlangan **movie clip** sichqoncha ko'rsatgichidan ancha uzoqqa siljiydi va ko'rsatgich bilan birga harakatlanadi. Sichqoncha qo'yib yuborilganda ham tugma bilan **movie clip**, ko'rsatgich ortidan harakatni davom ettiradi. Chunki **on (release)** holatining bajarilishi uchun sichqoncha tugma ustida bosilib, so'ng qo'yib yuborilishi kerak. Rasm va **movie clip** ko'rsatgichdan uzoqda harakatlanadi, shuning uchun ko'rsatgich tugma ustiga kelolmaydi. Bunga sabab, tugma filmning asosiy shkalasida joylashgan bo'lib, uning boshlang'ich koordinatalari oynaning yuqori chap qismi bo'ladi. Tugma dastur ishga tushganda oynaning yuqori chap burchagida joylashmagan. Shuning uchun **startDrag** ishga tushganida, markazlash qismi yoqilganda ya'ni, **startDrag("",true)** bo'lganda, uycha rasmi tasvirlangan tugma oynaning yuqori chap burchagidan qancha masofaga joylashgan bo'lsa, sichqoncha ko'rsatgichiga nisbatan ham shunday masofaga siljiydi. Bu hol noqulayliklarga

olib keladi. Bundan tashqari qizcha rasmi tasvirlangan **movie clip** ham tugma bilan birga harakatlanmasligi kerak. Bu noqulay effektlardan qutilish juda oddiy. Buning uchun uycha rasmi tasvirlangan tugmani asosiy diagrammadan olib tashlab, uni alohida **movie clip**ga joylashtirishimiz kerak. Tugmani **movie clip** ichiga joylashtirish uchun tugmani belgilab [F8] tugmasini bosamiz, shundan so'ng nomalab [ok] tugmasini bosamiz. Bunda tugma markazi, uning markaziy nuqtasi bilan ustma-ust tushadi. Uychaning ixtiyoriy nuqtasi ustida sichqoncha tugmasi bosilsa sichqoncha ko'rsatgichining uchi bilan uychaning markazi ustma-ust tushishiga e'tibor bering. Endi, yo'l qo'yilgan barcha kamchiliklar yo'qotildi.

Akvarium rasmi tasvirlangan **movie clip** bilan baliq rasmi tasvirlangan **movie clip**lar berilgan. Baliqni sichqoncha orqali harakatlantiring. Bunda baliq akvarium chegarasidan tashqariga chiqmasin.

Qo'yilgan masala shartiga ko'ra, baliq ustiga sichqoncha tugmasi bosib harakatlantirilsa, baliq ham sichqoncha bilan birga harakatlanishi kerak. Agar sichqoncha tugmasi bosib turilib, ko'rsatgichi akvarium chegarasidan tashqariga chiqarilsa, baliq harak qilmasligi kerak. Buning uchun avval ekranga to'rtburchakni chizib olamiz. "Info" bo'limi orqali uning yuqori chap va quyi o'ng nuqtalarini koordinatalarini bilib olamiz. Chizilgan to'rtburchakning yuqori chap qismida sichqoncha ko'rsatgichi keltirilsa "info" bo'limida shu nuqtaning koordinatalari ko'rinib turadi.

Biz chizgan misolda to'rtburchakning yuqri chap burchagining koordinatalari x o'qi bo'yicha 171, y o'qi bo'yicha 177ga quyi o'ng burchakning koordinatalari esa, x o'qi bo'yicha 366, y o'qi bo'yicha 294 ga teng. Endi shu to'rtburchak ustiga akvariumni shunday joylashtiramizki, berilgan to'rtburchak uning ostiga qolib, ko'rinmasin. Baliq keyinchalik shu to'rtburchak ichiga harakatlanadi. Endi baliqni akvarium ichiga joylashtiraiz:



19-rasm. Harakatlanish sohasida joylashtirilgan movie clip.

Baliqni belgilab unga quyidagi skriptni kiritamiz:

```
on (press) {
    startDrag("",true, 171, 177, 366, 294); }
```

```
on (release){  
stopDrag(); }
```

Endi baliq akvarium ichida harakatlanadi. Chunki **startDrag**(^{MM},true,171,177,366,294); - baliqning harakatini to'rtburchak soha bilan chegaralaydi. Endi baliqni akvarium ichida harakatlantira olamiz. Baliq **movie clip**ini o'zini, yanada chiroyli bo'lishi uchun, harakatlantirsa ham bo'ladi. Masalan **movie clip**da uni dumini yuqoridan pastga harakatlantirish mumkin. Bunda baliq akvarium ichida muallaq suzib turganday animatsiya hosil bo'ladi.

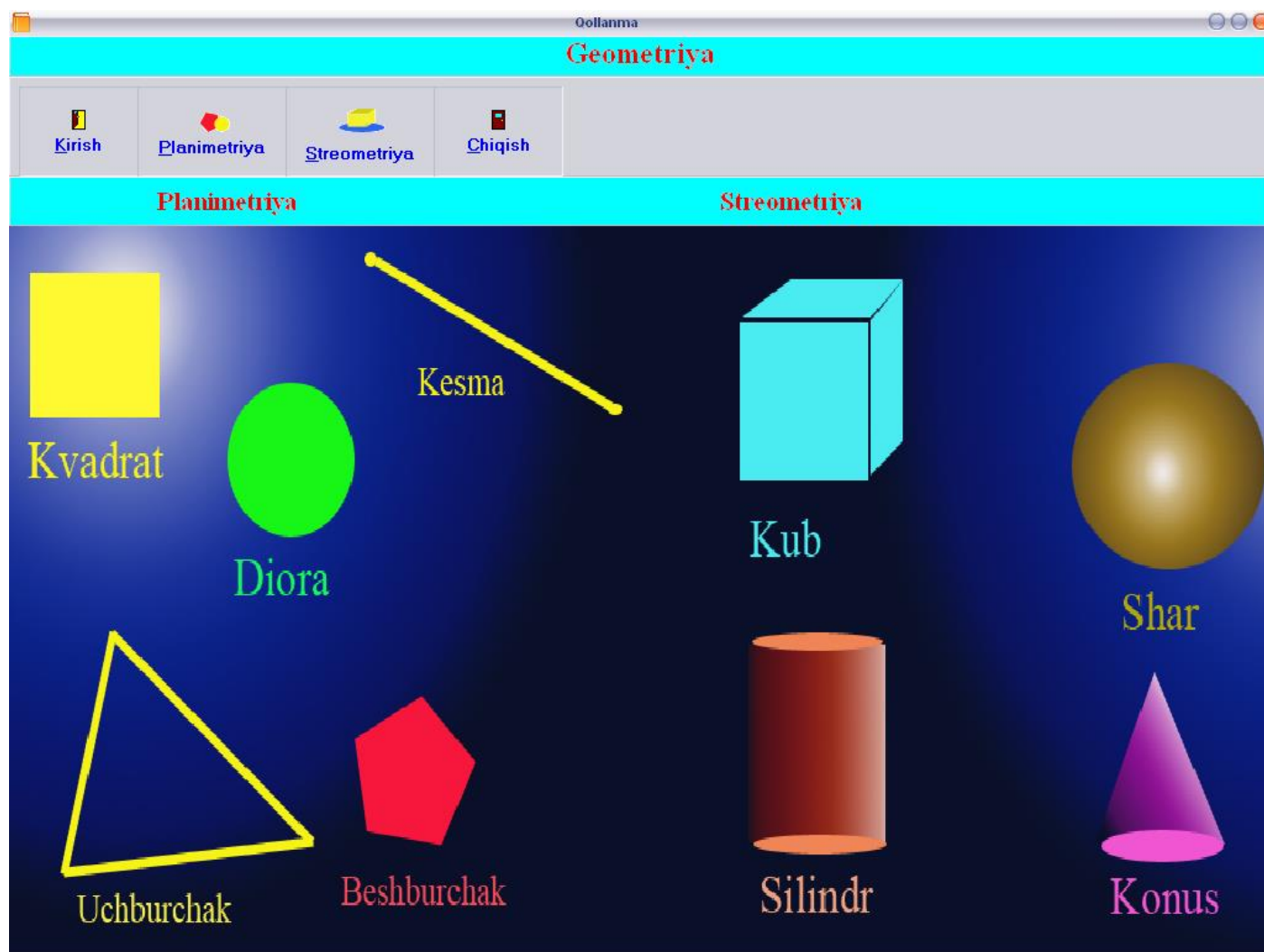
BIRINCHI BOBNING XULOSASI

Bu bobdan xulosa qilib shuni aytish mumkin, bu bobni o'qib o'zlashtirgan foydalanuvchi Flash oynalaridan va menyularidan foydalanish yo'llari haqida ma'lumot oladi. Masalan vaqt diagrammasida kadrlarni tasvirlash formatlari, xususiyatlar inspektori, foydalanuvchi interfeysni sozlash, klavishalarni belgilash, haqidagi menyular hisoblanadi. Flash dasturida foydalanuvchi o'zi qulay tarzda kerak bo'lgan komponentlar va buyruqlarni klaviatura yordamida tez va aniq amalga oshirilishi mumkin. Bu narsa foydalanuvchiga vaqtdan unumdor foydalanish imkoniyatini yaratib beradi. Flash (SWF) formati eng kuchli formatlardan biri bo'lmada (SWF) yaratuvchilarga grafikani imkoniyatlari, grafika bilan ishlovchi vositalari va natijani Web-sahifalarga qo'shish mexanizmlarni birlashtirish o'rtasida eng qulay yechimdir. Flash dasturini yana bir qulay ta'rafi shundaki kompyuter dasturlarida muammosiz ishlaydi. Bundan dasturlar hozirgi zamon talabiga javob beradigan dasturlardan iborat.

II bob Flash dasturi yordamida geometrik masalalarni tayyorlash.

2.1. Planimetriya bo'limiga doir masalalarni tayyorlash.

Malakaviy bitiruv ishini II bobini tayyorlashda Macromedia Flash, Delphi dasturi va web texnologiya usullardan foydalanib quyidagi ko'rinishda tayyorlandi.



1-rasm.Dasturning umumiy ko'runishi.

```
unit asos;  
interface  
uses  
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
  Dialogs, StdCtrls, Buttons, ExtCtrls, jpeg, ComCtrls, OleCtrls, SHDocVw;  
type  
  Tas1 = class(TForm)
```

```

SpeedButton1: TSpeedButton;
SpeedButton2: TSpeedButton;
SpeedButton3: TSpeedButton;
WebBrowser1: TWebBrowser;
SpeedButton4: TSpeedButton;
WebBrowser2: TWebBrowser;
SpeedButton9: TSpeedButton;
SpeedButton10: TSpeedButton;
SpeedButton11: TSpeedButton;
SpeedButton6: TSpeedButton;
ComboBox1: TComboBox;
ComboBox2: TComboBox;
procedure SpeedButton4Click(Sender: TObject);
procedure SpeedButton1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure SpeedButton2Click(Sender: TObject);
procedure SpeedButton10Click(Sender: TObject);
procedure SpeedButton11Click(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure ComboBox2Change(Sender: TObject);
procedure SpeedButton9Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  as1: Tas1;
S:STRING;
implementation
{$R *.dfm}
procedure Tas1.SpeedButton4Click(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  S:=S+'mundarija\Mundarija.mht';
webbrowser1.Navigate(S);
end;
procedure Tas1.SpeedButton1Click(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  S:=S+'reklama\reklama.swf';
webbrowser1.Navigate(S);
end;

```

```

procedure Tas1.FormCreate(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  S:=S+'reklama\reklama.swf';
  webbrowser1.Navigate(S);
  s:=ExtractFilePath(Application.ExeName);
  S:=S+'reklama\2\WWW.HTM';
  webbrowser2.Navigate(S);
end;

procedure Tas1.SpeedButton2Click(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  S:=S+'muallif\m.htm';
  webbrowser1.Navigate(S);
end;

procedure Tas1.SpeedButton10Click(Sender: TObject);
begin
  WebBrowser1.GoBack;
end;

procedure Tas1.SpeedButton11Click(Sender: TObject);
begin
  WebBrowser1.GoForward;
end;

procedure Tas1.ComboBox1Change(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  if ComboBox1.Text='1' then
    S:=S+'Mundarija\PL\1.swf';
  if ComboBox1.Text='2' then
    S:=S+'Mundarija\PL\2.swf';
  if ComboBox1.Text='3' then
    S:=S+'Mundarija\PL\3.swf';
  if ComboBox1.Text='4' then
    S:=S+'Mundarija\PL\4.swf';
  if ComboBox1.Text='5' then
    S:=S+'Mundarija\PL\5.swf';
  webbrowser1.Navigate(S);
end;

procedure Tas1.ComboBox2Change(Sender: TObject);
begin
  s:=ExtractFilePath(Application.ExeName);
  if ComboBox2.Text='1' then

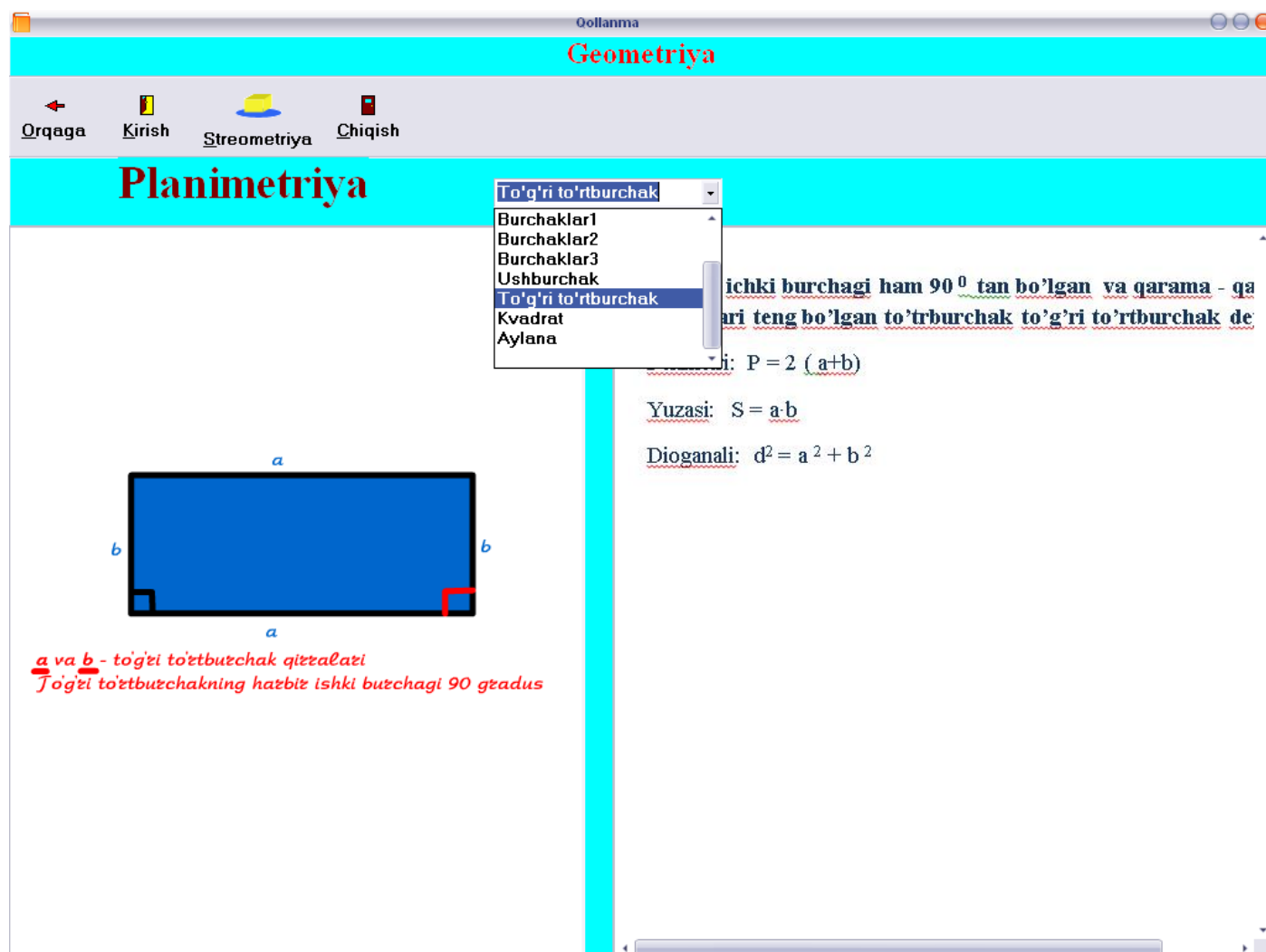
```



```

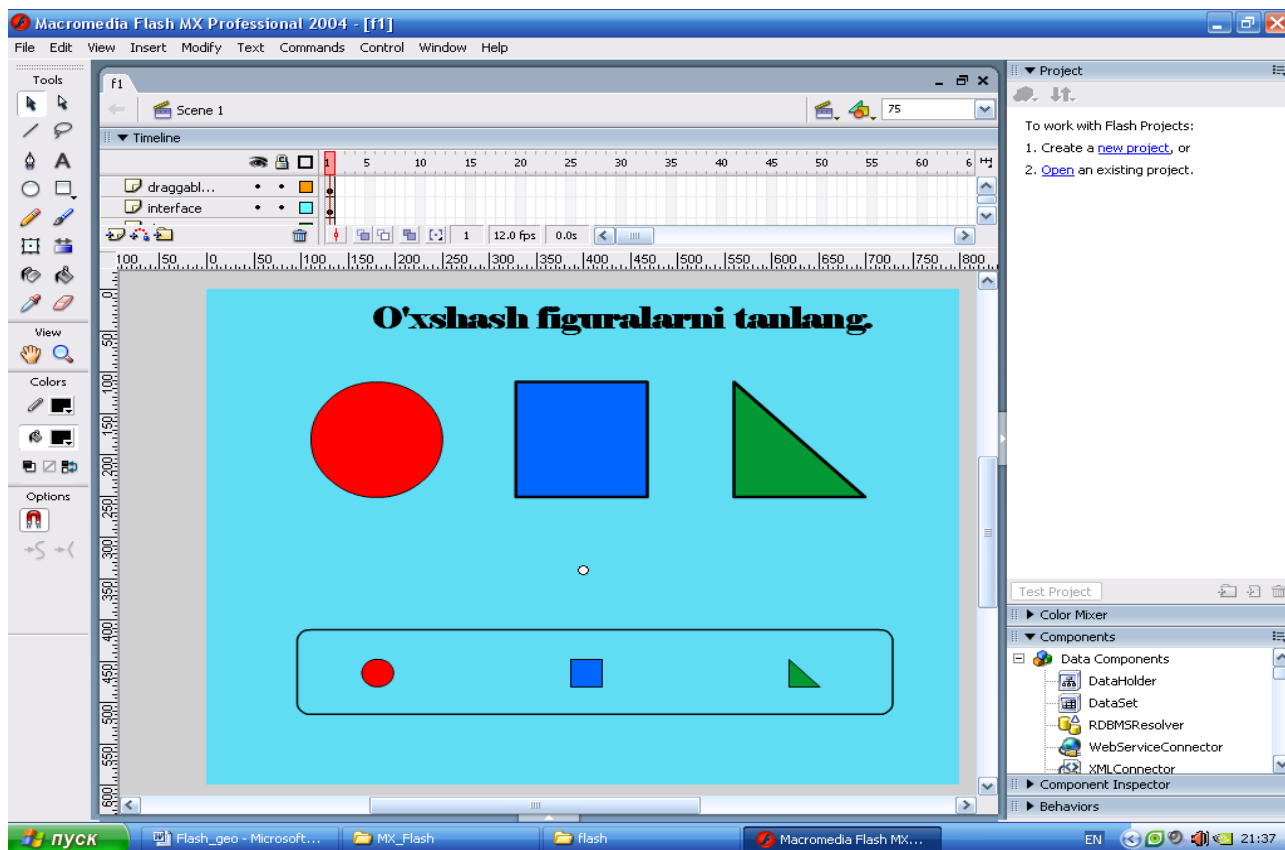
S:=S+'Mundarija\ST\figura.htm';
if ComboBox2.Text='2' then
S:=S+'Mundarija\ST\f2.swf';
if ComboBox2.Text='3' then
S:=S+'Mundarija\ST\f3.swf';
webbrowser1.Navigate(S);
end;
procedure Tas1.SpeedButton9Click(Sender: TObject);
begin
    s:=ExtractFilePath(Application.ExeName);
    S:=S+'Mundarija\adabiyot.mht';
webbrowser1.Navigate(S);
end; end.
Mualliflar bo'limi html tilida quyidagicha tayyorlandi.
<html>
<head>
<meta http-equiv="Content-Language" content="en-us">
<meta name="GENERATOR" content="Microsoft FrontPage 5.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>New Page 1</title>
</head>
<body bgcolor=#cccccc>
<bgsound src="2.mid " loop=2>
<p align=center> </p>
<p align=center> </p>
<script language="JavaScript1.2">
/*
Typing Text Script-
Last updated: 99/06/17
© Dynamic Drive (www.dynamicdrive.com)
For full source code, installation instructions,
100's more DHTML scripts, and Terms Of
Use, visit dynamicdrive.com
*/
var it=0
function initialize(){
mytext=typing.innerText

```

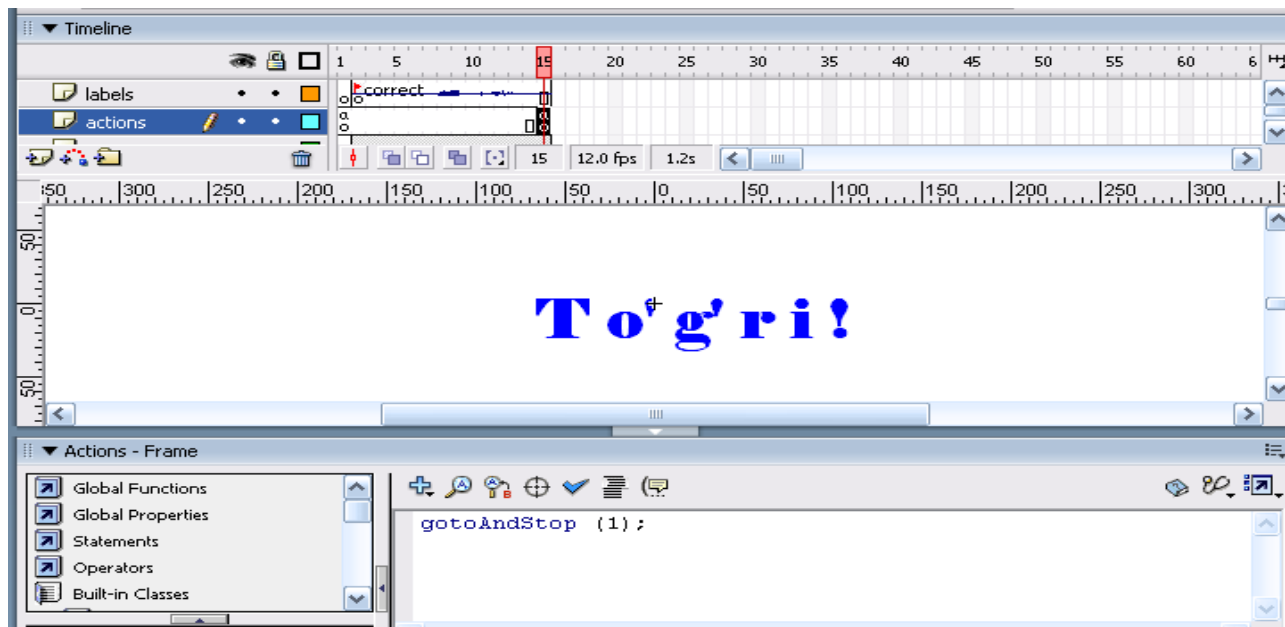
2-rasm Dasturda planimetriya bo'limi.

Bu bo'limda Flash va web texnologiya usullardan foydalanib tayyorlandi.

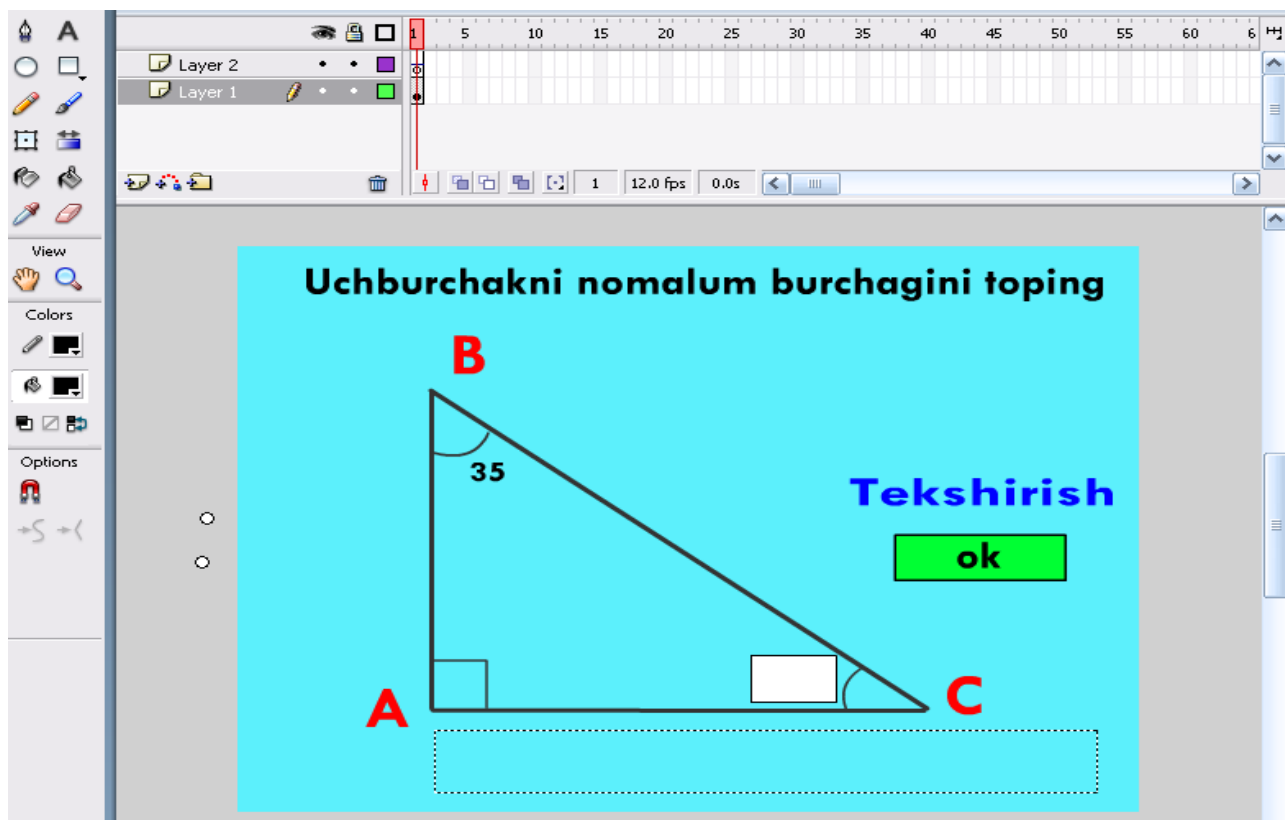


3-rasm. Planimetriya bo'limidagi 1-misolning ko'rinishi.

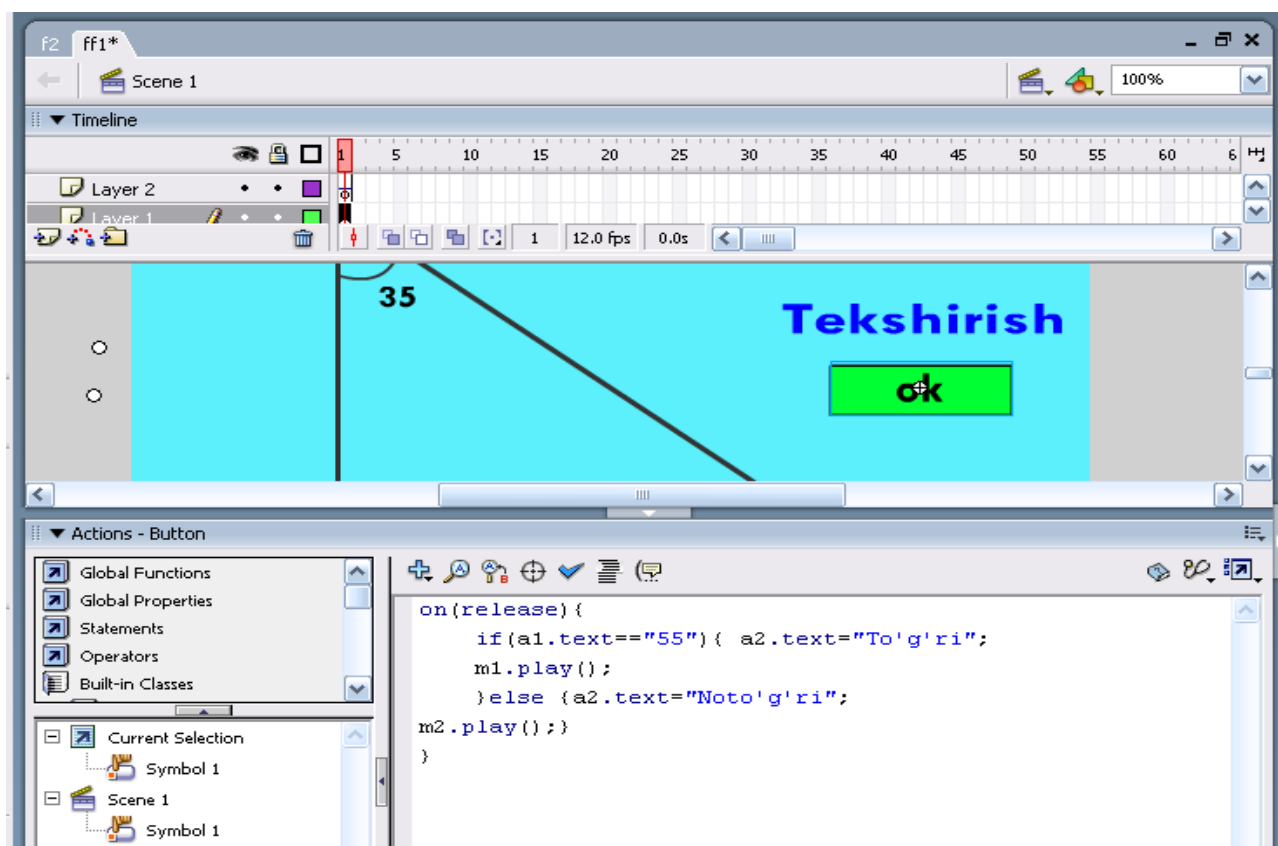
Bunda foydalanuvchi o'xshash figuralarni bir biriga mos qo'yib to'g'ri yoki noto'g'ri javobni oladi.



4-rasm .Action script ishlatish.



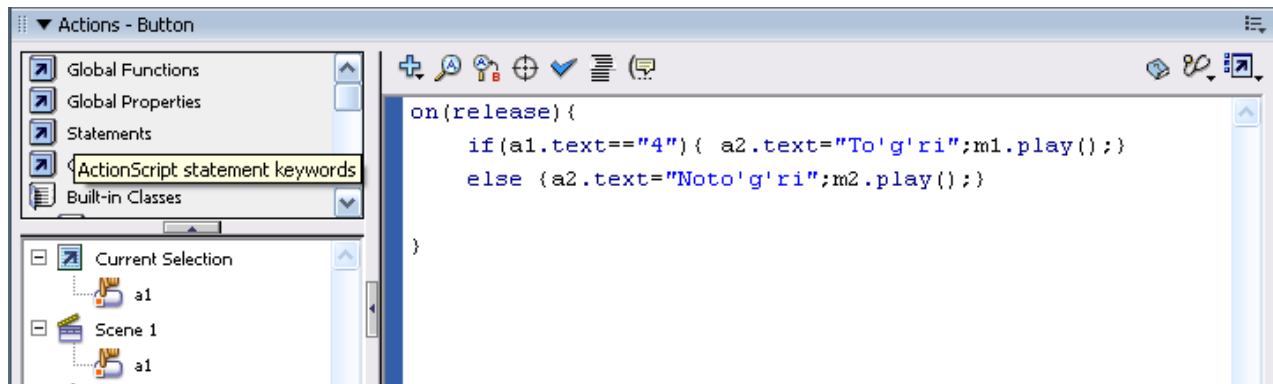
5-rasm. Planimetriya bo'limidagi 2-misolning ko'rinishi.



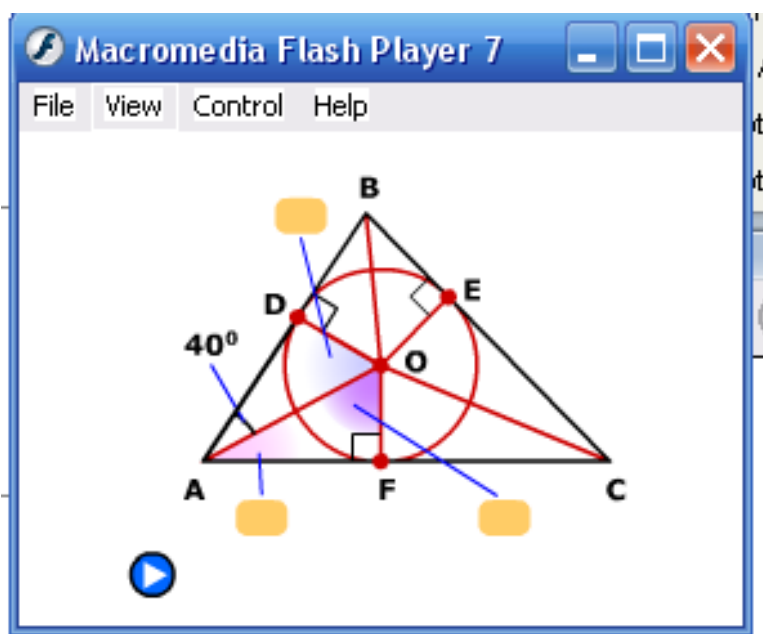
5-rasm.Ikkinchi masalada scriptlarni ishlatish.



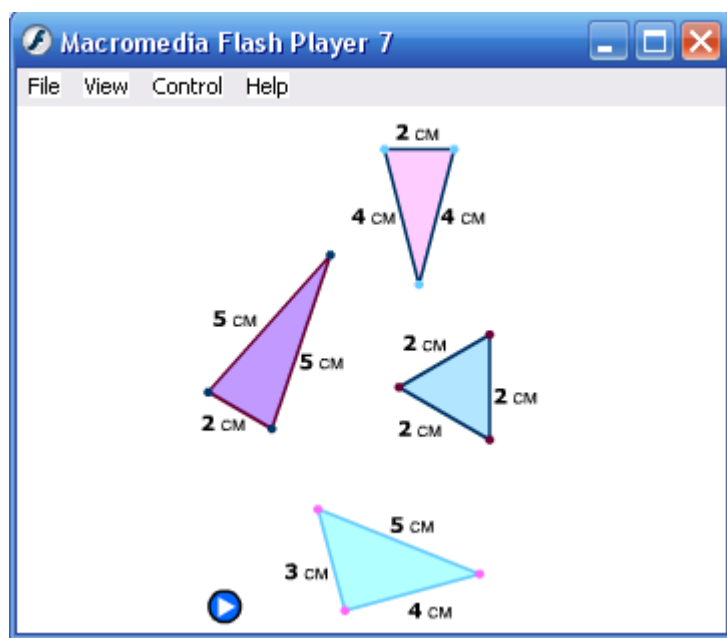
6-rasm. Planimetriya bo'limidagi 3-misolning ko'rinishi.



6-rasm.Uchinchi masalada scriptlarni ishlatish .



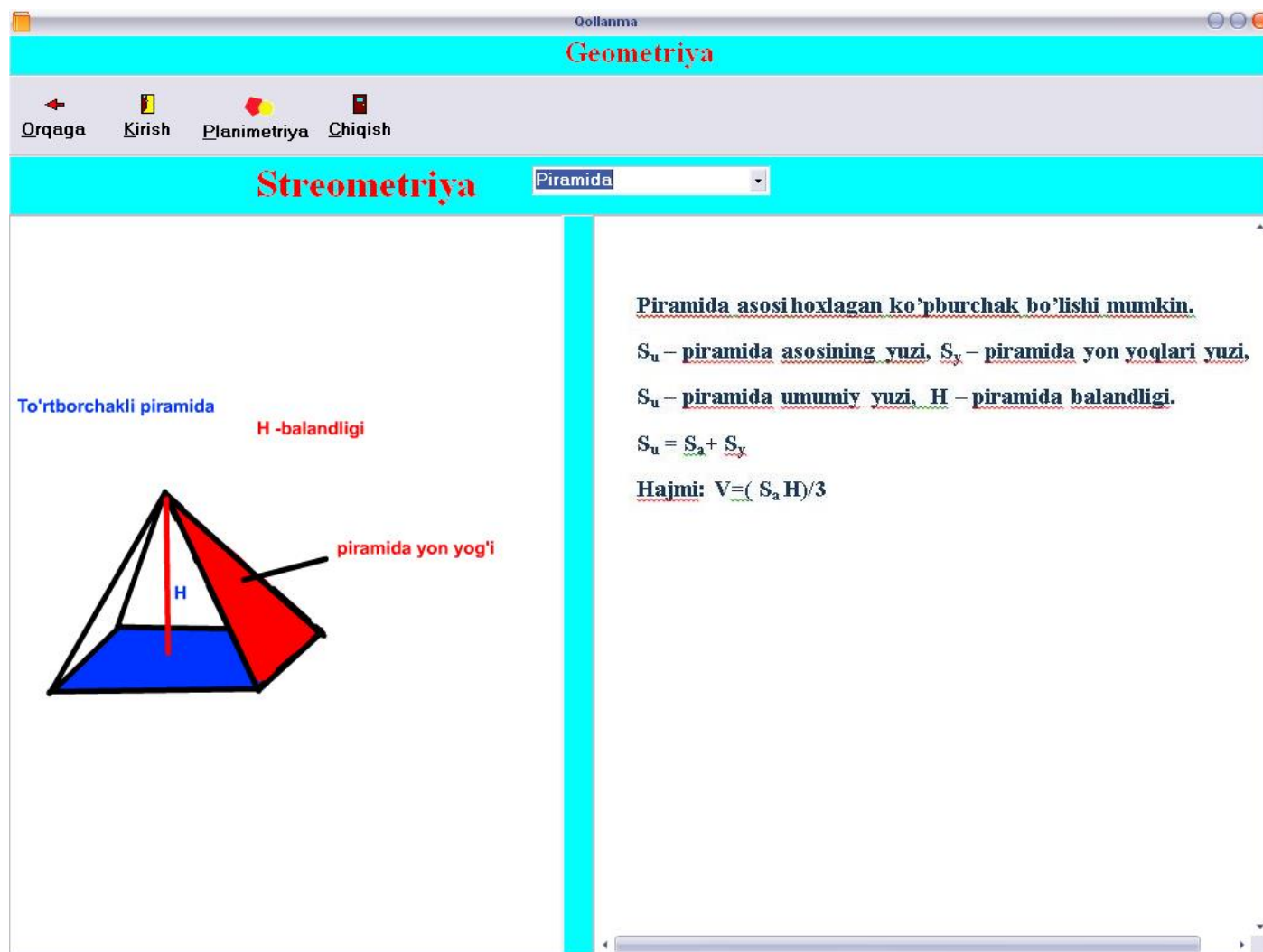
7-rasm. Planimetriya bo'limidagi 4-misolning ko'rinishi.



8-rasm. Planimetriya bo'limidagi 5-misolning ko'rinishi.

2.2.Stereometriya bolimiga doir masalalarni tayyorlash.

Bu bo'limda geometriya faning stereometriya bo'limiga doir bir nechta masalalar Flash dasturi yordamida virtual stand ko'rinishida tayyorlangan.Va har bir masala qanday tayyorlanganligi ketma ketligi keltirilgan.



9-rasm.Stereometriya bo'limidagi 1-masalaning ko'rinishi.

```
<html>
<head>
<meta http-equiv="Content-Language" content="en-us">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>KONUS</title>
<base target="rtop">
</head>
```



```

<body>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;</p>
<p align="left"><b><font size="5"><a target="rbottom" href="konus.swf">KONUS</a></font></b></p>
<p align="left"><b><font size="5"><a target="rbottom" href="piramida.swf">
PRAMIDA</a></font></b></p>
<p align="left"><b><font size="5"><a target="rbottom" href="cilindr.swf">SILINDR</a></font></b></p>
<p align="left"><b><font size="5"><a target="rbottom" href="sfera.swf">SFERA</a></font></b></p>
<p align="left"><b><font size="5"><a target="rbottom" href="kes_pir.swf">KESIK
PRAMIDA
</a></font></b></p>
</body>
</html>

```

```

<html xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schemas-microsoft-com:office:office"
xmlns="http://www.w3.org/TR/REC-html40">

```

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<link rel="File-List" href="ff2.files/filelist.xml">
<title>&#1053;&#1086;&#1074;&#1072;&#1103; &#1089;&#1090;&#1088;&#1072;&#1085;&#1080;&#1094;&#1072;
2</title>
<base target="rbottom">
<!--[if !mso]>
<style>
v\:*      { behavior: url(#default#VML) }
o\:*      { behavior: url(#default#VML) }
.shape    { behavior: url(#default#VML) }
</style>
<![endif]--><!--[if gte mso 9]>
<xml><o:shapedefaults v:ext="edit" spidmax="1027"/>
</xml><![endif]-->
</head>
<body>
<p align="center"><!--[if gte vml 1]><v:shapetype id="_x0000_t136"
coordsize="21600,21600" o:spt="136" adj="10800" path="m@7,l@8,m@5,21600l@6,21600e">
<v:formulas>
<v:f eqn="sum #0 0 10800"/>
<v:f eqn="prod #0 2 1"/>
<v:f eqn="sum 21600 0 @1"/>
<v:f eqn="sum 0 0 @2"/>

```

```

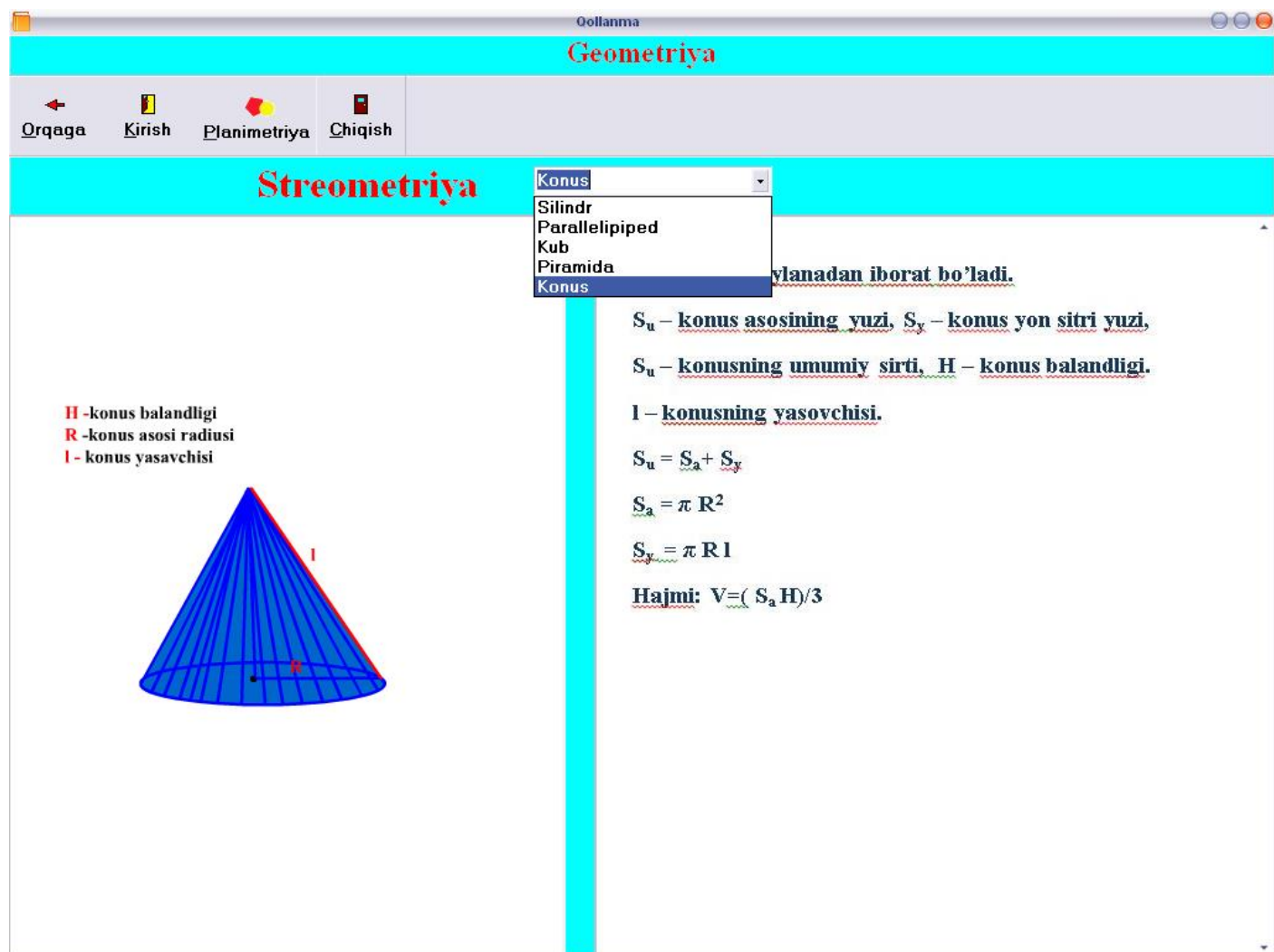
<v:f eqn="sum 21600 0 @3"/>
<v:f eqn="if @0 @3 0"/>
<v:f eqn="if @0 21600 @1"/>
<v:f eqn="if @0 0 @2"/>
<v:f eqn="if @0 @4 21600"/>
<v:f eqn="mid @5 @6"/>
<v:f eqn="mid @8 @5"/>
<v:f eqn="mid @7 @8"/>
<v:f eqn="mid @6 @7"/>
<v:f eqn="sum @6 0 @5"/>
</v:formulas>
<v:path textpathok="t" o:connecttype="custom" o:connectlocs="@9,0;@10,10800;@11,21600;@12,10800"
o:connectangles="270,180,90,0"/>
<v:textpath on="t" fitshape="t"/>
<v:handles>
<v:h position="#0,bottomRight" xrange="6629,14971"/>
</v:handles>
<o:lock v:ext="edit" text="t" shapetype="t"/>
</v:shapetype><v:shape id="_x0000_s1028" type="#_x0000_t136" alt="GEOMETRIK FIGURALAR"
style='width:585pt;height:30.75pt' fillcolor="red" strokecolor="blue">
<v:shadow on="t" color="#b2b2b2" opacity="52429f" offset="3pt"/>
<v:textpath style='font-family:"Times New Roman";v-text-kern:t' trim="t"
fitpath="t" string="GEOMETRIK FIGURALAR"/>
</v:shape><![endif]><![if !vml]><![endif]></p>
</body>
</html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>&#1053;&#1086;&#1074;&#1072;&#1103; &#1089;&#1090;&#1088;&#1072;&#1085;&#1080;&#1094;&#1072;
3</title>
</head>
<body>
</body>
</html>
<html>
<head>
<title>Новая страница 1</title>
</head>
<bgsound src="g1.mid" loop=1>
<frameset cols="150,*">
<frame name="left" scrolling="no" noresize target="rtop" src="ff1.htm">

```

```

<frameset rows="20%,*">
    <frame name="rtop" target="rbottom" src="ff2.htm">
    <frame name="rbottom" src="ff3.htm">
</frameset>
<noframes>
<body>
<p>Эта страница использует рамки, однако ваш обозреватель их не
поддерживает.</p>
</body>
</noframes>
</frameset>
</html>

```



10-rasm.Stereometriya bo'limidagi 2-masalaning ko'rinishi.

```

on(rollOver){
    f11._visible=true;
    n1.text="Prizma";
}

```

```

        on(rollOut){
            f11._visible=false;
            n1.text="Yo'q";
        }

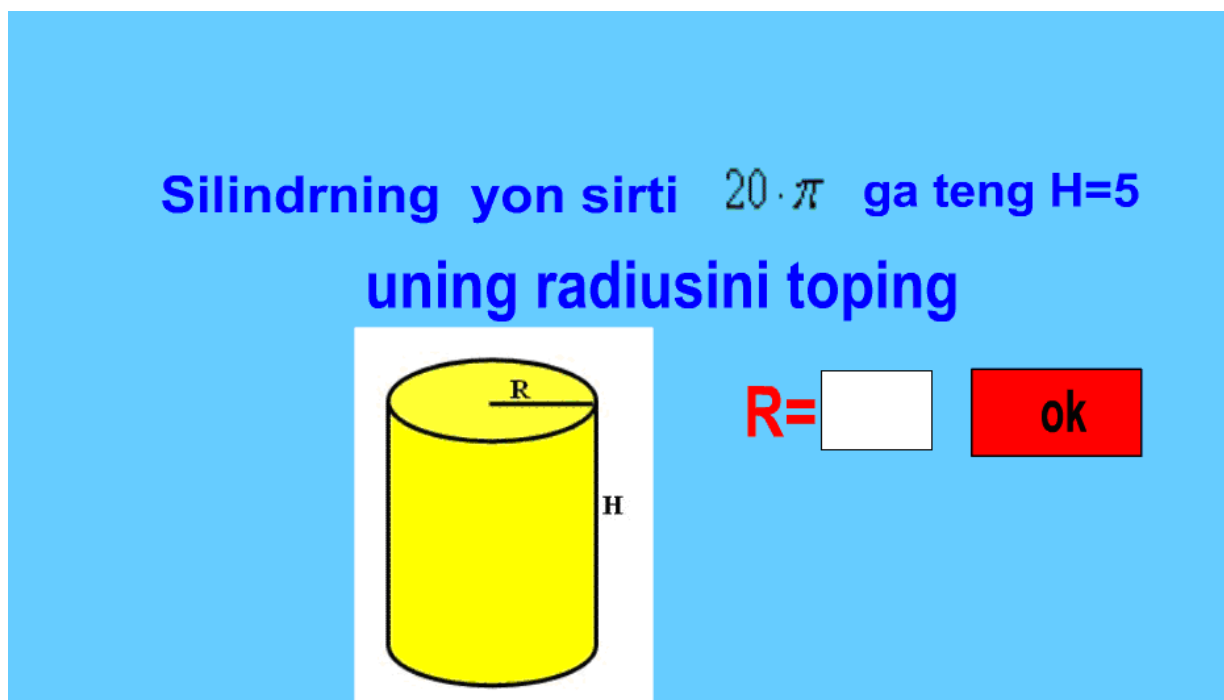
on(rollOver){
    f22._visible=true;
    n1.text="Silindr";}
    on(rollOut){
        f22._visible=false;
        n1.text="Yo'q";
    }

on(rollOver){
    f33._visible=true;
    n1.text="Konus";}
    on(rollOut){
        f33._visible=false;
        n1.text="Yo'q";
    }

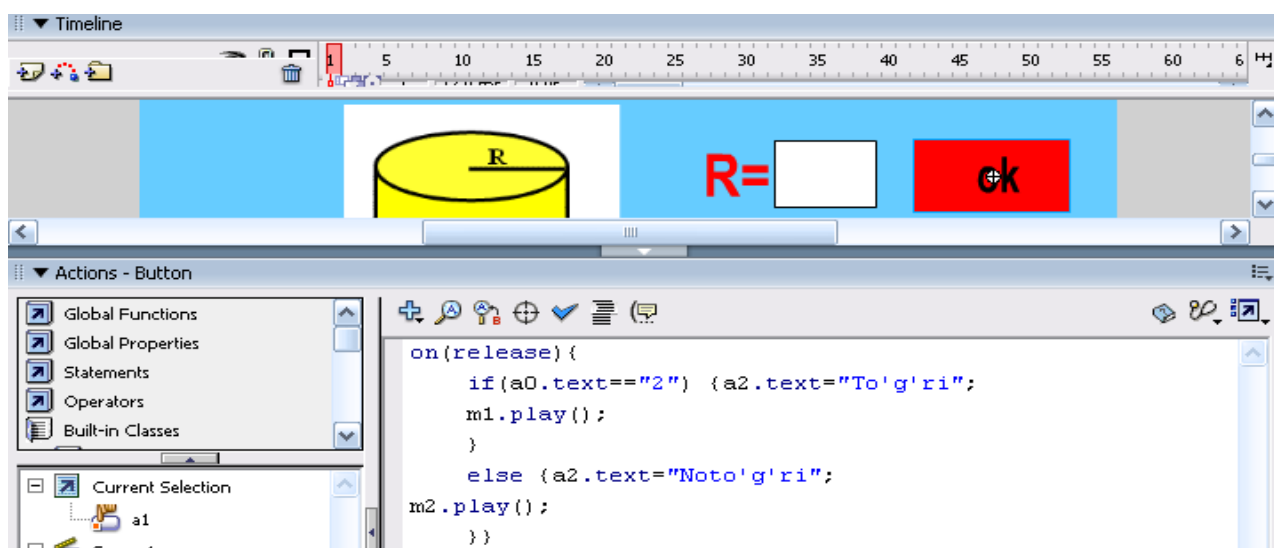
on(rollOver){
    f44._visible=true;
    n1.text="Pramida";}
    on(rollOut){
        f44._visible=false;
        n1.text="Yo'q";
    }

on(rollOver){
    f55._visible=true;
    n1.text="Sfera";}
    on(rollOut){
        f55._visible=false;
        n1.text="Yo'q";
    }

```



11-rasm.Stereometriya bo'limidagi 3-masalaning ko'rinishi.



12-rasm.3-masalaning ko'rinishi.

IKKINCHI BOBNING XULOSASI

Bu bobdan xulosa qilib shuni aytish mumkinki, bu bobni o'qib o'zlashtirgan foydalanuvchi Flash dasturi yordamida geometrik masalalarni tayyorlash haqida ma'lumot oladi. Masalan geometriyaning planimetriya hamda stereometriya bo'limlariga doir tayyorlangan masalalar haqidagi ma'lumotlar mavjud bo'lib, o'quvchi Flashda geometrik masalalarni qanday tayyorlanishi va uning dasturlari haqida ma'lumotga ega bo'ladi. Bu yaratilgan dasturdan geometriya darsini o'tishda, ko'rgazmali qurollar o'rnida foydalanish mumkin.

XULOSA

Bu BMIdan xulosa qilib shuni aytish mumkin, fodalanuvchi Flash ilovalaridan va menyularidan foydalanish yo'llari haqida ma'lumot oladi. Masalan vaqt diagrammasida kadrlarni tasvirlash formatlari, xususiyatlar inspektori, foydalanuvchi interfeysni sozlash, klavishalarni belgilash, haqidagi menyular hisoblanadi. Flash dasturida foydalanuvchi o'zi qulay tarzda kerak bo'lgan komponentlar va buyruqlarni klaviatura yordamida tez va aniq amalga oshirilishi mumkin. Bu narsa foydalanuvchiga vaqtdan unumdor foydalanish imkoniyatini yaratib beradi. Flash (SWF) formati eng kuchli formatlardan biri bo'lmada (SWF) yaratuvchilarga grafikani imkoniyatlari, grafika bilan ishlovchi vositalari va natijani Web-sahifalarga qo'shish mexanizmlarni birlashtirish o'rtasida eng qulay yechimdir. Flash dasturinig yana bir qulay ta'rafi shundaki kompyuter dasturlarida muammosiz ishlaydi. Bundan dasturlar hozirgi zamon talabiga javob beradigan dasturlardan iborat. Ikkinchi bobda Flash dasturi yordamida geometrik masalalarni tayyorlash haqida ma'lumot oladi. Masalan geometriyaning planimetriya hamda steriometriya bo'limlariga doir tayyorlangan masalalar haqidagi ma'lumotlar mavjud bo'lib, o'quvchi Flashda geometrik masalalarni qanday tayyorlanishi va uning dasturlari haqida ma'lulotga ega bo'ladi. Bu yaratilgan dasturdan geometriya darsini o'tishda, ko'rgazmali qurollar o'rnida fodalanish mumkin.

FOYDALANILGAN ADABIYOTLAR

1. O'zbekiston Respublikasi "Kadrlar tayyorlash Milliy dasturi" Barkamol avlod- O'zbekiston taraqqiyotining poydevori. Toshkent "Sharq" nashriyoti.
2. O'zbekiston Respublikasi Davlat ta'lim Standarti O'rta maxsus kasb-hunar ta'limi "Ma'rifat" gazetasi 2000-yil 4-noyabr N86-son.
3. Dranov Vladimir "Macromedia Flash MX. Sank-Peterburg 2003-yil. 620-628-bet.
4. Aripov M. "internet va E-mailda ishlash". Universitet nashriyoti 2006-yil 203-210-bet.
5. Begimqulov U.Sh, Mamarajov M.E, Tursunov S.K. "Flash MX dasturi va undan ta'limda foydalanish imkoniyatlari" Toshkent 2006-yil 23-42-bet.
6. [http: //Web-silver.ru/](http://Web-silver.ru/)
7. [http: //MX-Flash.ru/](http://MX-Flash.ru/)
8. Sayfullayeva H.M "Geometriya" kasb-hunar kollejlari uchun. Toshkent "O'qituvchi" 2000-yil 62-64-bet.
9. Isroilov I., Pashayev Z. "geometriyadan masalalar to'plami. "Akademik litseylar va kasb-hunar kollejlari uchun sinov darsligi. Toshkent "O'qituvchi" 2000-yil 35-38-bet.