

**ÓZBEKISTAN RESPUBLIKASÍ XALÍQ  
BILIMLENDIRIW MINISTRLIGI**

**ÁJINIYAZ ATINDAĞÍ NÓKIS MÁMLEKETLIK  
PEDAGOGIKALÍQ INSTITUTÍ**

**Aytimbetov Yusupbay Mambetyarovich**

**«Úsh ólshemli ob’ektlerdi programmalastırıwdı úyretiw metodikası»**

**5A110701– Tálimde xabar texnologiyaları**

**Magistr akademiyalıq dárejesin alıw ushın jazılğan**

**DISSERTACIYA**

**MAK da jaqlawğa ruxsat**

**Magistratura bólimi bashlıǵı:**

**p.i.k., doc. M.Allambergenova**

**Kafedra bashlıǵı:**

**f.-m.i.k. A.Tureniyazova**

**Ilmiy basshı:**

**t.i.d. B.Kaipbergenov**

**Nókis – 2017**

## MAZMUNI

Kirisiw.....	2
I-BAP. 3DS Max programmasında úsh ólshemli ob'ektlardi jaratıw hám onı eksport etiw.....	4
1.1. 3D Max Studio programması interfeysi.....	4
1.2. 3D Max Studio programmasında sistemalı blok modelin jaratıw .....	13
1.3. 3D Max Studio programmasında sistemalı blok modelin eksport etiw .....	29
II-BAP. C# ob'ektke baǵdarlangan programmalastırıw tilinde WindowsForm Application menen islesiw.....	30
2.1. C# ob'ektke baǵdarlangan programmalastırıw tili sintaksisi .....	30
2.2. C# tilinde ob'ektke baǵdarlangan programmalastırıw tili operatorları .....	37
2.3. C# tilinde WindowsForm Applicationda programma jaratıw.....	40
III-BAP. C# tiline TaoFrameWork bibliotekası járdeminde úsh ólshemli ob'ektlardi programmalastırıw járdeminde jaratıw,vizuallastırıw hám shaqırıw .....	50
3.1. TaoFramework bibliotekası ob'ektlari .....	50
3.2. C# ob'ektke baǵdarlangan programmalastırıw tiline úsh ólshemli ob'ektlardi jaratıw hám vizuallastırıw.....	52
3.3. C# ob'ektke baǵdarlangan programmalastırıw tiline úsh ólshemli ob'ektti shaqırıw hám programmalastırıw .....	64
Qosımshalar.....	71
Juwmaq.....	80
Paydalanılǵan ádebiyatlar.....	81

## Kirisiw

Zamanagoy xabar kommunikacion texnologiyasiniń hám olardı ámelge asırıw quralların júda tez rawajlanıwı xabar jámiyetiniń payda bolıwın aldınnan belgilep beredi. Bunday jámiyette insan iskerliginiń barlıq tarawlarında miynet etip atırǵanlardıń kópshiligi xabarlardı islep shıǵıw, saqlaw, qayta isle hám onıń eń joqarı formasi bilimlerdi tarqatıw menen shuǵıllanadı.

“Kompyuterlestiriwdi jánede rawajlandırıw hám xabar kommunikaciya texnologiyaların usınıs etiw haqqında” ǵı qararda xabar kommunikaciya texnologiyaları tarawında birinshi nábette programmalıq qurallardı, maǵlıwmatlar bazaların islep shıǵıw, respublika, tarmaq hám aymaqlıq xabar kommunikaciya texnikasını islep shıǵarıw tarawlarında islew ushın joqarı dárejeli qánigeler tayarlaw zárur degen waziypası sıpatında belgilep ótilgen.

«Bilimlendiriw haqqında»ǵı Ózbekstan Respublikası nızamı hám «Kadrlar tayarlawdıń Milliy baǵdarlaması» maqset hám waziypaların basqıshpa-basqısh ámelge asırılıwında zamanagoy xabar texnologiyaları hám sistemalarınıń róli júdá áhmiyetli bolıp kelmekte.

«Kadrlar tayarlawdıń Milliy baǵdarlaması» tolıq ámelge asıwdaǵı gózlengen maqsetlerdiń úlken ekenin esapqa alsaq, búgingi kúnge kelip, oqıtıw sistemalarınıń hár bir tarawında jańa texnologiyalardan paydalanıw zárurlıǵı ayırıqsha orında ekenligin seziw qiyin emes.

Házirgi kúnde mámleketimizde informaciyanı kompyuterlestiriw tiykarında júdá úlken jumıslar islendi, ómirimizdiń barlıq tarawlarında informaciylar menen islewdi jedellestiriw rejeleri «Kadrlar tayarlawdıń Milliy baǵdarlaması» da kórsetip ótilgen.

Bilimlendiriw processinde jańa texnologiyalar, pán hám texnika jetiskenliklerinen paydalanıw tayarlanıp atırǵan qánigelerdiń sıpatın qay dárejede ekenligin kórsetip beredi.

3d Studio MAX úsh ólshewli modellestiriw hám kórsetiw (vizualizaciya) jańa basqısh bolıp tabıladı. Bul programma járdeminde joqarı sıpatlı animaciya hám úsh ólshewli modellerdi professional dárejede jaratıw múmkin. Bunda siz eki ólshewli hám úsh ólshewli obektlerdi qollanıwıńız múmkin.

Bul programma járdeminde joqarı sıpatlı multiplikacion filmler, málim pánler boyınsha kórgizbeli programmalar jaratıw múmkin.

3D Studio MAX ta obektlerdi kóshiriw maydanın (viewport) ta jaratasız. Bunıń ushın siz kerekli asbaptı tańlap, kursordı kóriw maydanına keltirgenińizde kursor kórinisi ózgeredi. Mishka járdeminde obekttiń ólshewlerin beresiz. Jaratılğan obektlerde kino effektler jaratıw ushın arnawlı kamera hám jarıtuqısh asbapların qollawıńız múmkin. Obekt sırtı ushın hár túrli material tańlawıńız múmkin.

Kóri maydanında jaratılğan obektlerdi háreketlentirip, kishi animaciya payda etiw múmkin. Bunıń ushın (Animaciya) túymesin basıp, kadrlardı ózgertken jaǵdayda obektti háreketlentiriw menen aqırǵı kadrǵa alıp klinedi. Keyin, animaciya panelinen (Play) túymesini basıladı. Nátiyjede kadrlardan animaciya payda boladı. Bul jaratılğan animaciyanı fayl kórinisinde kompyuter yadında saqlaw hám qálegen video kórinislerdi oqıytuǵın programma járdeminde oqıwımız kerek. Fayl \*.avi keńeytpeli formatta saqlanadı.

### **Temanıń aktuallıǵı**

Házirgi waqıtta, zamanagoy kompyuter texnikası jaratılǵannan berli, olardı bilimlendiriw tarawında qollaw júdá áhmiyetli bolıp qaldı. Bilimlendiriwdiń dásturiy formaları, xalıqtıń jańa ásirdegi bilimge bolǵan talapların qandira almay qaldı. Sonıń ushın da, ashıq dep atalǵan, bilimlendiriwdiń jańa modeli júzege keldi. Bilimlendiriwdiń bul modeli jahanniń, bilim alıw processiniń hám insan maǵlıwmatınıń ashıqlıǵınan kelip shıǵadı.

## **I-BAP. 3DS Max programmasında úsh ólsheмли ob'ektlerdi jaratıw hám onı eksport etiw.**

### **1.1. 3D Max Studio programması interfeysi**

3D Studio Max versiyası 2017 programması Auto desk kompaniyasın jeke mülki bolıp esaplanadı hám onıń bólimlerinde islep shıǵılǵan – Piscneet firması tárepinen, Max 2017 programması úsh ólsheмли kompyuterdiń grafigi hám onı 3D grafigi dep ataydı (3Dinensional – úsh ólsheмли) tiyisli bolıp esaplanadı hám ayırıqsha súwretlerdi sintez etiw ushın mólsherlengen, saxnanı real ómirden yamasa oylap tabılǵan dunyalardı sıpat penen súwretli detal menen kórsetedi, bunnan tısqarı súwretlerdi, obektlerdi háreketlerdi hám onı animaciya dep ataydı hám sol kadrlar izbe-izligin támiyinleydi.

Ayırıqsha súwret hám animaciya jaratıwda Max 2017 programması menen támiyinlewshi bir neshe imkaniyatlar bar.

Olar: Qálegen úsh ólsheмли obektlerdi geometriyalıq formada modellesiwi – ápiwayıdan, sferaǵa deyin, tuwri múyeshli tábiyatta júz berip atırǵan obektler formasın, haywanlardıń denesi, terekler yamasa shayqalıp atırǵan suwdıń biyikligin kórsetiw múmkin.

Fizikalıq qásiyetlerdiń obekti, kópmúyeshli aynalı sáwleleniw hám jarıqlıq nurlarınıń sınırı, atmosferalıq hádiyseler, tuman yamasa tútin, tábiyat hádiyseleri, qar yamasa ot siyaqlı hádiyselerdi kórsetedi.

Úsh ólshewli saxnanı jaratıw imitaciyası qálegen sharayat ushın, shuqır kosmostan jariq kúnge deyin modelleniwshi obektlerin vizual real súwretli aynada hám sayası menen, bul fongá taslawshi esaplanadı.

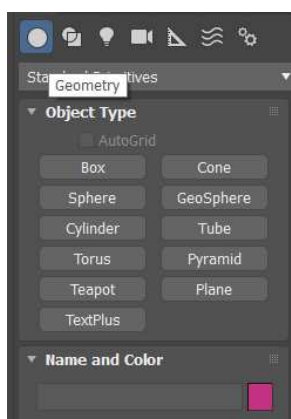
Barlıq parametrler obekti animaciyası onıń forması, ólsheми, jaǵdayı, reńi hám xarakteristikalıq materiallar hám taǵı baqalar menen basqarıw hár qıylı jolların reallastırıw yamasa animaciya jaǵdayında obektler qásiyetlerin ózgertiw, hár qıylı túrdegi háreketti imitaciya imkaniyatın támiyinlewshi menen baylanıslı.

### **3DS Max 2017 ob'ekti**

Obekt ataması (object) Max 2017 de úsh ólshemli dunyanı qálegen virtual elementlerine tiyisli, saxna quramına qosılıwı múmkin hám olarǵa modifikatorlar qollanılıwı múmkin.

MAX 2017 obektleri kategoriyalarǵa bólinedi, Bul kategoriyalardıń jetewi bar: Geometry (Geometriya), Shapes (formalar), Lighc (jarıqlıq deregi), Cameras (kameralar), Helpers (járdemshi obektler), Space Warps (kólem deformaciyası) hám Systems (sistemalar), bunnan tısqarı obekttiń sol kategoriyalarǵa tiyisli bolmaǵan úsh túri bar. Editable Spline (redaktorlawshı splayn). Editable Patch (redaktorlawshı bólek) hám Editable Mesh (redaktorlawshı setka). Sol úsh túrden tısqarı qalǵan barlıq MAX 2017 obektler parametrli esaplanadı, yaǵnıy jaratılıp atırǵanda anıq xarakteristikalıq parametrler toplamın payda etedi, obekttiń kordinataları jaǵdayı sıyaqlı onıń uzınlıǵı boyınsha ólshew, eni hám biyikligi, segmentler sanı yamasa tárepleri hám taǵı basqa. Bul parametrlerdı keyin ańsat ózgertiw múmkin, sonıń ushın obektlerdi jaratıw waqtında joqarı anıqlıq shárt emes. Tómendegi tiykarǵı MAX 2017 obektler klassifikaciyası keltiriledi. Isletiletuǵın obektler túri, sanı, qosımsha modullerden paydalanıwdı esapqa alıw múmkin.

**Geometry** kategoriyası bul kategoriyaǵa úsh ólshemli saxnanı geometriyalıq modeldi qurıwǵa mólsherlengen obektler kiredi hám tómendegilerge bólinedi:



Standard Primitives (standart primitivlar) – bul durıs geometriyalıq formadaǵı úsh ólshemli parallelepiped, sfera yamasa tor standart primitivlar sanına obektta kiredi, ol ózinde sháynektiń murnı menen kóz aldımızǵa keledi, qolı hám qappaǵı menen birgelikte.

Extended Primitives (Jaqsilangan primitivlar) – Standart primitivlarğa qarağanda bul úsh ólshemli ózek qıyınraq formağa iye boladı hám parametrlardi úlken sanı menen xarakterlenedi. Olarğa sonday obektler kiredi: parallelepiped, cilindr, kópmúyeshli hám tağı basqa.

Compound Objecc (dúzilgen obektler) – bul úsh ólshemli ózek, bir neshe ápiwayı obektlerden dúzilgen qağıydağa qarap, primitive – obektler bolıp tabıladı. Buğan Lof (laftingli) túrdegi obektler kiredi – úsh ólshemli ózek olar lafttıń metodı menen qurıladı, yağnıy qabıq qurılıs jolı menen berilgen traektoriyaúa jaylastırıladı.

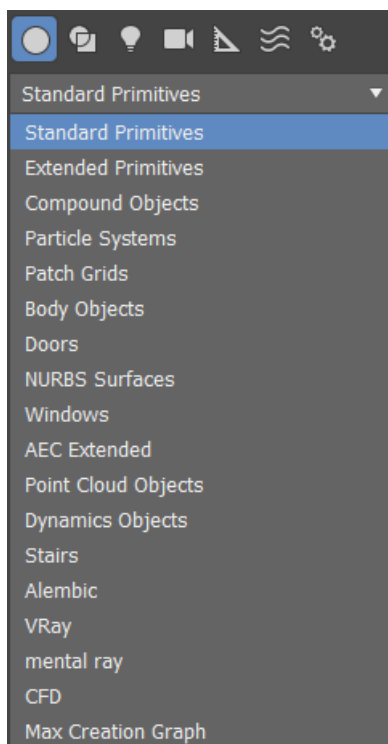
Particle Systems (qosımsha sistemaları) – bul kóplik mayda ekiqırlı yamasa úshqırnı qosımsha Sistema bolıp tabıladı.

Patch Grids (bólekler setkası) – bul biyiklikler Beze bóleklerinen ibarat hám tuwrı múyesh forması tegisligi fragmentine uqsas baslanğısh jaratılğan. Keyin bunday biyikliklerdi forma hám qıyalıǵı tochkaların basqarıwshı manipulyaciya boyınsha qadaǵalanıwı múmkin. Beze bólekler setkası járdeminde qolaylı ózgeriwshi qıya járdeminde biyiklikti modellestiriw qolaylı.

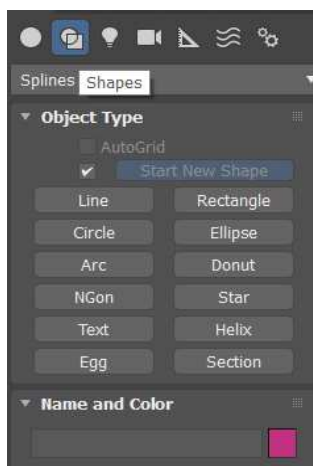
NURBS Surfaces (NURBS-biyikligi) – bul biyiklikler onıń forması racional B-splaynlar menen kóriledi (Non-NURBS Surfaces (Nurbis – biyiklikleri)). Nurbis – biyikliklerine baylanıslı bolğan jaǵdayda olar yamasa kosmosta berilgen saxnalardıń barlıq noqatlarınan ótiwi múmkin, yamasa qolaylı bolıwı múmkin. Bunday biyiklikler qıyın formadaǵı obektlerdi modellestirige saykes keliwi múmkin, jansız hám janlı tábiyatqa saykes jaǵdayda.

Dynamics Objecc (dinamikalıq obektler) – Bul standart obektlerdiń specifikalıq hár túrliligi eki túrdegi mexanikalıq qurılımlardı ańsat modellestiriwge ruxsat beredi; prujina hám amortizatorlardı. Bul obektler óziniń real prototipine kórinis boyınsha uqsaslıǵı úana emes, animaciya da olar tásir qılıp atırğan kúshke durıs jandasadı.

Doors (esikler) hám Windows (aynalar)- standart obektlerdiń hár qıylılıǵı, qıyın arxitekturalı elementlerdi ańsat modellestiriwge ruxsat beredi. Hár qıylı túrdegi ayna hám esikler, animaciya dawamında ashılıp jabılıwı múmkin.



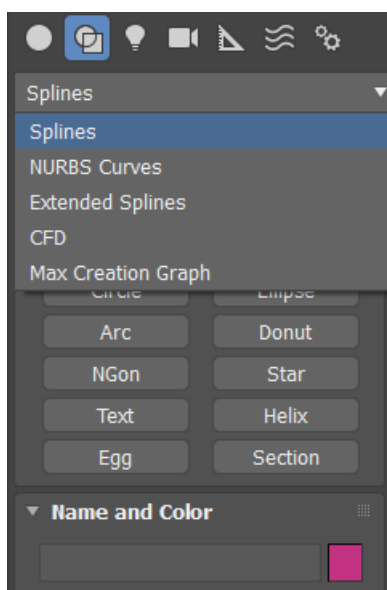
**Shapes** obekti kategoriyalari bul kategoriyaga hár qıylı sızıqlar bolğan qavariq, qavariq bolmağan ekiqıralı figuralardı payda etedi. Sızıqtıń bazı bir túrleri (máselen, spiral) olar tekislikte jaylaspawı múmkin. Úsh ólshemli keńislikte jaylasıwı múmkin. Formalar tayarlıqta xizmet etiwı múmkin.



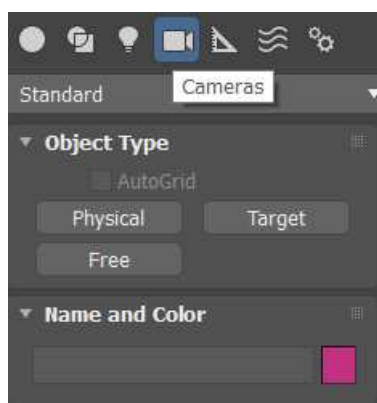
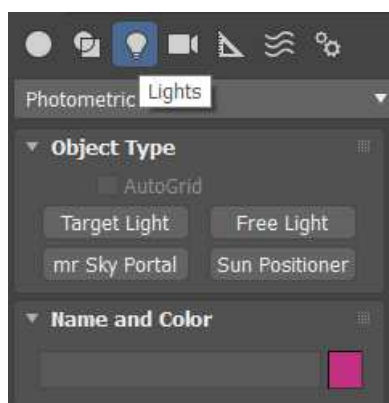
Splines (splaynlar) – eki kósherli geometriyalıq figuralar, (máselen: tuwrı múyesh, ellips yamasa juldız) hám sızıqlar qálegen qıyada hám konturlar tekstli simvolları bolıp tabıladı.



NURBS Curves ( NURBS – qıyalar)- bul sızıqlar hár qıylılıǵı, sızıwshılardı qurıwǵa ruxsat beriwshi, sınıq qıyalarǵa iye emes. NURBS- qıyaları keńisliktiń saxnasında belgilengen noqatlardan ótedi yamasa olardı ańsat iyeleydi.



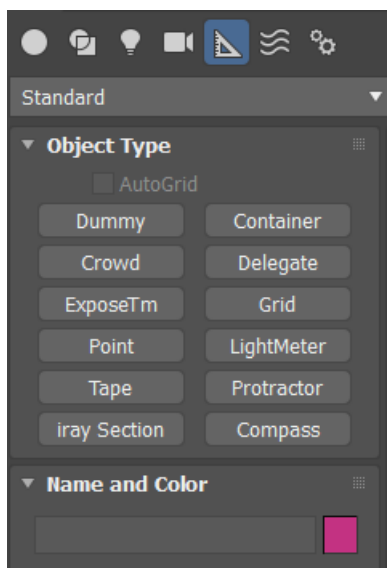
**Light** hám **Cameras** kategoriya obektleri bunday kategoriyalarǵa saxnanı jaratıwshı hár qıylı derekler imitaciyaǵa mólsherlengen hám syomkalı kameralar obektivi arqalı saxnanı gúzetiw ushın mólsherlengen. Bizde ekranda MAX 2017 obekti qanday kóriniste bolıwı kórsetilgen.



Jarıqlıq deregi hám kameralar qaratılǵan hám erkin bolıwı múmkin. Baǵdarlangan jarıqlıq deregi bul kameralar noqatlı obekttiń gózlegenı menen parıqlanadı. Erkin jarıqlıq derekler hám kameralar mólsherge iye emes.

**Helpers** kategoriya obektlari juwmaqlawshı saxna súwretin esapqa almastan hám onıń modellestiriw yamasa animaciyanı qısqartırıw ushın mólsherlengen hám bul obektler sol kategoriyaǵa kiredi. Járdemshi obektler tómendegi túrlerge bólinedi.

Standard (standartlar)- bul obektlerdi qayta islewde hám animaciyanıń geometriyalıq modelinde járdemshi waziypanı orınlaydı.



- Atmospheric Apparatus
- CameraMatch
- VRML 1.0/VRBL i VRML97

**Space Warps** obekti kategoriyası geometriyalıq modeli yamasa atomlar hár qıylı kúshler immitacion tásirge mólsherlengen obektler sol kategoriyaǵa kiredi, kúshlengen maydan tásirine túsedı.



Bul awırlıq kúsh tásin immitaciya etiwge ruxsat beredi yamasa tekstti atomlar sistemasına yamasa tegis obektin maydanın deformaciyalaw suwdı súwretlewshi vektorlı tolqınlar yamasa sheńberlerdi taslangan tastan modellestiriw ushın. Kólemli

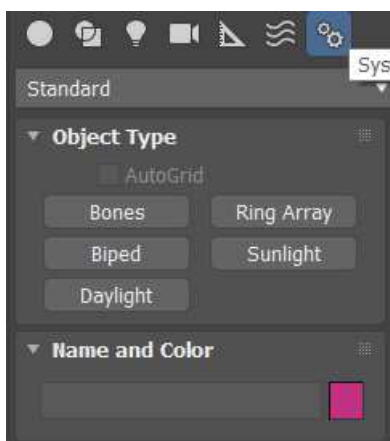
derekler deformaciyası aynada súwretlenedi. Shártli belgiler kórinisindegi proyeksiyalar juwamaqlawshı saxna, saxna súwretine kiritilmeyde. MAX 2017 kólemli deformaciyalardıń hár qıylı túri bar.

- Geometric/Deformable (deformaciyalanğan geometriya) hám Modifier-Based (modifikatorlar dereginde) – hár qıylı jol menen kólemli geometriylıq modeldi deformaciyalaw mómkin:

- Particles & Dynamics (bólekler hám dinamika) hám Particles Only (tek bólekler) - sistemalı bóleklerde ayırıqsha bólekler kúshli tásir ótkizedi, awırlıq kúshi yamasa samalda immitacion tásiiri ushın isletiledi.

- Dynamics Interface (Dinamikalıq interfeys) dinamikalıq sistemaǵa tásir etiw ushın xızmet etedi (MAX 2017 obektler toplamı dep ataladı).

**System** obekti kategoriyası hár bir sistema kóğlegen bir-biri menen baylanışlı obektlerdi óz ishine aladı, tárepler qosındısı, sistemanıń animaciyasın támiyinleydi. Bul kategoriya basınan óz ishine obektler qosıw ushın mólsherlengen, olar qosımsha programmalı moduller arqalı jaratıladı. MAX 2017 ornatiw komplektine úsh túrli sisteması kiredi:



- Bones (súyekler)- skelet súyegin eske saladı hám tiri janzatlardı animacion modelin jaratıwda paydalanıladı, yamasa mexanikalıq instrumentlerin jasaw. Bul obektler vizuallaspaydı hám proyeksiya aynasında kórinbeydi;

- Ring Array (xorovod)- berilgen obektler sanınan toplam jaratıwda paydalanıladı, kórsetilgen radiustı sheńber boyınsha tártip penen jaylasıwı hám sol obektlerdi animaciya háreketin orınlaw bolıp tabıladı.

- Sunlight (quyash nurı) –parallel jarıqlıq nurı deregin jaratıwğa ruxsat beredi, quyash jarqırawın immitaciyalaydı, geometriyalıq jaǵdaydı esapqa alıp quyash háreketin tikleydi, modellewshi saxnanı sutka hám máwsimine qarap jaylastıradı.

### Autodesk 3D Maya interfeys elementleri.

Tiykargı menyuda Windows operacion buyırıq toparları hám olardıń tańlaw baǵdarları standartları isletiledi. Tómenirekte hár bir menyu bóliminiń qısqasha analizi berilgen, buyırıq menyusınıń isletiliwi kerekli dárejede isletiledi.

File menyusı. File (fayl) buyırıq menyusi ashıwğa ruxsat beredi, saqlaw, importlaw, úsh ólshemli saxna faylın eksportlaw, súwretler faylın hám hár bir formattaǵı animaciyalardı kóriw. Bunnan tısqari, MAX 2017 konfiguraciyalaw hám dúzetiwge ruxsat beredi. Menyudiń tómengi bóliminde ashılıp atırǵan aqırǵı fayllardıń atlar dizimin jámlegen, olardıń sanı Recent Files in File Menu shechigine beriledi. (Menyu atı faylda sanı) óz úlesin qosadı Files (Fayllar) Preference Settings dialog aynası (parametrler nastroykası) Customizes Preferences (nastroyka-parametrler) komandası arqalı shıǵarıladı hám ol 9 ǵa teń.

Edit menyusı operaciyanı qaytarıw hám qaldırıw komandalarına ruxsat beredi, ajıratıw, nusqa kóshiriw, óshiriw hám obektler qásiyetlerin dúzetiw, saxna jaǵdayın tikle hám registraciyalaw múmkin.

-Undo (Biykar etiw)-aqırǵı orınlangan jumıs yamasa operaciyanı háreketin biykar etedi, onıń súwreti menyuda kórinedi. Undo (biykar etiw) úskeneler paneline yamasa Ctrl+z klavishalar kombinaciyasın basıw arqalı ámelge asırıw múmkin. Bul buyırıq tek aqırında qalǵan 20 ámeldi yadta saqlap qalıwı múmkin, biraq dizim kólemi Undo Cevec (biykar etiw dárejesi) esaplaǵısh járdeminde asıwı múmkin.

Preference Settings (parametrlerdi sazlawları) dialog aynası, menyu komandası arqalı shaqırıladı Customiz Preferences (dúzetiw) parametrler;

-Redo (tákırlaw) –Undo (biykar etiw) komandasın aqırǵı háreketin biykar etedi, yaǵnıy aqırǵı biykar etilgen komandanı tákırlaydı. Sonday maqsette Redo (tákırlaw) úskeneleri paneli hám xızmet etedi. Deleted (óshiriw) – belgilengen obektti óshiredi yamasa saxnanı belgilengen obektler jıyındısı. Óshiriw ushın jáne Del knopkasın basıw múmkin.

**Tools** menyusı. Tools (servis) menyusı hár qıylı baslanǵısh obektlerdi isletiwge ruxsat beredi, komandalı politralardı shaqırıwshın támiyinleydi hám Material Editor (materiallar redaktoru) hám Material/Map Browser (materiallardı kóriw kart tekstura).

**Group** menyusı. Group (topar) menyusı atlı, toparlı obektlerdi buzıwǵa hám redaktorlawǵa jaratıwǵa ruxsat beredi.

**Views** menyusı. Views (proekciyalar) menyusı Max 2017 obektleriniń barlıq aspektler menen kórsetiwge hám basqarıwǵa ruxsat beredi, kórsetiwdi esapqa alıp, járdemshi kordinatalı setkaların teńlestiriwi hám aktivlestiriw támiyinleydi hám basqarıw komandasın saxnada kórsetiliwin támiyinleydi proekciyalar aynasında ámelge asırıladı.

**Rendering** menyusı. Rendering (Vizuallastırıw) menyusı saxna vizuallastırıw komandalarına ruxsat beredi, animaciya eskizlerin kóriw hám jaratıw dialogı ayna Video Post (videomontaj) shaqırıwǵa ruxsat beredi hám parametrlerin dúzetedı.

Track View hám Schematic View menyusı. Track View (trekler kóriw) menyusı dialog oynasın manipulyaciyalaw komandasın óz ishine aladı. Track View (treklerdi kóriw) Schematic View (strukturasın kóriw) menyusı quramı boyınsha analogiyalıq-manipulyaciyalaw aynası dialogı komandası Schematic View strukturasın kóriw ushın xızmet etedi.

**Customize** menyusi. Customize (Dúzetiw) menyusı dúzetiw komandasın óz ishine aladı, Max 2017 zagruzka kórinisi interfeysi hám saqlaw, bazalı parametrlerdi dúzetiw,

kordinatalar setkasın hám baylanıslılıgın ólshew birligi konfiguraciyalaw jolları komandaları papkalar, fayllar menen proekciya aynası bolıp tabıladı.

**Max Script** menyusı. Max Script menyusı komandaların óz ishine aladı, jaratıw ushın mólsherlengen makroslar isletiliwi hám otkladkası programması, Max 2017 tilinde qurılğan hám dúzilgen Max Script hám programmalar dı avtomat tárizde boladı hám animaciya úsh ólshemli saxnasın, Max 2017 interfeysi elementlerin rawajlanıwı ushın xızmet etedi.

**Help** menyusı. Help (járdem) menyusı Max járdemshi sistemasına kiriwge ruxsat beredi, járdemshi maǵlıwmatlardı obnovleniye etiwdi támiyinleydi. Internet tolqını boyınsha, bunnan tısqarı kesiwshi programmanıń tolqını informaciya óz ishine aladı.

Komandalı paneller. Autodesk 3D Maya da 6 dana komandalı panellerge iye koreshkalar menen jaylasqan: Create (jaratıw), Modify (ózgertiw), Hierarchy (Ierarxiya), Motion (háreket), Display (Ekran) hám Utilities (servis). Kerekli panel komandasın tańlaw ushın sáykes keliwshi koreshkanı basıw kerek.

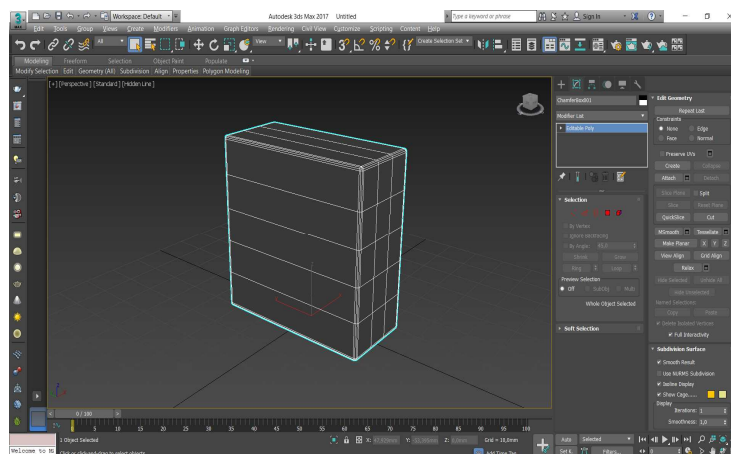
## 1.2. 3D Max Studio programmasında sistemalı blok modelin jaratıw

3ds Max 2017 programmasında Sistemalı bloktıń 3d modelin jaratıw izbe-izligi, ob'ekt teksturasın jaratıw hám teksturanı ob'ektke jaylastırıw kórsetilip ótilgen.

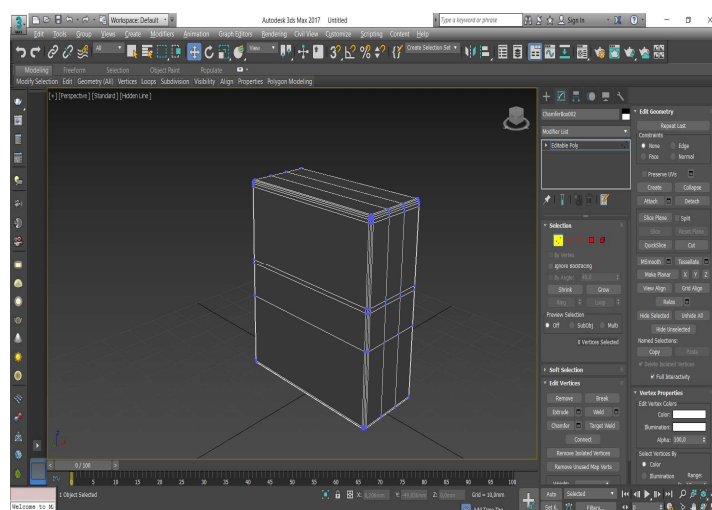
Tekstura jaylastırılátın ob'ektti jaratıw. Birinshiden teksturanı jaylastırátın ob'ektti jaratıp alamız. Ob'ekt sıpatında kompyuter sistemalı blokı saylandı. Jumısımızdı ChamferBox ob'ektin jaratıw(Create->Extended Primitives->ChamferBox) arqalı baslaymız:



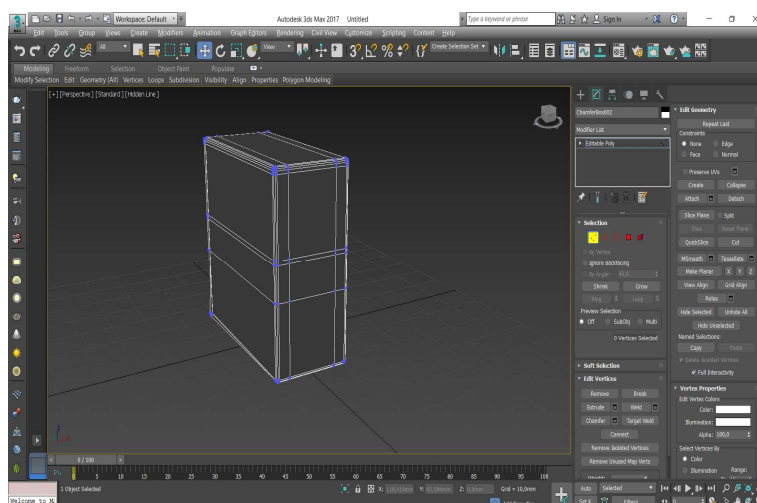
Kórsetilgen parametrlar menen ChamferBox jaratamız.



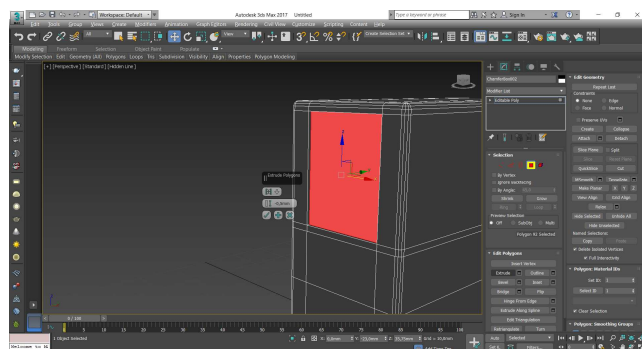
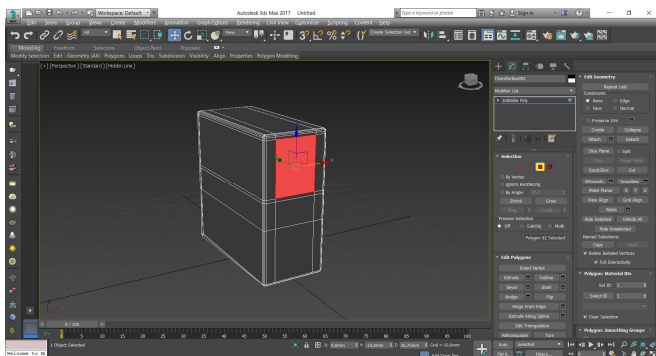
Jaratılğan ob'ektimizdi Redaktorlanıwshı poligonğa(Editable Poly) ózgartiremiz hámde eń joqardağı noqatların belgilep, z koordinatasın(yaǵnıy biyikligin) 44.5 mm ge ózgertemiz. Keyingi noqatlardıda usı tártipte, yaǵnıy 27.0, 26.3 hám 17.8 mm ge ózgertemiz. Nátiyjede ob'ektimiz tómendegishe kóriniske iye boladı:



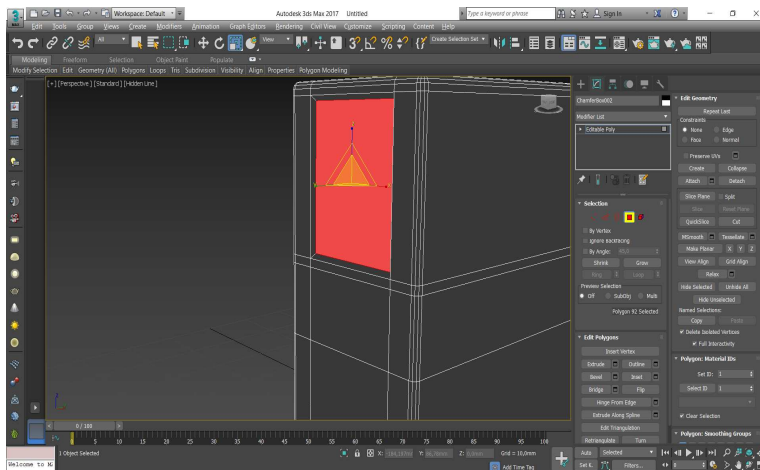
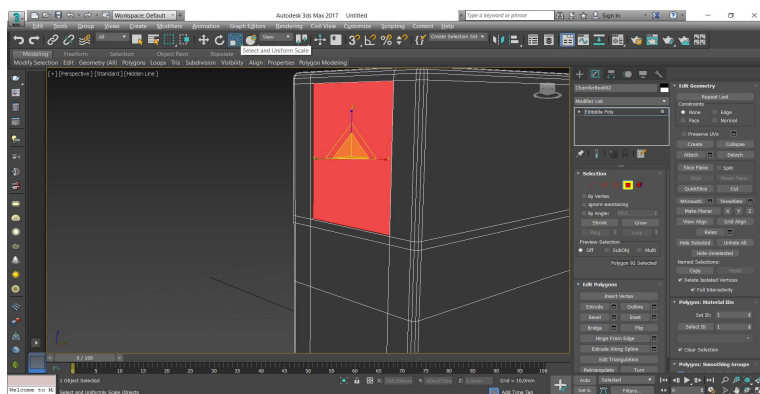
Ob'ekt ortasındagı vertikal qabırǵalardı ob'ekt shetinen 3 mm aralıqqa jaylastıramız:



Ob'ektimiz aldınğı tárepindegi joqarǵı ortadaǵı poligondı belgilep, Extrude instrumenti járdeminde -0.5 mm ge ob'ektin ish betine kiritilsin.

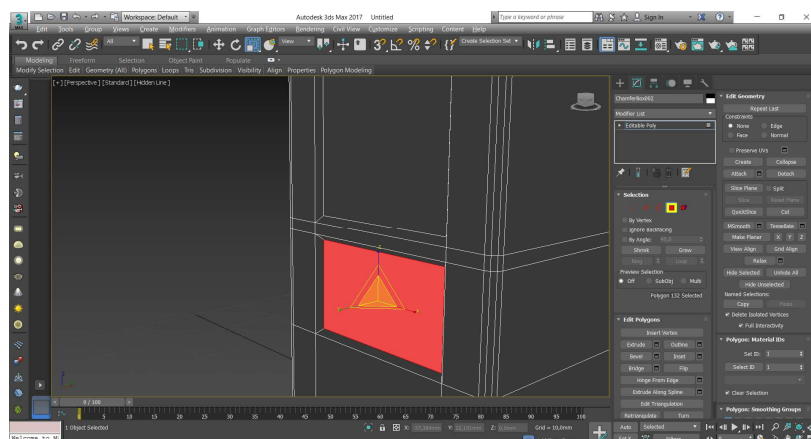
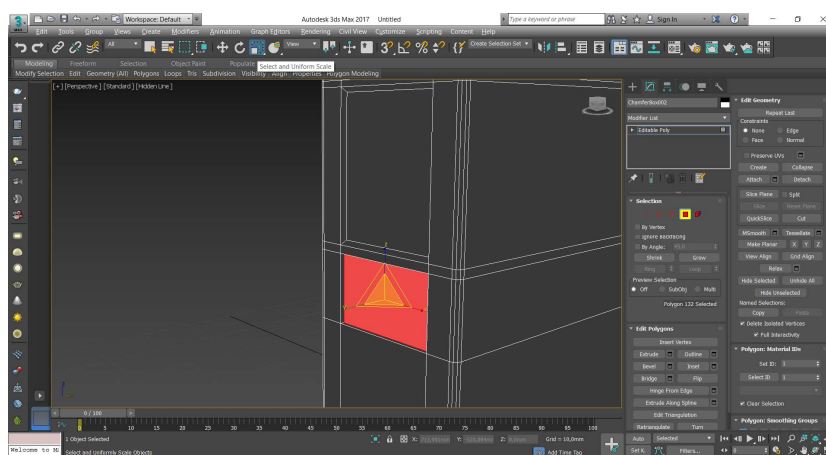
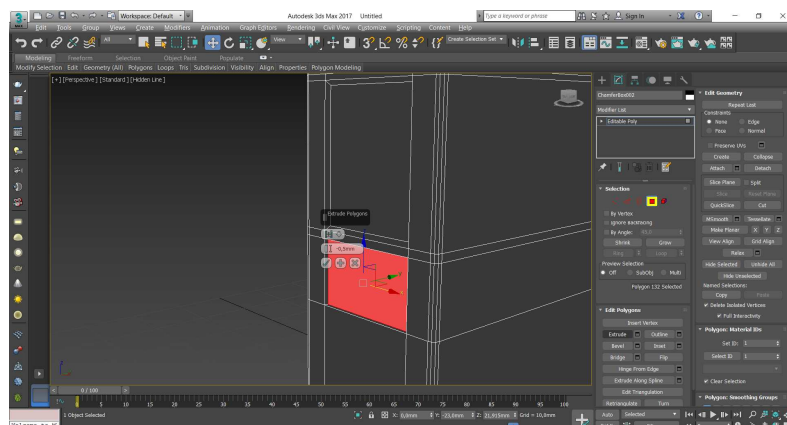


Ob'ekt ish tárepine shuqurlıq payda etiw ushın Select and Uniform Scale instrumenti járdeminde polygon ólshemin ózgerťin(bul jaǵdayda kishireytemiz).

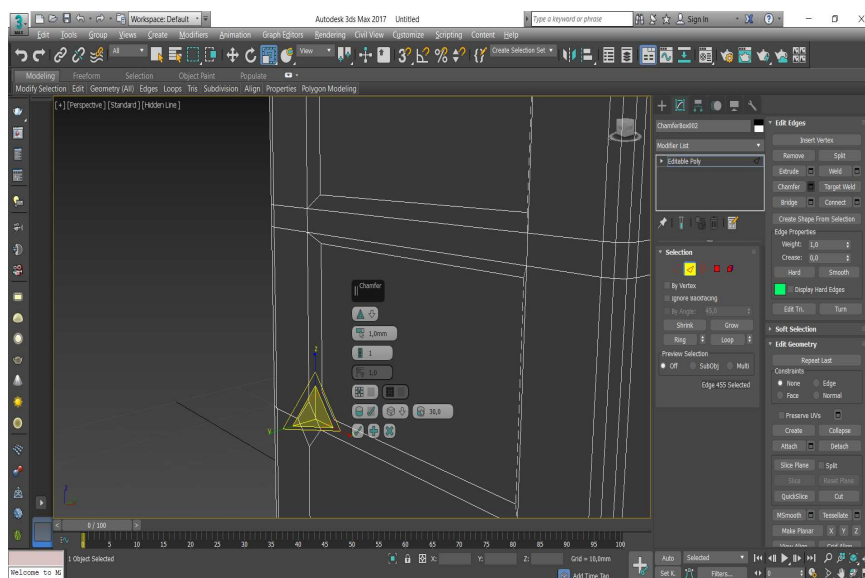
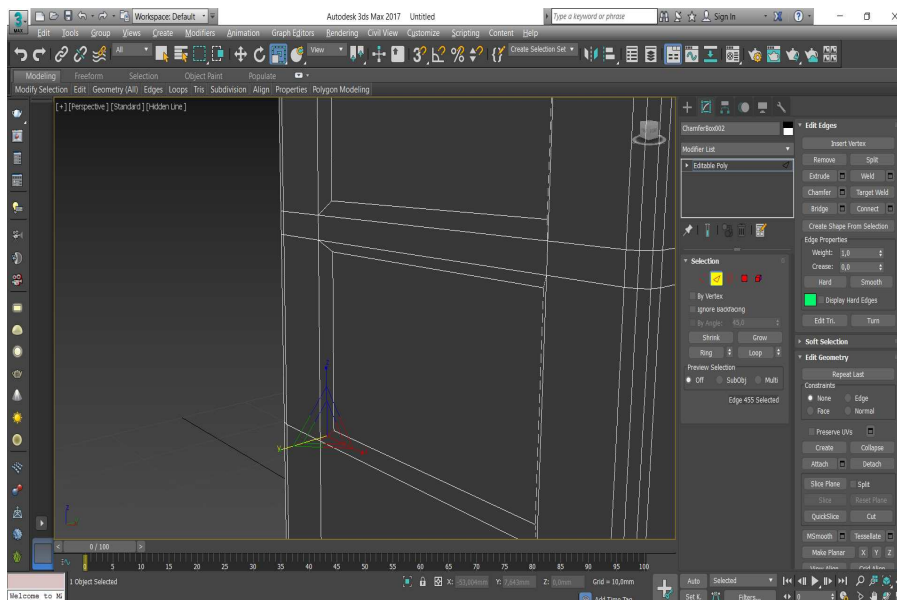




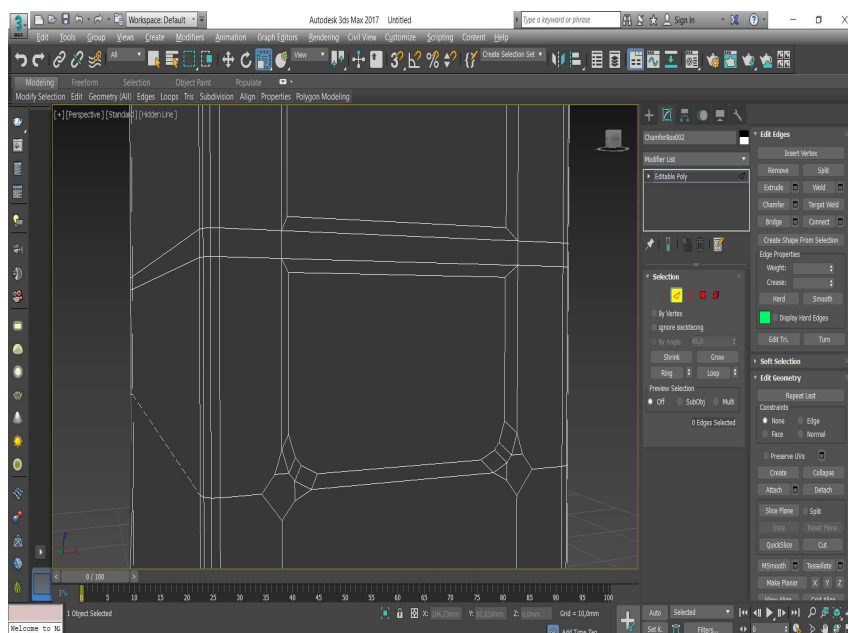
Tap sonday operaciyanı ortadağı poligongada orınlaw kerek.



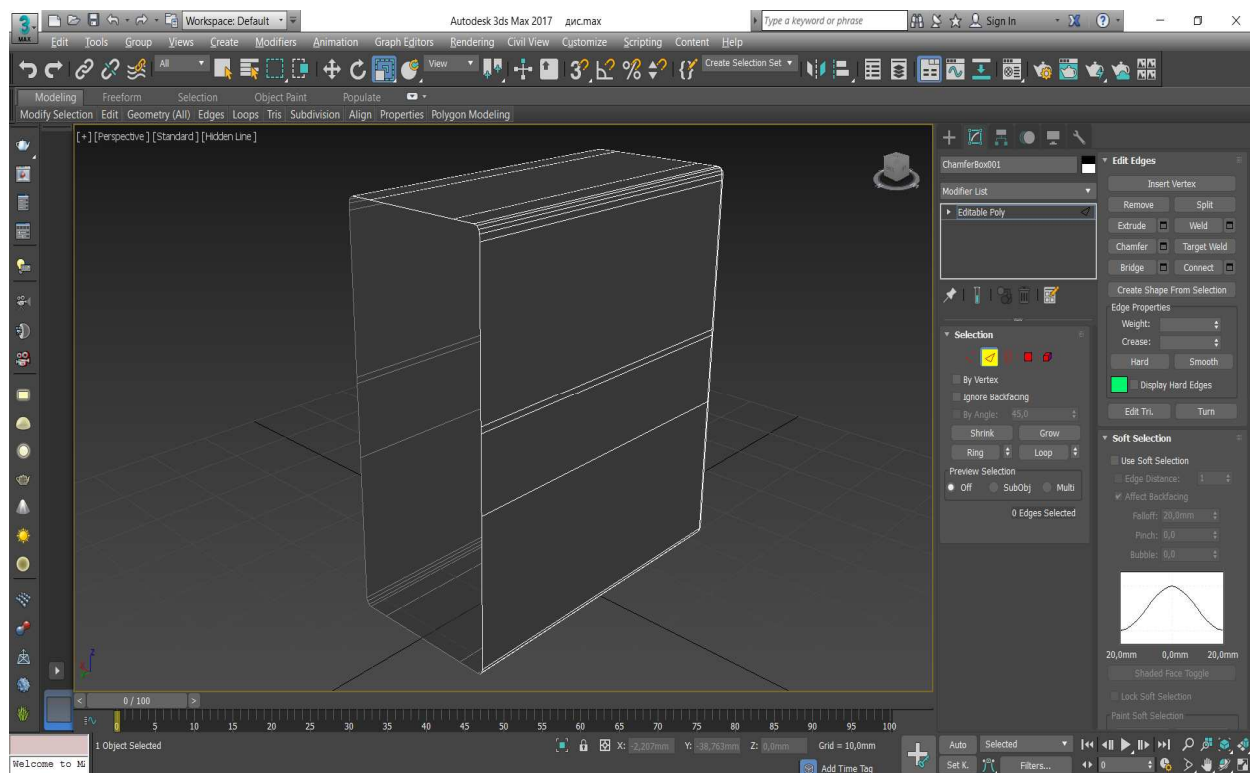
Ortadağı poligondağı jaratılğan shuqurlıqtıń tómenge tárepın aylanba kórinisine alıp keliw kerek. Bul ushın tómenge muyeshtegi qabırǵanı belgileń, hám usı qabırǵa ústinde 2 márte Chamfer buyruǵın orınlaw kerek.(birinshisinde Champer ólshemi 1 mm, ekinshisinde 0.5 mm).



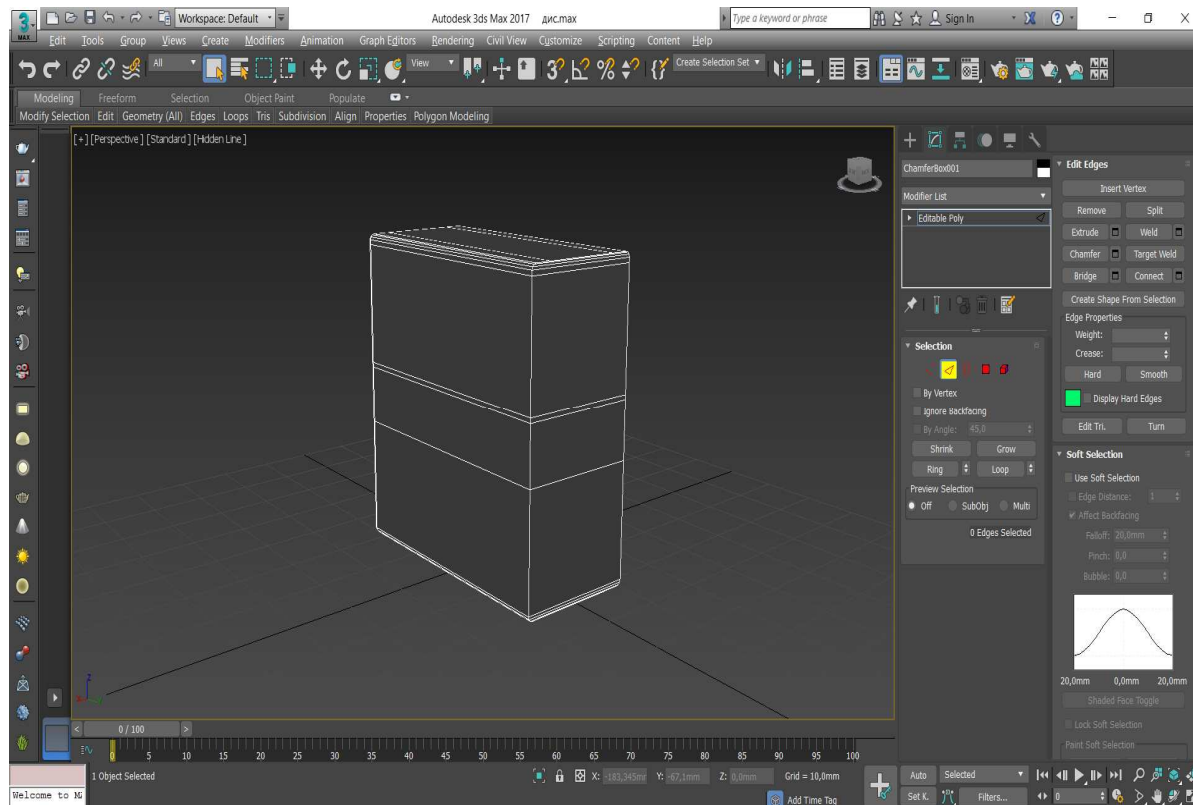
Tap usı háreketlerdi 2-shi muyeshtegi qabırǵa ushında orınlanadı:



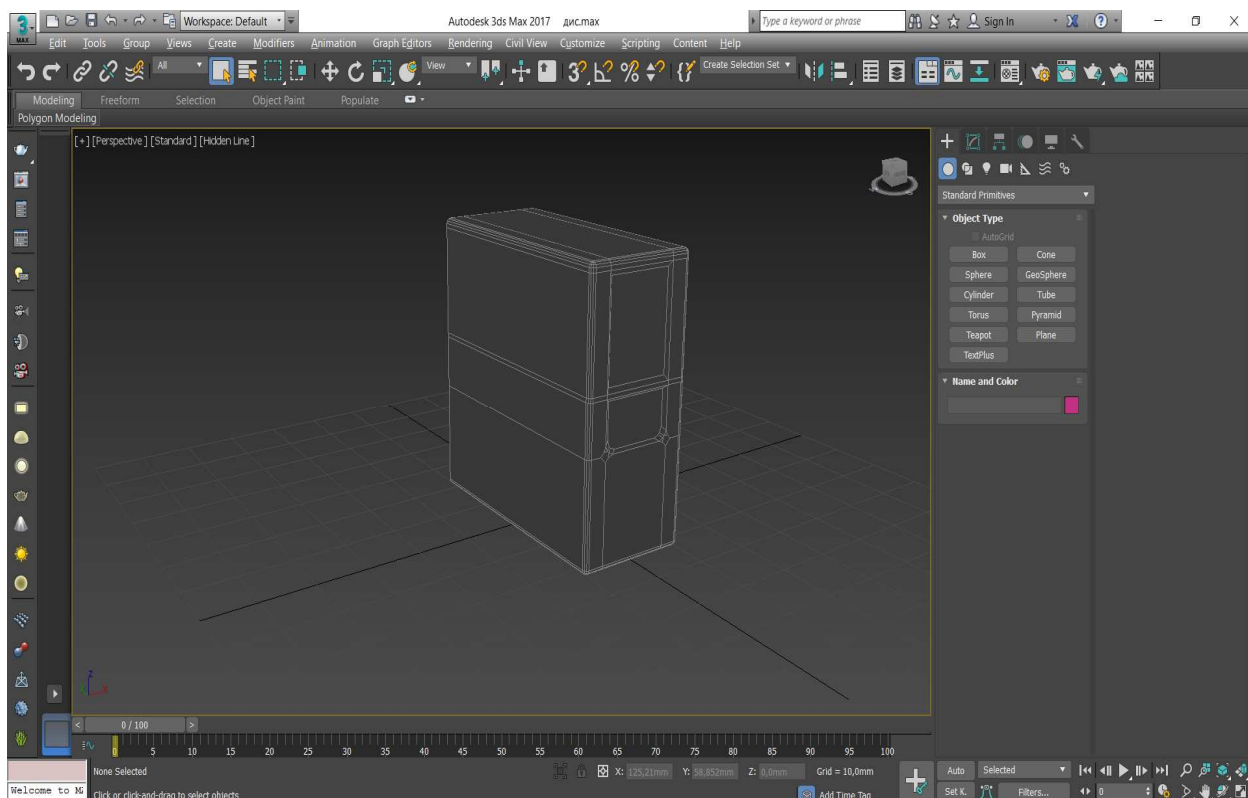
Endi ob'ektimiz artqı tárepindeki aylanbalı táreplerin óshiremiz. Bunıń ushın aylanbalı táreplerin belgilep delete klavishası arqalı óshiriwge boladı.



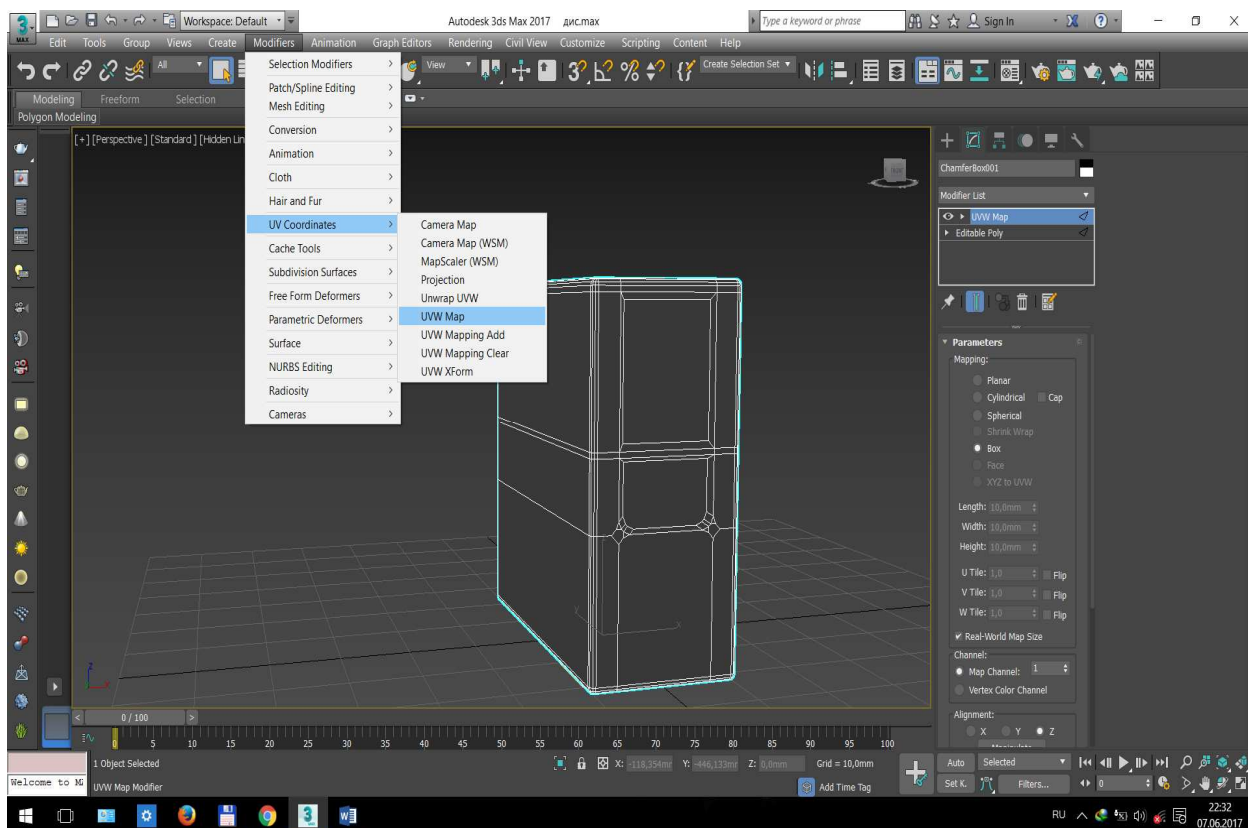
Óshirgenimizden keyin, qabırğaların Connect instrumenti járdeminde baylanıstırıp shıǵamız:



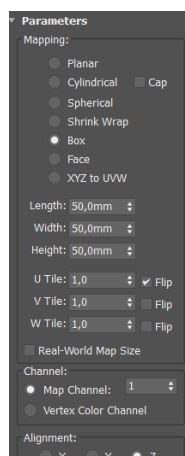
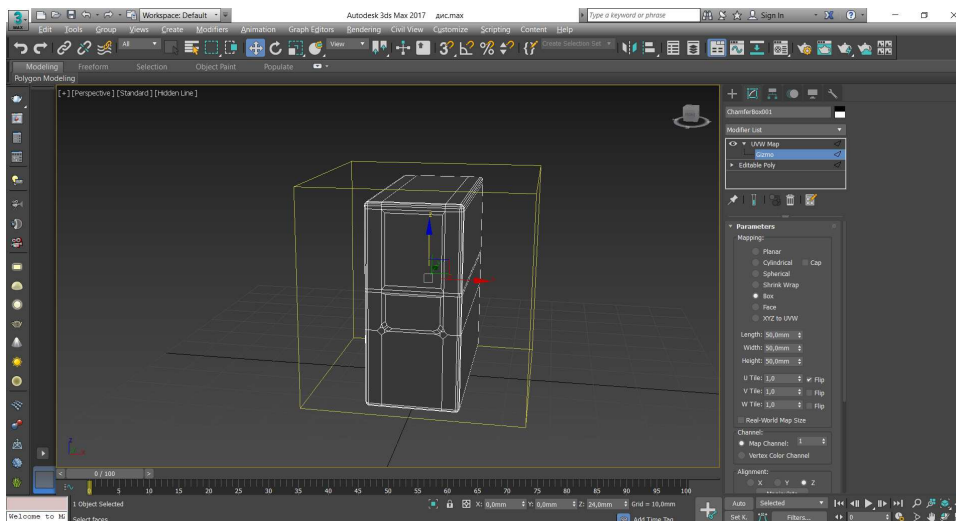
Nátiyjede teksturasız sistemalı blok jarattıq:



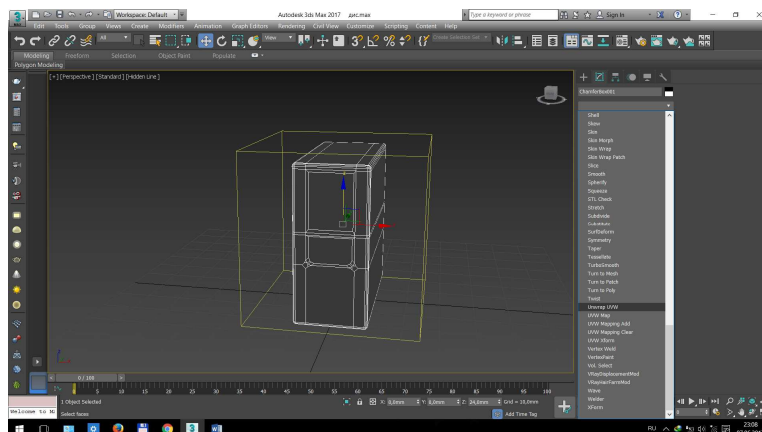
Jayılmalar(развертки) jaratıw – bul ob’ekt ushlarına tekstura koordinataların berip shıǵıw esaplanadı. Bunıń ushın ob’ektke UVW Map modifikaciyasın jaylastıramız. Keyin kerek bolǵan koordinatalardı keltirip shıǵaramız:



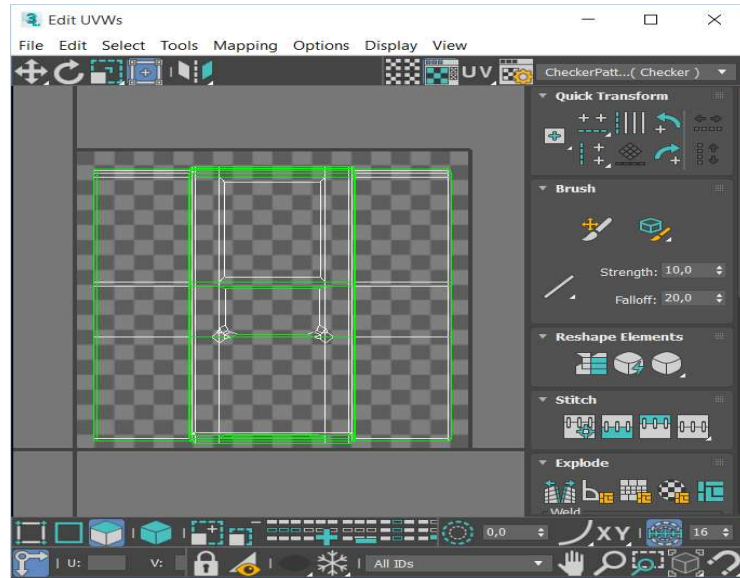
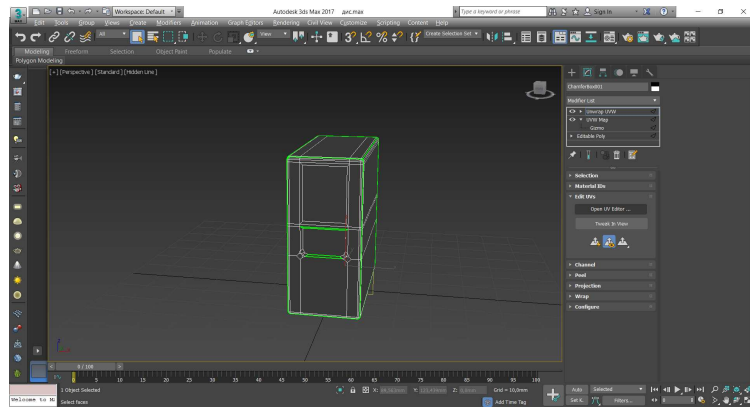
Eger islengen háreketlerden keyin Gizmo(ob’ekt átirapındaǵı sargısh sızıqlar) ob’ektti tolıq ishine alıp turmaǵan bolsa, Gizmo panelinen Gizmo ólshemlerin ózgertiw kerek:



Endi kerekli bolǵan tekstura koordinatalarına iye boldıq, tek ǵana olardı kerekli jaǵdayǵa alıp keliw kerek. Bunıń ushın ob’ektke Unwrap UVW modifikatorın beremiz. Egerde Open UV Editor instrumentin tańlasaq:





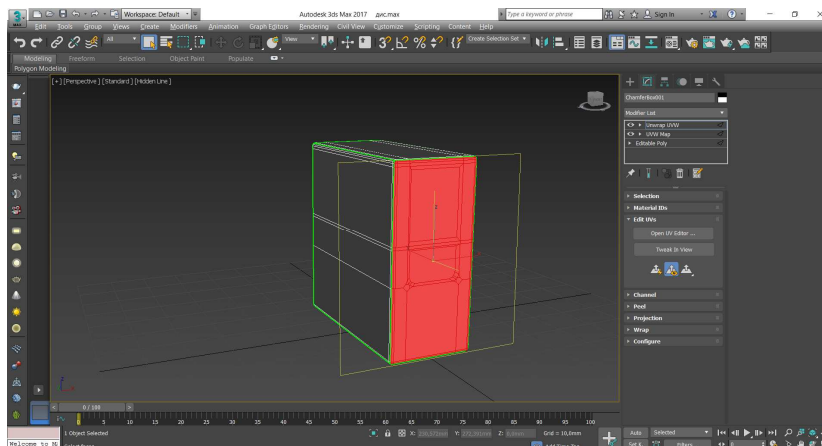


Ol jagdayda tekstura koordinataların redaktorlaw mumkin bolğan ayna ashıladı. UVW Map tı paydalanğanımız sebepli, aynamızda jasıl reñdegi sızıqlar bar, bul sızıqlar ob’ektimiz poligonları(tärepleri) esaplanadı. Options->Filter Selected Faces buyırǵı járdeminde belgilenbegen poligonlardı aynamızda kórsetpey qoyıwımız mumkin.

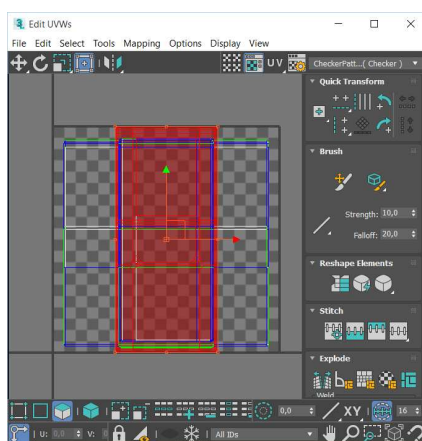
Endi Unwrap UVW modifikatorı poligonlardı redaktorlaw rejimine ótiń.



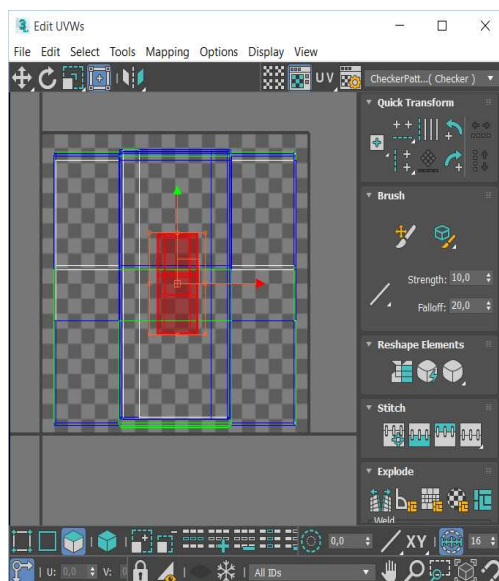
Endi ob'ektnıń aldınǵı tárepindegi poligonlardı belgileń:



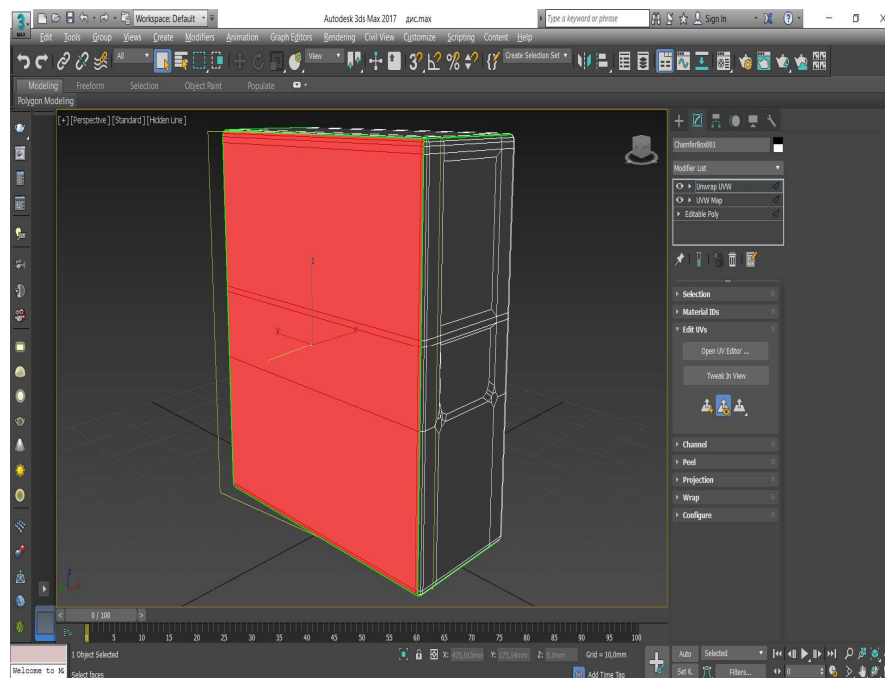
Tekstura koordinataların redaktorlaw aynasına ótiń(Open UV Editor knopkası), bul jerde:



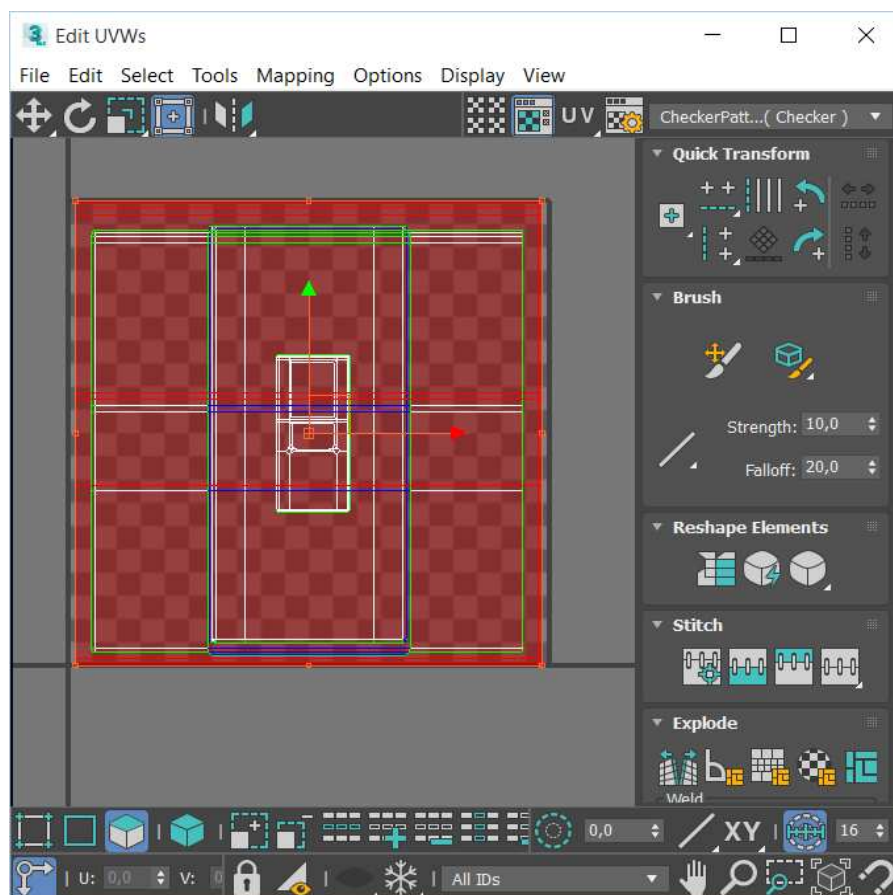
Ólshemin hám ornın ózgertiw instrumentlerin paydalanǵan halda belgilengen poligonlardı tómendegishe kóriniske keltiriń:



Shep tárepindegi poligonlardı belgileñ:

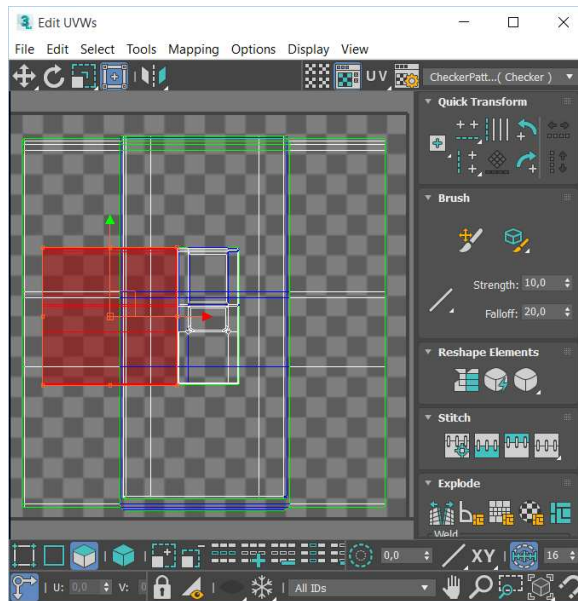


Redaktorlaw aynasında tómendegishe kóriniske iye boladı:

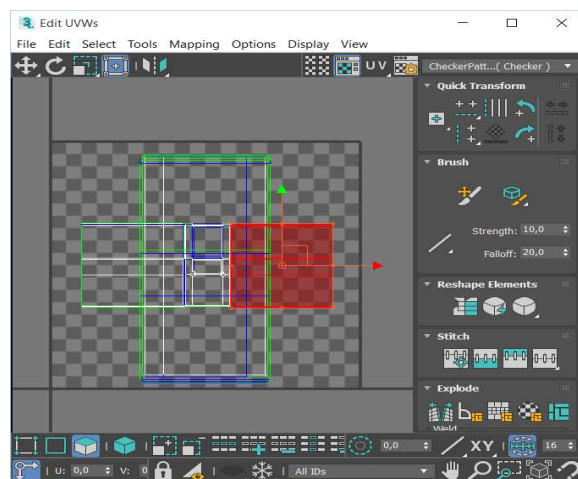




Tómendegishe kóriniske keltiriń:

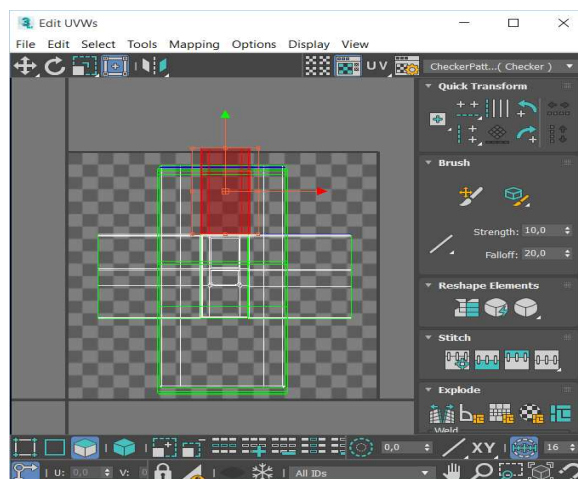


Tap sonday oń tárepindegi poligonlardıa tómendegishe kóriniske keltiriń:

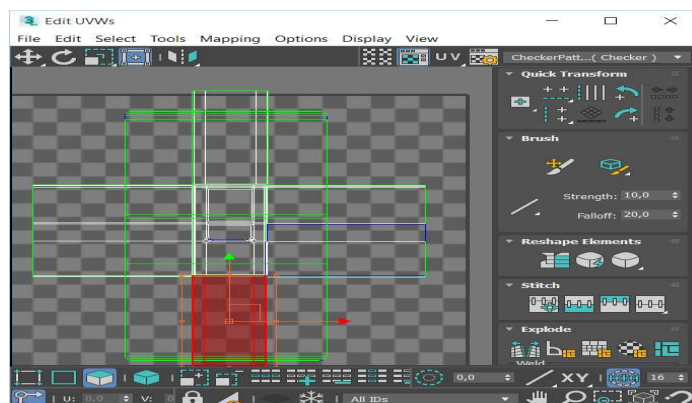


Usı jol járdeminde joqarǵı, tómengi hám arqa tárepinde tómendegishe kórinislerge keltiriń:

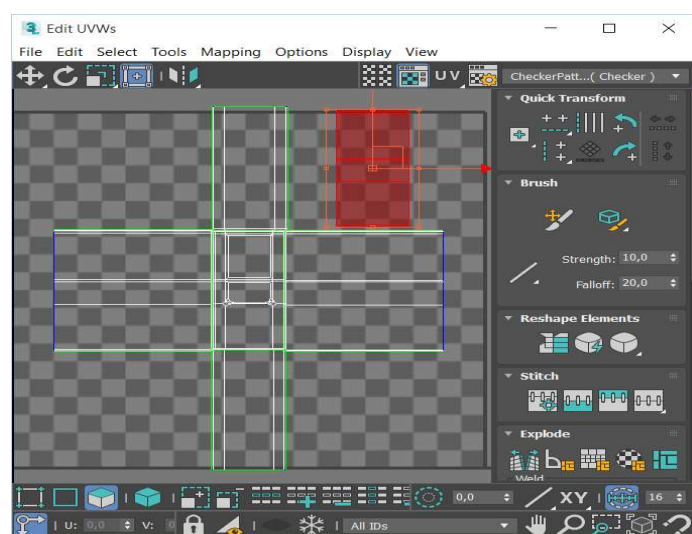
Joqarı tárepi:



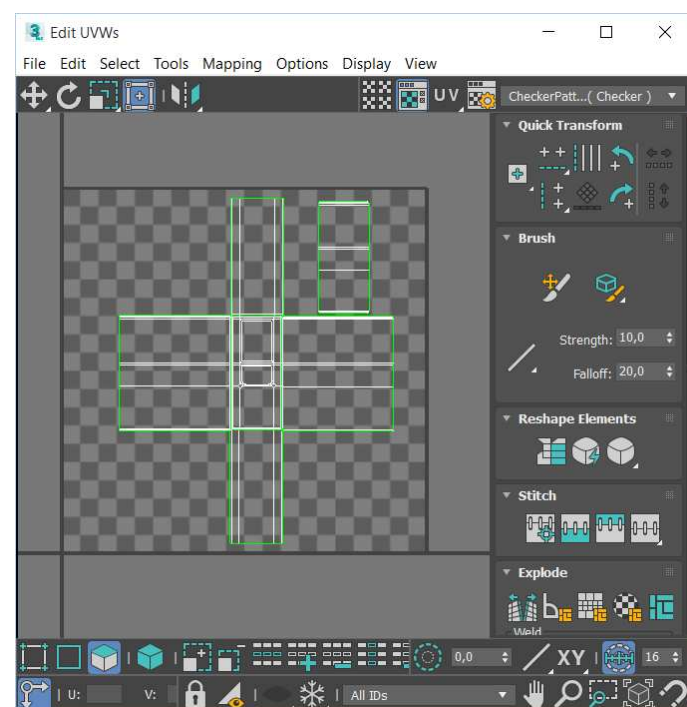
Tomengi tárepi:



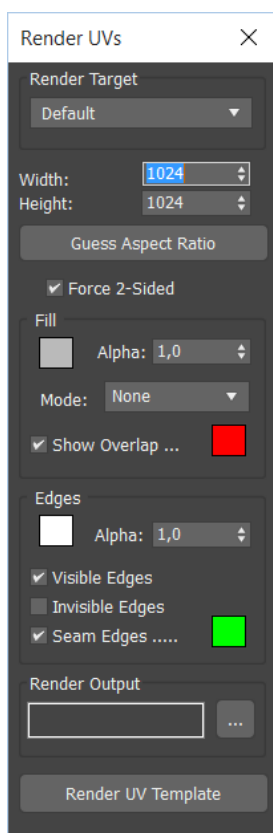
Arqa tarepi:



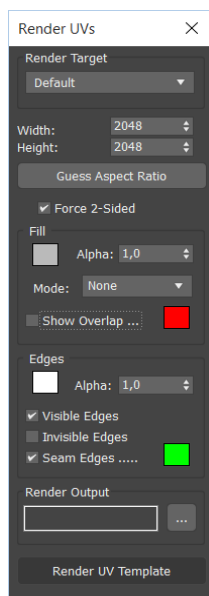
Son'inda tómendegishe kóriniske keledi:



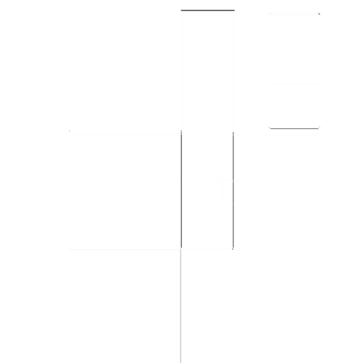
Usı tárizde ob'ektti jayılmısına alıp keldik. Endi usı jayılmının grafik kórinisin alıw kerek boladı. Bunıń ushın Tools-> Render UVW Template instrumentinen paydalanamız. Bul instrumenttiń aynası tómendegishe:



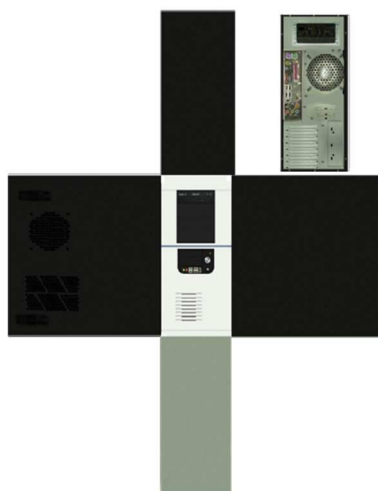
Kvadratlı tekstura payda etiw ushın enin(width) hám biyikligin(height) bir ólshemde, yaǵnıy 2ge eseli sanlarda(256,512,1024,2048 hám t.b.) qoyılǵanı maqsetke muwapıq:



Render UV Template knopkasın basıp, tómendegishe kórinisti alamız.



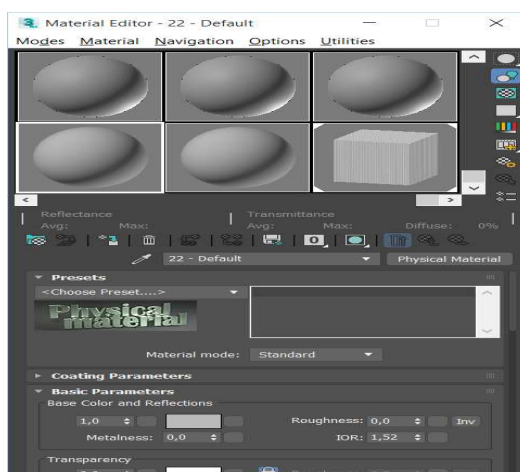
Save Image knopkası járdeminde suwretimizdi ózimizge qolaylı bir formatta saqlaymız. Grafik redaktorı járdeminde ózimizge unáğan tárizde bezeymiz:

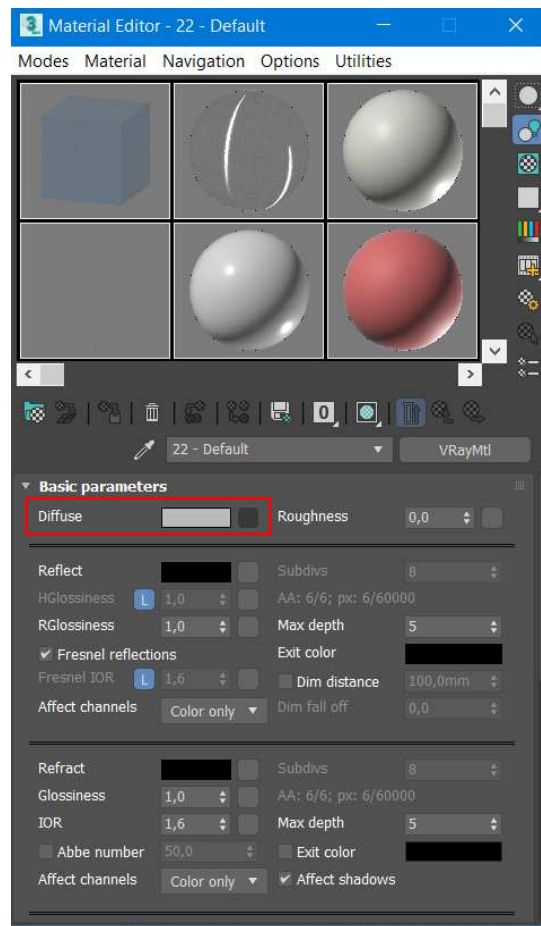


Usı menen teksturamızdı jarattıq. Endi bul tekstura menen ob’ektimizdi qaplaymız.

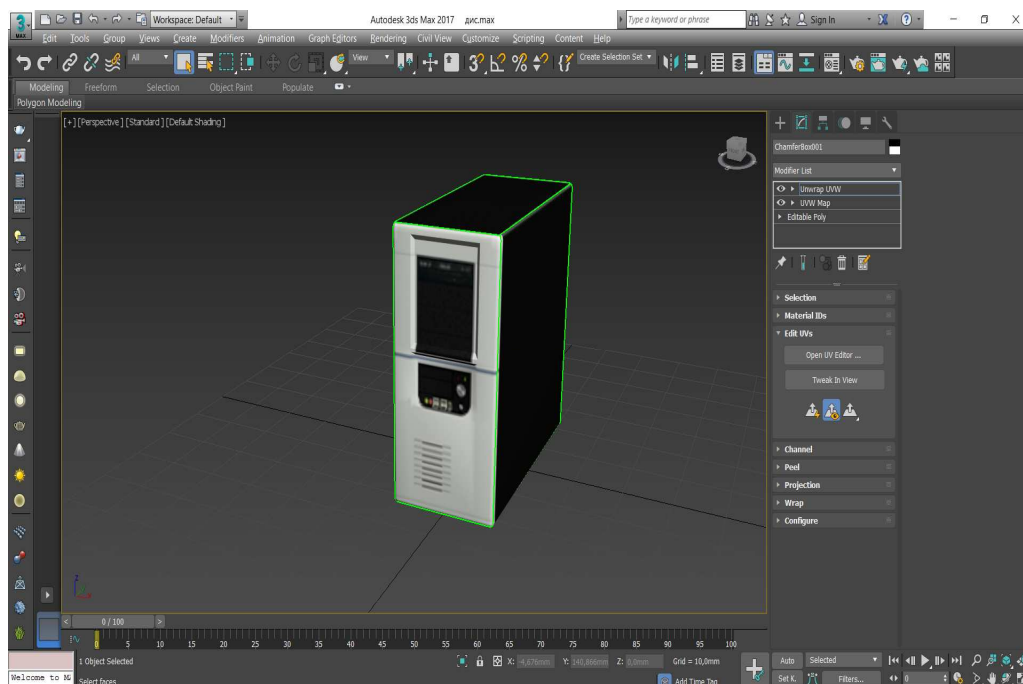
Ob’ektti tekstura menen qaplaw

Materiallardı redaktorlaw aynasın ashıń(Klavish M), Diffuse qasındaǵı knopkaǵa basıladı:





Júklengen aynada Bitmaptı tańlap, saqlanǵan súwretimizdi ashıń hám Assign Material to Selection klavishasını basıp, ob’ektke jaratılǵan materialdı qollań. Nátiyjede ob’ektimiz tómendegishe kóriniske iye bolıwı kerek:



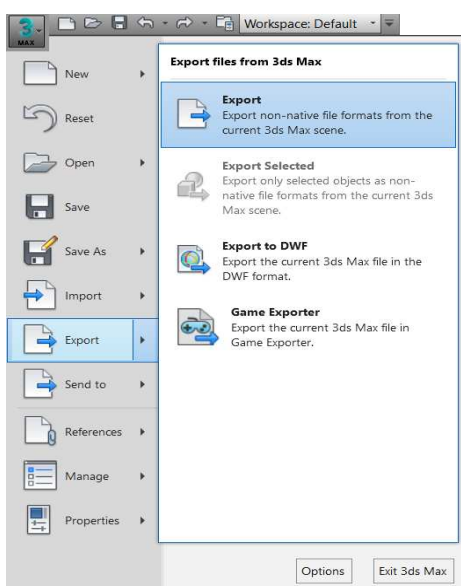
### 1.3. 3D Max Studio programmasında sistemalı blok modelin eksport etiw

GLUT bibliotekası kóplegen 3D-ob'ektlerdi jaratıwǵa múmkinshilik beredi. Bunda bizler hár qıylı usıllar menen jaratıwımızǵa boladı. Biraq quramalı grafikalıq saxnalardı jaratıw ushın 3DS Max úsh ólshemli ob'ektlerdi modellestiriw ushın arnalǵan paket programması járdem beredi. 3DS Max programmasınan eksport etkende, bizler geometriyalıq ólshemlerin hám keńisliktegi kordinataların hám jaqtılıq taratıwshı ornın alamız.

Geometriyalıq maǵlıwmatlardıń saqlanıwı, saxnanıń barlıq elementleri N sıızıqtan turatın bildiredi. Bul degeni úsh ólshemli ob'ektiń keńisliktegi kordinatalar jıynaǵına aytıladı. Bizler úsh ólshemli ob'ektti júklegende misal etip ASE formatın kóremiz. Bul formatta 3DS Max programmasında jaratılǵan ob'ektti jeńil saqlawǵa boladı.

Endi 3DS Max programmasında jaratılǵan Sistemalı blok ob'ektin eksport qılamız:

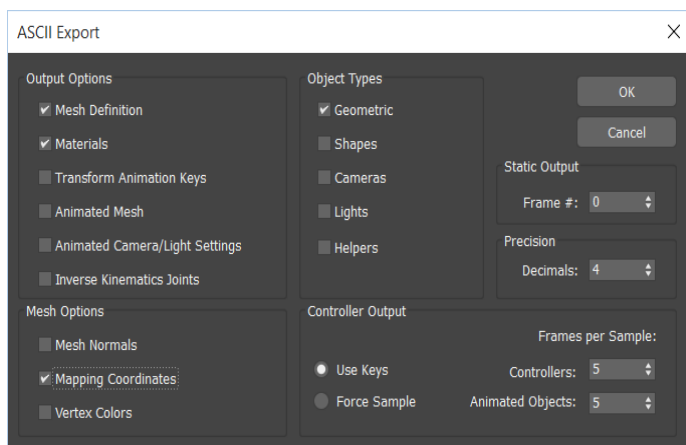
Bunıń ushın File menyusınan Export ti tańlaymız:



Payda bolǵan aynadan File name: Sisblok Save as Type: ASCII Scene Export (\*.ASE) ni tańlaymız hám Save túymesin basamız:



Keyin export etiw qásiyetlerin tómendegishe ózgertemiz:



## II-BAP C# OB'EKTKE BAĞDARLANGÁN PROGRAMMALASTÍRÍW TILINDE WINDOWSFORM APPLICATION MENEN ISLESIW

### 2.1. C# ob'ektke baǵdarlangán programmalastırıw tili sintaksisi

Hár bir hár ózgeriwshi olar ústinde orinlaniwshi ámeller aniqlawshi túrlerge iye boladi. Máselen, `int i`; táriyiplewi ózgeriwshi `int` túrine tiyisli, yaǵniyi pútín ózgeriwshi dep aniqlaydi. Táriyiplew— dástúrge at kiritiwshi buyriq. Táriyiplew ózgeriwshiniń túrin aniqlaydi. Túr ati hám ańlatpalardan tuwri paydalaniwdi aniqlaydi. Pútín túr ushin tómendegi ámeller aniqlangan: `+`, `-`, `*` va `/`.

Tiykargı túrler Ózi-ózinen apparat táminatına juwap beretuǵın tiykargı túrler tómendegishe:

`char`; `short`; `int`; `long`; `float`; `double`.

Birinshi tórt túr pútín ólshemliklerdi, aqirǵı ekewi soziwshi noqatli, yaǵniy bólshek sanlardı kórsetiw ushin isletiledi. `char` túrindegi ózgeriwshi usı kompyuterde belgilerdi (ádette bayt) saqlaw ólshewine iye, inttúrindegi ózgeriwshi usı kompyuterde pútín arifmetikaǵa say ólshewgeiye (ádette sóz). Túrler menenkórsetilgan pútín sanlar diapazonı onıń ólshewinebaylanisli boladı (onı `sizeof` buyrıǵı járdemindeesaplaw múmkin).

C++ te ólshewler `char` túrindegi ólshemliklar ólshewi birliginde ólshenedi. Tiykargı túrler ortasındaǵı qanasıqlardı tómendegishejaziw múmkin:

$1 = \text{sizeof(char)} \leq \text{sizeof(short)} \leq \text{sizeof(int)} \leq \text{sizeof(long)} = \text{sizeof(float)} \leq \text{sizeof(double)}$ .

Uliwma, tiykargı túrler haqqında jáne basqa nárselerdi qiyal qiliw mánissiz. Tiykarinnan, kórsetkishlerdi saqlaw ushin pútin túr jeterli, degen juwmaq barliq kompyuterler ushin tuwri emes. Tiykargı túrlerge const sózin qosip táriyiplew múmkin. Bul baslanǵısh túrgeusi túrdıń ózini beredi, tek bul jaǵdayda const túrindegi ózgeriwshilerinıń mánisleri inicializaciyanan soń ózgeriwi múmkin emes. Misali:

```
const float pi = 3.14;
```

```
const char plus = '+';
```

Birlik qostirnaqqa alınǵan belgiler belgisi ózgermesler dep esaplanadi. Soǵan itibar beriw lazim, bul usilda táriyiplengen ózgermesler yadda oriniyelemeydi. Onıńmánisi talap qilingan orindaerkin isletiledi. Ózgermesler inicializaciya waqtında táriyipleniwi shárt. Ózgeriwshiler ushin inicializaciya shárt emes, biraq álbetteusini qilinadi. Lokal ózgeriwshilerdi inicializaciyasız kiritiw tiykarları júdá kóp.

Bul túrlerdiń tańlawsız kombinaciyasına tómendegi arifmetik ámeller qollaniliwi múmkin:

+ (plyus, unar hám binar); - (minus, unar hám binar);

\* (kóbeytiriw); / (bóliw).

Hámde salıstiriw ámelleri:

== (teń); != (teń emes);

< (kishi); > (úlken);

<= (kishi yáki teń); >= (úlken yáki teń).

Eger operandlar qoyılǵan shártti qanaatlantırsa, ol halda salıstiriw ámelleri nátiyjede 1 mánisti beredi, kerı halda bolsa 0 mánisini beredi.

Pútinge bóliw ámeli pútin nátiyjeni beredi:  $7/2 = 3$ .



Pútin ólshemlikler ústinde % - qaldıqtı esaplaw ámeli orınlanadı:  $7\%2 = 1$

Ózlestiriwde hám arifmetik ámellerde C++ olardı guruppalam ushin tiykargı túrler ortasında barlıq mánisli almastiriwları orınlaydı:

```
double d = 1;
```

```
int i = 1;
```

```
d = d + i;
```

```
i = d + i;
```

Qatarlı túrler

C++ te belgilerdiń ayırım bir izbe-izligi (massivlar) dan ibarat tekst qatarların yadda saqlaw ushin arnawlı `AnsiString` maǵlıwmatlar túri qollanılıadı. «Stroka» - «Qatar» túrindegi ózgeriwshiler barlıq basqa ózgeriwshiler sıyaqlı daǵaza hám inicializaciya etiledi.

Kompilyatorǵa nábettegi belgiler izbe-izligi jańa ózgeriwshinińati emes, bálki qatar ekenligini bildiriw ushin qatarlar birlik qostırnaq ishine alınıadı.

Misal:

```
AnsiString st = 'matn qatori';
```

Qatar túrindegi ózgeriwshiler ústinde basqa qatar ózgeriwshiler menen qosiw ámeli orınlaniwi múmkin. Bul ámel eki qatardı olardıń keliw tártibinde birlestiriw dep túsiniledi.

Misal:

```
AnsiString s1 = 'dástúrlewshi';
```

```
AnsiString s2 = 'teksti';
```

```
AnsiString s = s1 + s2;
```

Nátíyjede s ózgeriwshi s1hám s2 ózgeriwshilerden payda bolǵan dástúrlewshi teksti degen mánisti qabıl qıladı.

### Qosımsha túrler

Borland C++ te pútin mánisli ózgeriwshilerdiń túrlerini qosımsha ajratıw imkani bar. Bul jaǵdayda ózgeriwshilerdiń barlıq túr atlari ómendegishe jazıladi - int X, buljerda X ózgeriwshiniń bitlardaǵı maydan ólshemi. X tómendegimánislerden birewi qabil qiliwi múmkin: 8, 16, 32 hám 64. Bul túrdegi ózgeriwshilerden paydalaniw standart túrde aniqlanǵan ózgeriwshilerden paydalaniwdan parq qilmaydi.

Tómendegi keste de bunday túrler menen islewaniq kórsetilgen.

Túr ati Ózgeriwshini táriyiplewge misal Ólshem

\_\_int8 \_\_int8 c = 128; 8 bit

\_\_int16 \_\_int16 s = 32767; 16 bit

\_\_int32 \_\_int32 i = 123456789; 32 bit

\_\_int64 \_\_int64 big = 12345654321; 64 bit

unsigned \_\_int64 unsigned \_\_int64 huge = 1234567887654321; 64 bit

Gilt sózler – bul sózler programma kodın jazıw ushın paydalanıladı.

C# taǵı gilt sózler			
Abstract	as	base	bool
Break	byte	case	catch
Char	checked	class	const
Continue	decimal	default	delegate
Do	double	else	enum
Event	explicit	extern	false
Finally	fixed	float	for

Foreach	goto	if	implicit
In	in	int	interface
Internal	Is	lock	long
Namespace	new	null	object
Operator	Out	out	override
Params	private	protected	public
Readonly	ref	return	Sbyte
Sealed	short	sizeof	stackalloc
Static	string	struct	Switch
This	throw	true	Try
Typeof	uint	ulong	unchecked
Unsafe	ushort	using	virtual
void	volatile	while	

### O'zgeriwshini ózi anıqlaw programması

```
string @out;
string identy;
string var {get; set;};
```

```
public void function()
{
    var s = ""; }

```

### Konstantlar

Const operatori menen anıqlanadı bul ózgeriwshi programma dawamında ózgertilmeydi. Mısalı:

```
class Foo
{
    const double PI = 3.1415;
}
```

```

    Foo()
    {
        const int countOfPlanets = 8;
    }
}

```

Kod blokları

{...} operatorları kodlar bloğının başlanıwı hám tamamlanıwın bildiredi, Máselen:

```

void Misal()
{
    int a;
    {
        int b;
        a = 0;
    }
    a = 1;
    b = 1;
}

```

Programma strukturası

---

C# ta jazılğan programmalar klasslardan hám onıń elementlerinen turadı. Klasslar hám basqa tipler programma kodı eń basında anıqlanadı.

```

namespace FooNamespace
{
    class Foo
    {
        class IncludedClass
        {
        }
    }
}

```

Operatorlar

---

Operatorlar kategoriyası	Operatorlar
--------------------------	-------------

Arifmetikalıq	<code>+ - * / %</code>
Logikalıq	<code>&amp;   ^ ! ~ &amp;&amp;   </code>
Конкатенация строк	<code>+</code>
Arttiriw kemeytiw	<code>++ --</code>
Aralastiriw	<code>&lt;&lt; &gt;&gt;</code>
Ten'lestiriw	<code>== != &lt; &gt; &lt;= &gt;=</code>
Ózlestiriw	<code>= += -= *= /= %= &amp;=  = ^= &lt;&lt;= &gt;&gt;=</code>
Elementlerge ruxsat	<code>.</code>
Indekslew	<code>[]</code>
Tiplerdi keltiriw	<code>()</code>
Shártli	<code>? :</code>
Ob'ektti jaratıw	<code>New</code>
Tip haqqında maǵlıwmat	<code>as is sizeof typeof</code>
Basılǵanın tekseriw	<code>checked unchecked</code>

Kórsetkish penen jumıs islew	* -> [] &
------------------------------	-----------

## 2.2. C# tilinde ob'ektke baǵdarlangan programmalastırıw tili operatorları

### Shártli operatorlar **If** hám **switch**.

Saylaw operatorı (shártli saylaw) if – bul operator shárt orınlansa yamasa orınlanbasa isleydi. Eger berilgen shárt ras bolsa onda then operatorınan keyingi ámeller orınlanadı. Al orınlanbasa else operatorınan keyingi ámeller orınlanadı.

```
if (resault == 7)
{
    System.Console.WriteLine('Siz durıs taptıńız');
}
else
{
    System.Console.WriteLine('Iltimas qaytaldan urınıp kóriń');
}
```

Tarmaqlanıwshı operator switch – bul operator kóplegen tarmaqlangán ámellerdi beriliwi múmkin, olardan birin saylap alıw mánis penen belgilenedi. Bul arnawlı parametrler berilgende júdá qolaylı boladı. C# ta ol tómendegishe jazıladı:

```
switch ( value )
{
    default:
    {
        System.Console.WriteLine('Bul variant ushin amel aniqlanbagań');
        break;
    }
    case 1 :
    {
        System.Console.WriteLine('1 cifrasi tanlangań');
        break;
    }
    case 2 :
        System.Console.WriteLine('2 cifrasi tanlangan ');
}
```

```

        break;
    case 3 :
    {
        System.Console.WriteLine('3 cifrasi tanlangan ');
        break;
    }
}

```

Cikl operatorlari while, do while, for , foreach

While operatori – bul operator shart orinlansa cikl denesi aylanadi. Bunda shart aldinnan berilgen boladi. Misali: 100 den 5 ke deyin sanlarni shigariw izbe-izligi:

```

int a = 100;
while (a > 5)
{
    System.Console.WriteLine(a);
    a--;
}

```

Bul jerde, a—bul a=a-1 di orinlaydi System.Console.Writeline(a); a ni manisin ekranga shigaradi. Bul jerde a=100, a=99, a=98 ,..... A=6, ga deyin orinlaydi.

Do While operatori – bul operatorda cikl denesi aylanangannan keyin shart tekseriledi. Bunda shart aqirinda qoyilgan boladi. Misali:

```

int a = 100;
do{
    System.Console.WriteLine(a);
    a=a-4;
}while (a > 5)

```

For operatori – bul operator shegara aniq bolganda ham maniske putin sanlar qosilganda isleytugin cikl operatori bolip tabiladi. Maselen bizge 1 den 10 deyin sanlar kerek onda cikl operatori tomendegishe orinlanadi:

```

for (int a =1; a <= 5; a ++)
{
    System.Console.WriteLine(a);
}

```

```
}
```

Foreach – bul operator massivtiń barlıq elementlerin paydalanıw ushın paydalı. Máselen, massiv a 3 elementten turadı. Foreach cikli a massivtiń barlıq elementlerin izbe-iz x qa teńlestirip beredi.

```
int[] a = new int[] { 1,2,3 }; // nash massiv
foreach (int x in a)
{
    System.Console.WriteLine(x);
}
```

### Massivler.

C# ta massivler tipi System.Array menen belgilenedi. Olardı foreach ciklında paydalansa boladı. Massiv qásiyeti length massiv elementleri sanın beredi.

Bir ólshemli massivler ózinde mánislerdi vektor kórinisinde jaylastıradı. C# ta Dev C++ ke sáykes elementler sanı nólden baslanadı. Bir ólshemli massivler tómendegishe daǵazalanadı:

```
uint[] mas_name = new uint[100];
```

Bul jerde mas\_name – massiv atı, 100 elementler sanı. C# tilinde massiv 0 den baslanǵanlıtan birinshi elementi mas\_name[0]; aqırǵısı mas\_name[99] boladı.

C# tilinde massivti toltırıwdıń úsh túri bar:

```
int arr_name = new int[2];
```

```
arr_name[0] = 1;
```

```
arr_name[1] = 2;
```

```
int[] arr_name = new int[] { 1,2};
```

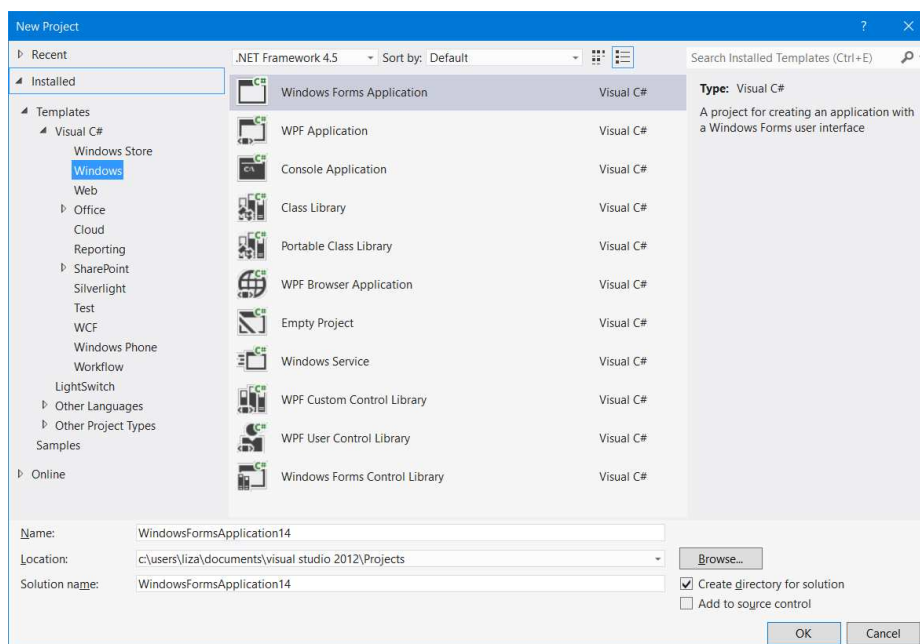
```
int[] arr_name = { 1,2};
```



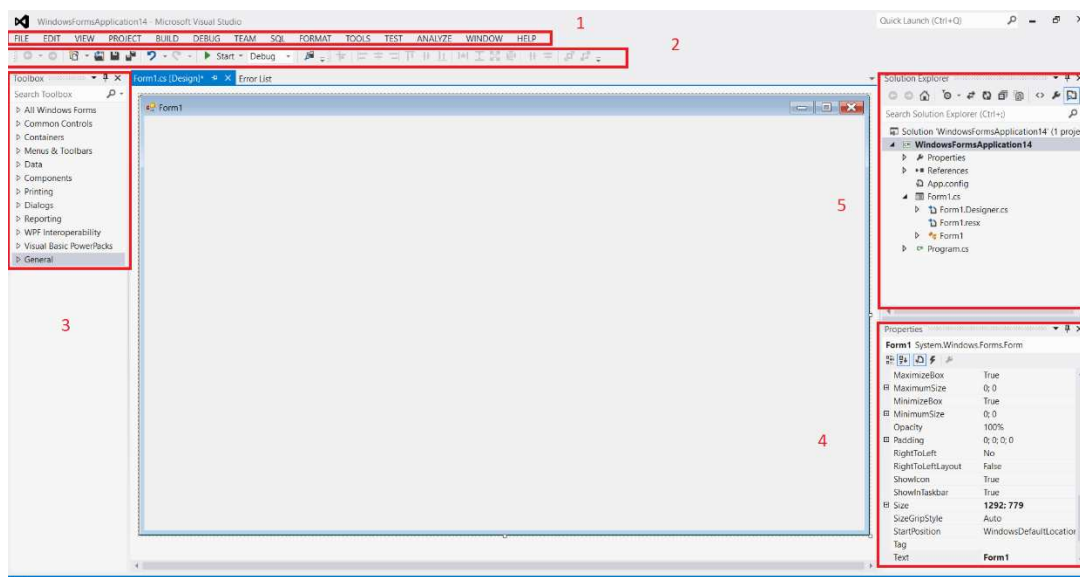
Tuwrımúyeshli massivler – bul massivler eki ólshemli massivler bolıp tabıladı. Olar tómendegishe daǵazalanadı:

```
int[, ] arr_name_1 = new int[4,2];  
int[, ] arr_name_2 = { {0, 1, 2, 3}, {0, 1, 2} };  
  
for(i = 0; i < arr_name_1.GetLength(0); i++){  
    for (j = 0; j < arr_name_1.GetLength(1); j++){  
        {  
            System.Console.WriteLine(arr_name_1[i,j]);  
        }  
    }  
}
```

### 2.3. C# tilinde WindowsForm Applicationda programma jaratıw



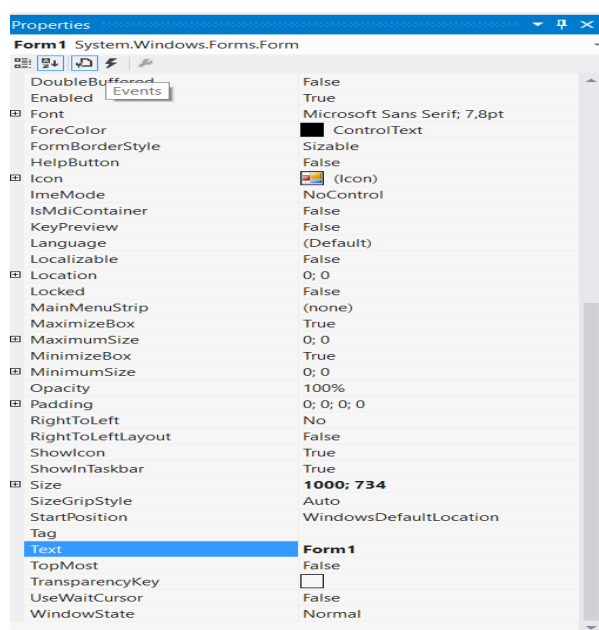
C# ob'ektke baǵdarlangan programmalaştırıw tili interfeysi:



C# ob'ektke bağdarlangan programmalaştırıw tili bas jumıs ornı elementleri:

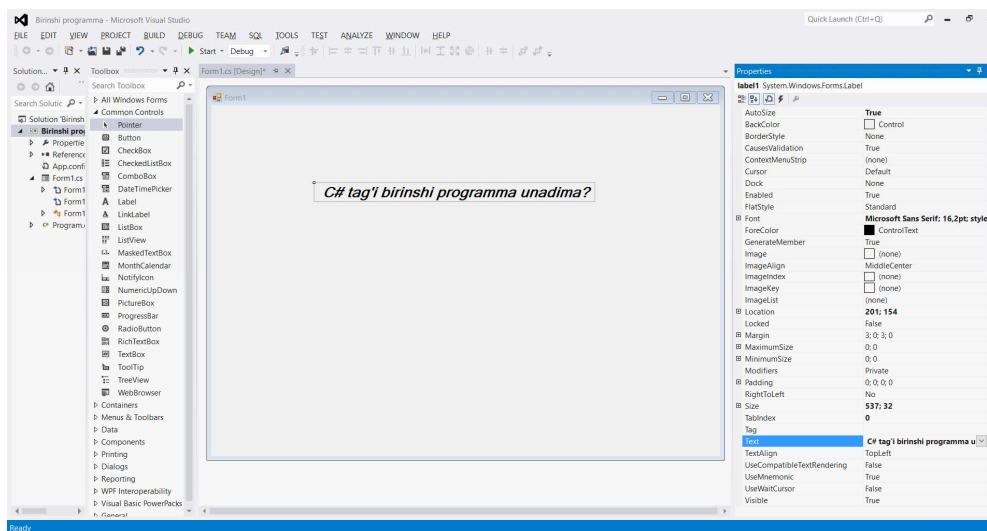
1. Menyular qatarı
2. Instrumentler paneli
3. Toolbox (basqarıw elementleri paneli)
4. Properties (Qásiyetler)
5. Solution Explorer

Belgilengen ob'ekt qásiyetleri Properties elementinde kórinedi. Events – bul (sobit'e) yaǵniy waqıyalar degen edi.

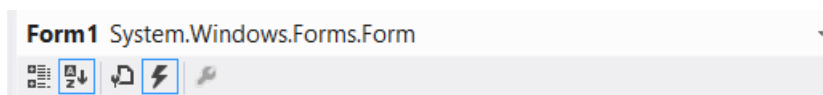


Hár bir komponenta ústine tıshqansha shep tárepın bir márte basılsa onda bul komponenta qásiyeti kórinedi, al tıshqansha shep túymesin eki mártebe basılsa onda bul

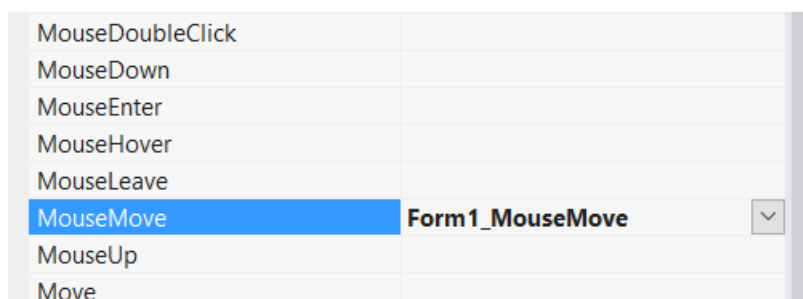
komponentaga programma kodi jaziladi. Label komponentasina bir marte basip onin tekst qasiyatin C# tagi birinshi programma atina o'zgertemiz.



Keyinen forma qasiyatinin events b'limine o'temiz.



Keyin bul qasiyetler toplamidan MouseMove niin on qaptalina tishqansha shep t'ymesin 2 marte tez-tez basamiz:



Keyin t'omendegi programma kodi kiritiledi.

```
private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    textBox1.Text = e.X.ToString();

    textBox2.Text = e.Y.ToString();
}
}
```

‘Awa’ atlı knopkağa tómendegi programma kodı kiritiledi.

```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show("Raxmet");

    Application.Exit();
}
}
```

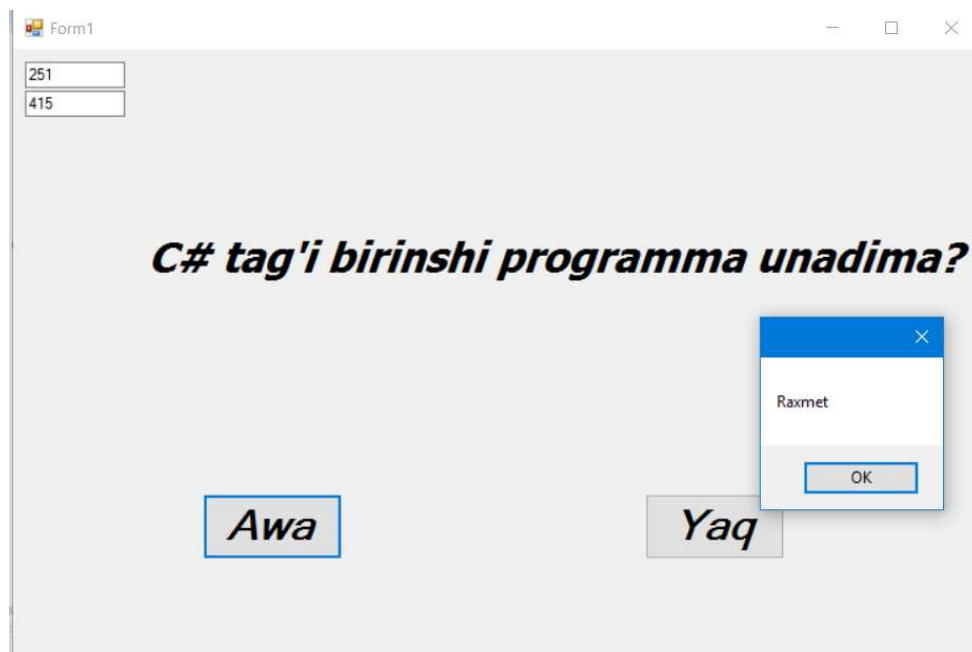
‘Yaq’ atlı knopkağa tómendegi programma kodı kiritiledi:

```
private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("Senba", "Ex!!!", MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
```

Nátiyjeni “Start” túymesin basıw arqalı kóremiz.



Awa túymesin basqandağı nátiyje:



## Eki ólsheмли графикалық об'ектлерді шақыру

C# программасында екі ólsheмли об'ектлерді шақыруда PictureBox, openFileDialog, menustrip, toolstrip компоненталары керек болады. Бунın' ushın еń дáslep proekt atın "2d графика" деп атаймыз.

PictureBox компонентасы графикалық об'ектлерді кórsetiw ushın арналған. Оl bmp, jpg, gif, png formattaғы графикаларды кórsetiw ushın арналған,

PictureBox компонентасы қásiyetleri:

- Image – Image об'ekt tipin ornatađı;
- ImageLocation – súwrettiń ornına bađdar beredi. Máselen kompyuterde yamasa internette;
- InitialImage – Image de turған súwretti kórsetedi
- ErrorImage: Eger bas súwret PictureBox qa júklenbese onda ErrorImage degi súwret kórsetiledi.

Súwretti kod арqalı PictureBox qa шақыру:

```
PictureBox1.Image = Image.FromFile("C:\\Users\\Liza\\Pictures\\12.jpg");
```

Súwret ólshemlerin ornatiw ushın *PictureBox* компонентасынın *SizeMode* қásiyeti elementleri:

- *Normal* – súwret PictureBox компонентасынın joqarǵı shep múyeshinde jaylasadı, hám súwret ólsheми ózgermeydi, Eger PictureBox компонентасы úlken ólsheмде bolsa onda оń hám tómengi táreplerde bos orınlar payda boladı, eger kishkentay bolsa onda súwret qırqıladı.

- *StretchImage* – súwret PictureBox компонентасы ólsheмlerine sáykes úlkeyedi yamasa kishireyedi;

- *AutoSize* – PictureBox компонентасы súwret ólsheмlerine sáykes ózgeredi;

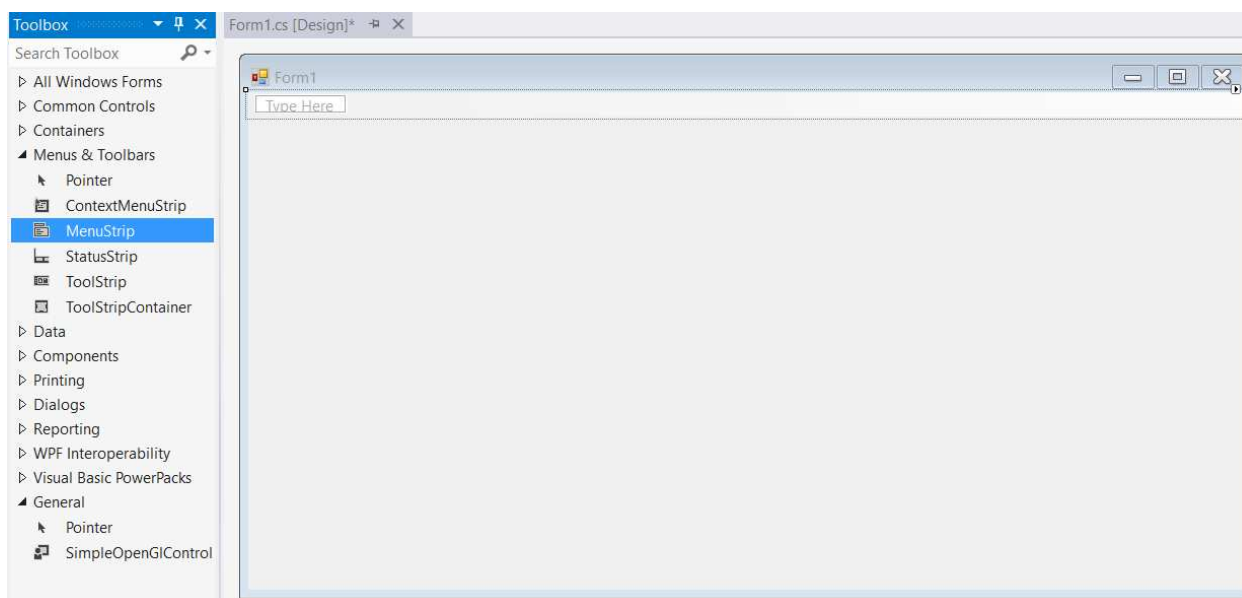
- *CenterImage* - eger PictureBox súwretten kishkene bolsa, onda súwrettiń ortası kórinedi, al keri jađdayda súwret PictureBox ortasında jaylasadı;

- *Zoom* – Súwret PictureBox ólsheмlerine qaray proporciyanı buzbaған halda ózgeredi,

*MenuStrip* komponentası qásiyetleri:

- *Dock* – menyudi formanıń bir tárepine jaylastırıw;
- *ShowItemTollTips* – Menyudiń hár bir elementi ushın járdemshi sózler kórsetilema yáki kórsetilmeytuǵınlıǵın támiyinleydi;
- *Stretch* = paneldi konteyner uzınlıǵına deyin sozadı;
- *TextDirection* – menyu punktlerindeki tekst baǵdarın beredi;

Dáslep proektimizge *MenuStrip* komponentasını jaylastıramız:



Keyinen oǵan bir márte basamız hám Fayl, Juklew atlı 2 menyu kiritemiz. Fayl menyusına Shıǵıw atlı buyırıq kiritemiz. Juklew menyusına jpg formatta hám png formatta atlı buyırıqlardı kiritemiz:



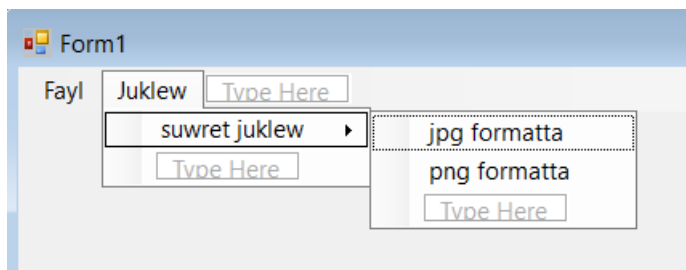
Fayl Shıǵıw buyırǵı kodın tıshqansha shep túymesin 2 márte basıw arqalı ašamız hám tómendegi kodımızdı kiritemiz:

```
private void shigiwToolStripMenuItem_Click(object sender, EventArgs e)
{
```

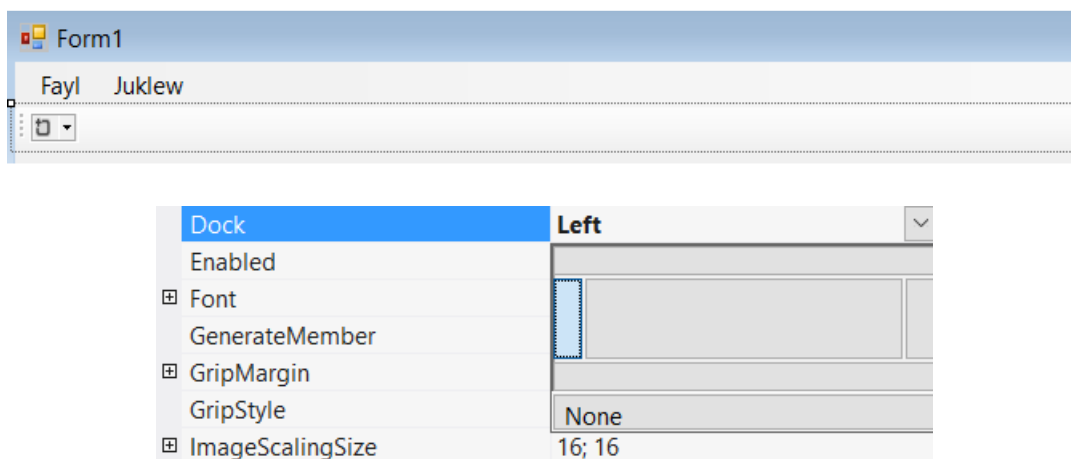
```

    Application.Exit();
}

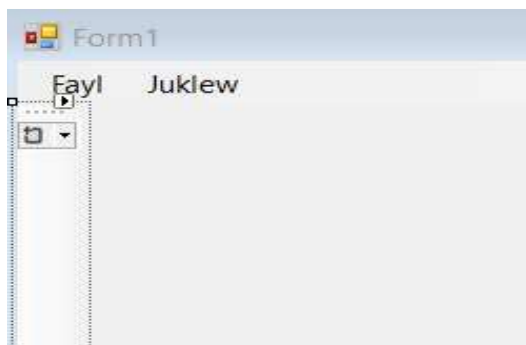
```



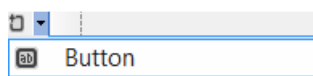
Keyinen ToolStrip komponentasın jaylastıramız hám onın qásiyeti Dock járdeminde shep tárepke jaylastıramız:



Nátiyjede ToolStrip komponentası konteyneri forma shep tárepinde jaylasadı:



Keyingi adımda ToolStrip komponentasına 2 button jaylastıramız bul knopkalar jpg hám png formatlardı júklew ushın paydalanıladı:



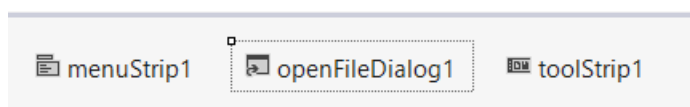
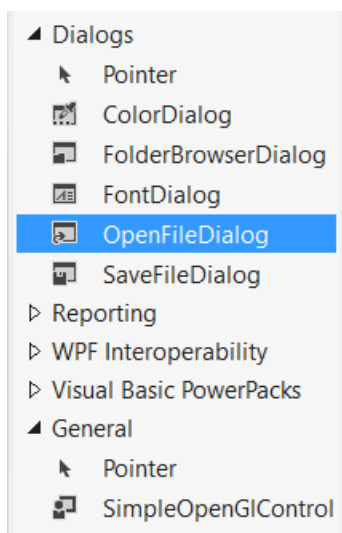
Endi knopkalarğa arawlı súwretlerdi qoyamız, ol Image qásiyeti járdeminde ámelge asırıladi.



Nátiyje:

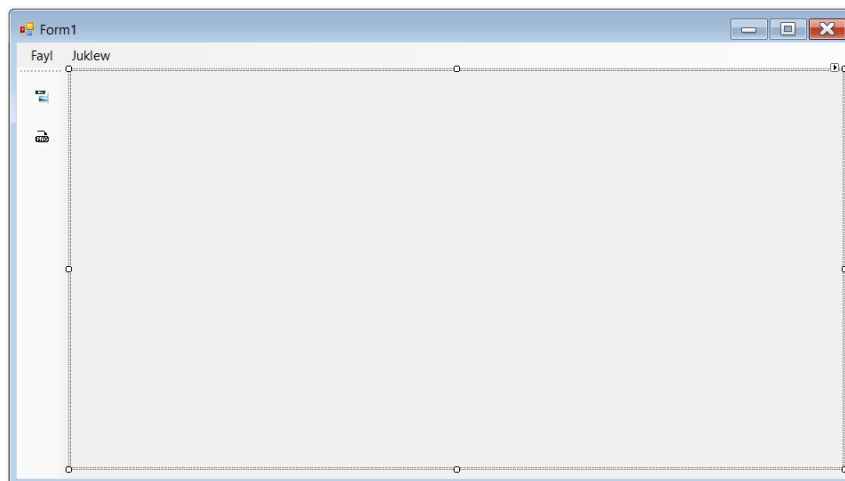


Endigi nábette eki ólshemli grafikanı ashıwshi OpenFileDialog komponentasın jaylastıramız openFileDialog komponentası formanıń qálegen jerin qoyıwǵa boladı biraq ol formada kórinbeydi, onı belgilew ushın formaǵa jaylastırılǵan menyu, toolStrip qatarında kórinedi:



Keyingi nábette PictureBox komponentasın tómendegi kóriniske sáykes jaylastıramız:





PictureBox komponentası qásiyeti Image di StretchImage etip, jaylastırılğan súwret PictureBox komponentası ólshemine sáykes ózgertiriledi.

Endigi náwbette súwretti júklew kodın kiritemiz:

Súwretti júklew funkciyası

Image MemForImage;

```
private void LoadImage(bool jpg)
{
    openFileDialog1.InitialDirectory = "c:";
    if (jpg)
    {
        openFileDialog1.Filter = "image (JPEG) files (*.jpg)|*.jpg|All files (*.*)|*.*";
    }
    else
    {
        openFileDialog1.Filter = "image (PNG) files (*.png)|*.png|All files (*.*)|*.*";
    }
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        try
        {
            MemForImage = Image.FromFile(openFileDialog1.FileName);
            pictureBox1.Image = MemForImage;
        }
    }
}
```

```

catch (Exception ex)
{
    MessageBox.Show("Fayldi ju'klew mu'mkin emes: " + ex.Message);
}
}
}

```

Keyin hár bir knopka hám menyudağı buyırıqlarğa LoadImage (true/false) qoyıp shıǵamız:

Jpg formatta buyırığı kodı

```

private void jpgFormattaToolStripMenuItem_Click(object sender, EventArgs e)
{
    LoadImage(true);
}

```

Png formatta buyırığı kodı

```

private void pngFormattaToolStripMenuItem_Click(object sender, EventArgs e)
{
    LoadImage(false);
}

```

ToolStripta jaylasqan knopkalarğa:

```

private void toolStripButton3_Click(object sender, EventArgs e)
{
    LoadImage(false);
}

```

```

private void toolStripButton2_Click(object sender, EventArgs e)
{
    LoadImage(true);
}

```

Nátiyjeni kórıw ushın F5 túymesin basamız:



### **III-BAP. C# tiline TaoFrameWork bibliotekası járdeminde úsh ólshemli ob'ekterdi programmalaştırıw járdeminde jaratıw, vizuallastırıw hám shaqırıw**

#### **3.1. TaoFramework bibliotekası ob'ektleri**

Tao Framework – bul ashıq kodlar bibliotekası bolıp, ol .NET Framework C# ortalığı ushın multimedialı programmalıq támiynatın qolaylı hám tez jaratıw ushın arnalğan. Búgingi kúnde Tao Framework .Net programmalaştırıw tili C# ortalığında OpenGL bibliotekası paydalanıwda eń jaqsı baǵdar bolıp tabıladı. Házirgi waqıtta bul biblioteka quramına multimedialı programma támiynatın jaratıwda kerek bolatuǵın barlıq zamanagoy elementleri bar:

OpenGL bibliotekası menen islew;

FreeGlut bibliotekası menen islew;

DevIL (OpenIL standartı tiykarında jeńil isleydi – Open Image Library) hám taǵı basqa eń jańa funkciyalar kiredi;

Tao Framework kiritilgen eń qızıqlı bibliotekalar:

OpenGL 2.1.0.12 – 2D hám 3D grakalardı vizuallastırıw hám programmalıq interfeys jaratıw ushın qolaylı; FreeGlut 2.4.0.2 – GLUT (OpenGL Utility Toolkit) ashıq kodlar tiykarında islewshi biblioteka;

DevIL 1.6.8.3 (OpenIL) – Súwretler menen islewdi támiyinlewshi biblioteka klassı. Házirgi waqıtta biblioteka 43 formattaǵı súwretler menen islew hám úsh ólshemli ob'ekterdi jaratıw ushın paydalanıladı;

Cg 2.0.0.0 – joqarı dárejeli til bolıp, ol teksturalar hám biyikliklerdiń sheyderleri menen islew ushın arnalğan;

OpenAL 1.1.0.1 – 3D dawıslardı hám EAX-effekterdi qayta islew ushın arnalğan programmalıq interfeys bolıp tabıladı;

PhysFS 1.0.1.2 – sistemalı fayllardı kiritiw/shıǵarıw ushın paydalanılatuǵın biblioteka, sonıń menen birge arxivlew, API. SDL 1.2.13.0 – multimedialı biblioteka

hám operacion sistemada multimedialıq programmalar dı jaratıw ushın aktiv paydalanıladı;

GNU/Linux ODE 0.9.0.0 – fizikalıq programmalıq interfeys bolıp, onıń tiykargı funkciyası ob’ektiń dinamikasın hám sistemalı sáykeslerdi anıqlaydı;

FreeType 2.3.5.0 – shriftlar menen islesiwshi biblioteka. Bul biblioteka X11 (standartlar instrumentler hám GUI (paydalanıwshi grafikalıq interfeys) jaratıw ushın jaratılğan;

FFmpeg 0.4.9.0 – hár qıylı formattağı audio hám video fayllar menen islewdi támiyinlewshi multimedialı biblioteka.

Kórip turğanıńızda Tao Framework bibliotekası multimedialıq ob’ektlerdi programmalastırıw ushın eń kúshli instrumentlerge iye kross-platforma bolıp tabıladı. Bul biblioteka menen islew, paydalanıwshı yamasa topar paydalanıwshi jıllar dawamında islep shıǵılğan algoritmlerden paydalanıwǵa boladı, bul programma ónimin jaratıw waqtın kóp márte tezletedi. Kompyuter grafikasın programmalastırıwdı úyreniwde bul biblioteka klassları júdá úlken ról atqaradı. Házirgi waqıtta C# ob’ektke baǵdarlangan programmalastırıw tili qolaylı hám tez islew múmkinshiligin beredi. .NET Framework ózinen aldınǵı programmalastırıw tilleriniń kemshiliklerin saplastırıp jaratılğan programmalastırıw tili bolıp, bul qásiyetler bul programmalastırıw tilin qolaylı hám eń jaqsı programmalastırıw tili dep atawǵa boladı.

OpenGL (Open Graphics Library) – qaraqalpaq tiline awdarganda ashıq grafikalıq biblioteka degen mánisti bildiredi. OpenGL – bul ozinde júzlegen funkciyanı jámlegen specifikaciya. Ol eki ólshemli hám úsh ólshemli kompyuter grafikasın paydalanıp programma jaratıw hám kross-platformasına iye programma interfeysi bar bolǵan specifikaciya bolıp tabıladı. Eń birinshi OpenGL 1992 – jılı úsh ólshemli ob’ektler menen jumıs alıp barıwshi Silicon Graphics Inc., kompaniyası tárepinen jaratılğan.

17 jıl dawamında biblioteka 3.1. versiyasına deyin kemshilikleri saplastırılıp keń kólemde paydalana basladı. Onın jaratıwshıları oǵan keńeytpe mexanizmin qostı, onıń

járdeminde (video kartalar islep shıǵıwshılar) OpenGL di qollawshı specifikalıq múmkinshilikler jarattı.

Házirgi waqıtta, OpenGL hám DirectX bibliotekaları MS Windows platforması ushın bir-birine básekelesip kelmekte.

Specifikaciya OpenGL 4.0 versiyası Khronos ta OpenGL ARB (Architecture Review Board) topar ishshileri tárepinen islep shıǵılǵan hám ol GLSL 4.00 OpenGL Shading tilin kiritti, ol grafika sapasın kúsheytiwshi GPU tezletiwshisin óz ishine aladı, bul programmalastırıw tezligin kúsheytiriwge arnalǵan edi.

### 3.2. C# ob'ektke baǵdarlangan programmalastırıw tiline úsh ólsheмли ob'ektlerdi jaratıw hám vizuallastırıw

Birinshi úsh ólsheмли ob'ektti C# ob'ektke baǵdarlangan programmalastırıw tiline shaqırıw. Mısal etip, Sferanı alamız:

Dáslep C:\Programm files (86)\TaoFramework\lib papkası ishindegi elementlerdi nusqalap, C:\Windows\System32 niń ishine kóshirip alamız.

Bul programmanı Tao Framework bibliotekasınıń islewin tekseriw ushın paydalanamız.

Keyin C# ob'ektke baǵdarlangan programmalastırıw tili ortalıǵında Tao Framework bibliotekasınan Tao.Opengl.dll; Tao.Platform.Windows.dll; Tao.FreeGlut.dll; orınlawshı faylların shaqıramız, yaǵnıy View menyusınan Solution Explorer di saylaymız. C# tıń jumısshı aynasında tómendegi kórinis payda boladı:

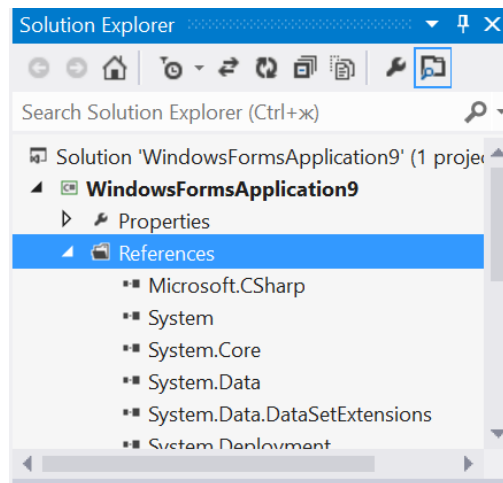
```
Using Tao.FreeGlut;
```

```
Using Tao.Platform.Windows;
```

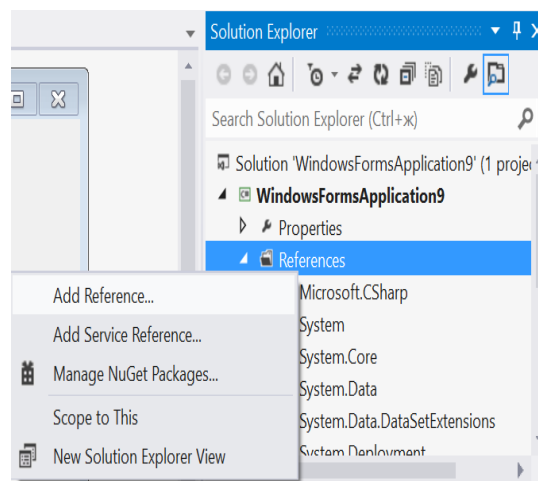
```
Using Tao.Opengl;
```

```
Using Tao.Devill;
```

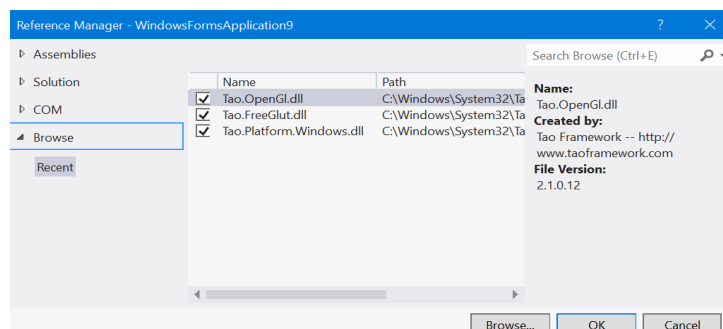
```
Kerek bolatuǵın dll fayllar;
```



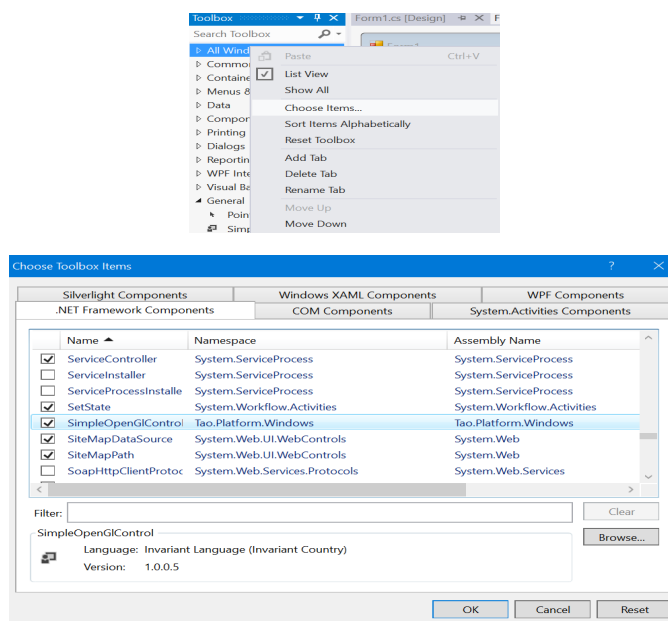
References buyırǵınıń ústine tıshqanshanıń oń tuymesin basamız, payda bolǵan kontekstli menyudan Add References di saylaymız. Payda bolǵan aynadan Browse tuymesini járdeminde Tao.Opengl.dll; Tao.Platform.Windows.dll; Tao.FreeGlut.dll; elementlerin shaqıramız keyin OK tuymesini basıladı.



1. Tao.OpenGL.dll – OpenGL bibliotekasını orınladıw ushın juwap beredi.
2. Tao.FreeGlut.dll – Glut bibliotekası funkciyaların orınladıw ushın juwap beredi, vizler onı renderdı iniciyalizaciya etiwge paydalanamız.
3. Tao.Platform.Windows.dll – Windows platformasında Vizuallastırıw ushın qollap quwatlaydı.



Bunnan soñ ToolBox bóliminen All Windows Forms ústine tıshqanshanıń oń túymesin basıp kontekstli menyudan Choose Items... ti saylaymız. Payda bolǵan aynadan SimpleOpenGLControl ge bayraqsha qoyamız:



Endigi nábette eki button hám SimpleOpenGLControl komponentaların jaylastıramız: Button komponenta qásiyeti Text ti tómendegishe ózgertemiz:

Button1 - Vizuallastiriw

Button2 - Shig'iw

SimpleOpenGLControl qásiyeti name di “AnT” dep ataymız. Bul bizlerde tez-tez paydalanatúǵın ob’ektimiz bolǵanlıqtan onıń atın ápiwayı etip qoyamız (programma kodında kóp paydalanıladı).

Endigi nábette biz OpenGL di inicializaciya etemiz:

```
public Form1()
{
    InitializeComponent();
    AnT.InitializeContexts();
}
```

Keyin Form1 jazıwınıń ústine eki márte basamız. Keyin tómendegi kodtı kiritemiz:

```
private void Form1_Load(object sender, EventArgs e)
```

```

{
    Glut.glutInit();
    Glut.glutInitDisplayMode(Glut.GLUT_RGB | Glut.GLUT_DOUBLE |
    Glut.GLUT_DEPTH);
    Gl.glClearColor(255, 255, 255, 1);
    Gl.glViewport(0, 0, AnT.Width, AnT.Height);
    Gl.glMatrixMode(Gl.GL_PROJECTION);
    Gl.glLoadIdentity();
    Glu.gluPerspective(45, (float)AnT.Width / (float)AnT.Height, 0.1, 200);
    Gl.glMatrixMode(Gl.GL_MODELVIEW); Gl.glLoadIdentity();
    Gl.glEnable(Gl.GL_DEPTH_TEST);
}

```

Bul jerde,

*Glut.glutInit()*; bul funkciyanı bibliotekanıń basqa funkciyaların paydalanbastan aldın paydalanıw kerek. Sebebi bul Glut bibliotekasıń iniciyalizaciya payda etedi.

```

    Glut.glutInitDisplayMode(Glut.GLUT_RGB | Glut.GLUT_DOUBLE |
    Glut.GLUT_DEPTH);

```

Bul programma kodı kórsetiw waziypasın atqaradı. Bizlerdiń misalda RGB rejimi vizuallastırıw ushın qoyılğan.

```

    Gl.glClearColor(255, 255, 255, 1);

```

Formadağı SMC di iniciyalizaciya etken soń, onı tazalawshı reń payda etedi.

Keyingi programma kodların túsiniw ushın, OpenGL saxnası jaratılıwı processın kórip shıǵıwımız kerek.

- Barlıǵı keńislikte kólemin ornalıwdan baslanadı, Máselen kameranı qandayda bir keńislikte kordinatalar tochkasına qoyǵanda kórinedi;
- Endi sol keńislikte qandayda bir ob'ektti ornatamız hám onı kóriw ushın qoyılğan kamera kordinataları úlken ról oynaydı;

```

    Gl.glViewport(0, 0, AnT.Width, AnT.Height);

```



Bul jerde bizler OpenGL bibliotekasına AnT (formada jaylasqan úsh ólshemli ob'ektlerdi vizuallastırıw ushın arnalğan komponenta) elementiniń belgili bir oblastında jaylasıwın kórsetedi. Model vizuallastırılıp atırğan waqıtta kórsetiw portın anıqlaydı.

*Gl.glMatrixMode(Gl.GL\_PROJECTION);*

GLMatrixMode funkciyası matricalıq rejim beriw ushın arnalğan: Aldında orınlawımız kerek bolğan jumıslar ushın matrica anıqlanadı. Bizlerdiń mısalda GL\_PROJECTION – matrica proekciyası

Endigi buyırıq glLoadIdentity(); járdeminde matricanı tazalaymız (funkciya matricanı birlik matricağa almastıradı) . Keyin proekciya tipin gluPerspective funkciyası járdeminde ornatadı.

*Gl.glLoadIdentity();*

*Glu.gluPerspective(45, (float)AnT.Width / (float)AnT.Height, 0.1, 200);*

gluPerspective funkciyası GLU – OpenGL Utility Library (GLU) da anıqlangan. Bul biblioteka OpenGL bibliotekasına qosımsha bolıp, ol keyingi jaratılğan funkciyalardı paydalanıwğa múmkinshilik beredi. Ol jaqsı tarqalatúǵın hám OpenGL bibliotekası menen bir júredi.

Endigi nábette bizlerdiń proekciya anıq bolǵanlıqtan keyin onı tómendegi funkciya menen tazalaymız:

*Gl.glMatrixMode(Gl.GL\_MODELVIEW);*

*Gl.glLoadIdentity();*

Aqırında bizler saxnamız odanda jaqsı kóriniwi ushın onıń glubina sın hám onıń reńin qoyamız.

*Gl.glEnable(Gl.GL\_DEPTH\_TEST);*

*Gl.glEnable(Gl.GL\_COLOR\_MATERIAL);*

Shıǵıw túymesine jazılatúǵın kodlar:

```
private void button2_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

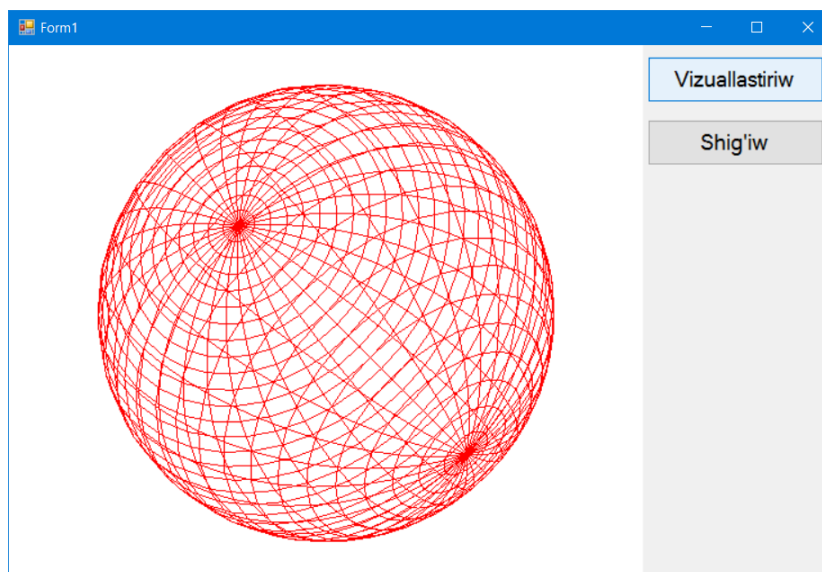
Vizuallastırıw túymesine jazılatúǵın kodlar:

```

private void button1_Click(object sender, EventArgs e)
{
    Gl.glClear(Gl.GL_COLOR_BUFFER_BIT | Gl.GL_DEPTH_BUFFER_BIT);
    Gl.glLoadIdentity();
    Gl.glColor3f(1.0f, 0, 0);
    Gl.glPushMatrix();
    Gl.glTranslated(0,0,-6);
    Gl.glRotated(45, 1, 1, 0);
    Glut.glutWireSphere(2, 32, 32); Gl.glPopMatrix();
    Gl.glFlush();
    AnT.Invalidate();
}

```

Nátiyjenı F5 túymesin basıw arqalı kóremiz:



Vizuallastiriw knopkasına qoyılğan kodlardıń túsindirmesi:

*glClear* funkciyası qanday parametrdagi buferlerdi tazalaw kerek ekenligi beriledi. Bizlerdiń mısalda glubina hám reń di ańlatadı.

*Gl.glColor3f(1.0f, 0, 0);* bul funkciya RGB tiykarında ob'ektimizge reń beredi. Bizlerdiń mısalda qızıl reńde boladı.

Gl.PushMatrix() funksiyası matricanı stek matricaga ózgertedi. Keyin stek matricadan Gl.PopMatrix() járdeminde matricanı qaytarıp alıw ushın paydalanıladı. Bul funksiyalar járdeminde keńislikte sıızılǵan sferanı kóshiriwde, matricanı ózgertpesten, kamera jaylasqan kórsetiw ornı ózgermeydi. Eger bul funksiyalar paydalanılmasa, onda hár bir vizuallastırıw knopkası basılǵanda kamera kóship ornın ózgerte beredi, keyin onı taba almay qalıwǵada boladı.

Matricanı stekte saqlap turıp, ob’ektti Z kósheri boyınsha 6 ǵa kóshiredi, keyin 45 gradusqa eki kosher x,y boyınsha burılıwın orınlaymız:

*Gl.glTranslated(0,0,-6);*

*Gl.glRotated(45, 1, 1, 0);*

Keyin sferanı FreeGlut bibliotekası járdeminde jaratamız:

Radiusı = 2

Meridianı = 32

Paraleli = 32

*Glut.glutWireSphere(2, 32, 32);*

Keyin OpenGL bibliotekası ob’ekti vizuallastırıwdı kútiwshi funkciyanı jazamız:

*Gl.glFlush();*

Endigi nábette vizuallastırıw saxnasın AnT (SimpleOpenglControl) komponentasına kórsetiw ushın jazıladı.

*AnT.Invalidade();*

Úsh ólshemli ob’ektlér ústinde geometriyalıq ózgerisler hám ámeller orınlaw

Geometriyalıq ózgerisler – bul keńislikte jaylasqan grafikalıq ob’ekttiń baǵdarı, ólshemi hám jaylasıw orınların anıqlaydı. Bunda keńislik ózgermey qaladı, grafikalıq ob’ekttiń strukturası sıyaqlı. Sonday etip, geometriyalıq ózgerislerde tek ǵana tochkalar menen islesetuǵın bolǵanlıqtan ob’ekt tipi qanday ekenin saylamaydı.

Tiykargı geometriyalıq ózgerislerge tómendegiler kiredi:

- Kordinatar kósheri baslanıwın kóshiriw (Перенос начало координат)
- Ólshemin ózgertiw (Масштабирование)
- Kordinatar kósheri boyınsha burıw (Поворот осей координат)

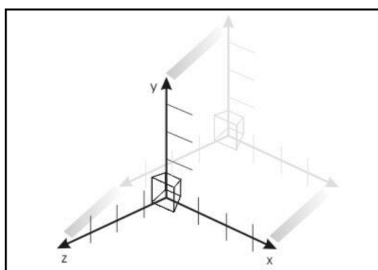
Kordinatar kósheri baslanıwın kóshiriw (Перенос начало координат) – bul buyırıqtı orınlaw arqalı kordinatar kósheri basin qálegen orıńǵa kóshiriwge boladı. Bul buyırıq orınlangannan keyin kordinatar kósherleri baǵdarları hám ólshemleri ózgermeydi.

Ólshemin ózgertiw (Масштабирование) - bul operaciyanı orınlaganda hár bir kósher  $x, y, z$  ler sáykes tárizde ózgeredi.

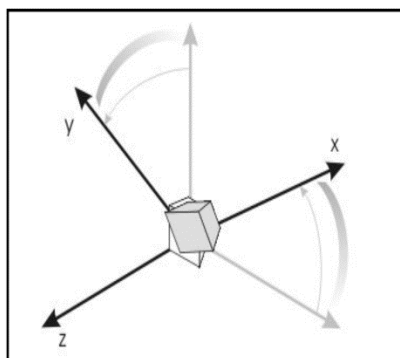
Kordinatar kósheri boyınsha burıw (Поворот осей координат) – bul operaciya keńislikte ob’ektti qálegen múyesh astında burıwǵa múmkinshilik beredi.

OpenGL Utility Toolkit (GLUT) – bul biblioteka ózinde qosımsha imkaniyatlar beriwshi qosımsha utilitalar menen bayıtılǵan, ol járdeminde OpenGL API da programma jaratılǵan. Bul biblioteka Silicon Graphics Inc kompaniyası jumısshısı Mark Kilgard tárepinen jaratılǵan. Bul biblioteka ózinde `gltranslate`, `glScale`, `glrotate` siyaqlı buyırıqlardı óz ishine aladı.

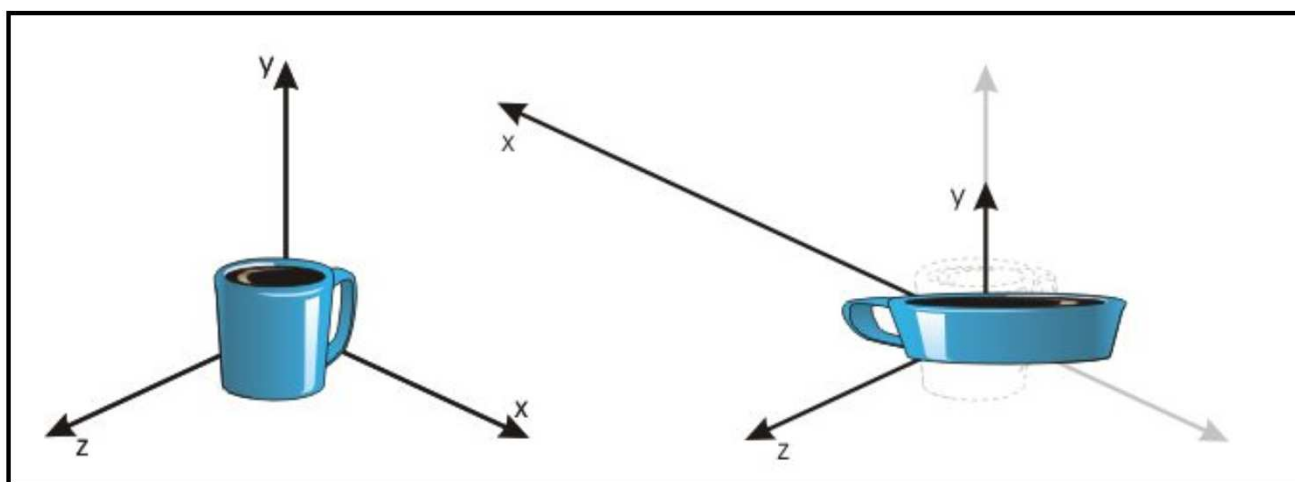
`glTranslate{fd}(type x, type y, type z);` (kóshiriw) – bul buyırıq  $x, y, z$  kósherleri boyınsha ob’ektti kóshiriw múmkinshiligin beredi.



`glRotate{fd}(type angle, type x, type y, type z);` (burıw) – bul buyırıq ob’ekttiń qandayda múyesh astında saat kórsetkishine qarsı  $x, y, z$  kósheri boyınsha burıladı.

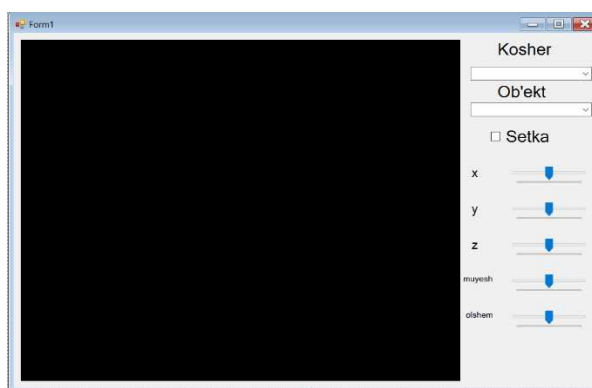


`glScale{fd} (TYPE x, TYPE y, TYPE z);` - bul buyırıq kordinatalar kóshetine sáykes ólshemin kishireytedi hám úlkeytedi.



`glTranslate`, `glRotate`, `glScale` funkciyalarınan paydalanıw arqalı úsh ólshemli ob'ektlardi programmalastırıwǵa mısál:

C# ta taza proekt jaratamız hám `simpleopenglcontrol` 2 combobox, 5 trackbar, 1 checkbox, sózlerdi jazıw ushın label komponentaları qollanıladı hám onıń kórinisi tómendegishe boladı:



Keyin Combobox1 ge x,y,z atların kiritemiz, Combobox2 ushın Cilindr, Sfera, Kub, Konus atların kiritemiz. Trackbar1,2,3 ler qásiyetinen minimum -50000 hám maximum 50000 ti kiritemiz. Al múyeshtiń qásiyetine -360; 360 ti kiritemiz. Ólshemge bolsa, -5000 hám 5000 di kiritemiz. Nátiyjede trackbar komponentaları kórsetkishleri sızıq ortasına kelip turadı.

Keyin Formamız ústine eki márte tez-tez basamız. Ol jerge tómendegi kodtı kiritemiz:

```
double a = 0, b = 0, c = -5, d = 0, zoom = 1;
int os_x = 1, os_y = 0, os_z = 0;

bool Wire = false ;

private void Form1_Load(object sender, EventArgs e)
{
    Glut.glutInit();
    Glut.glutInitDisplayMode(Glut.GLUT_RGB | Glut.GLUT_DOUBLE |
Glut.GLUT_DEPTH);
    Gl.glClearColor(255, 255, 255, 1);

    Gl.glViewport(0, 0, AnT.Width, AnT.Height);

    Gl.glMatrixMode(Gl.GL_PROJECTION);

    Gl.glLoadIdentity();

    Glu.gluPerspective(45, (float)AnT.Width / (float)AnT.Height, 0.1, 200);

    Gl.glMatrixMode(Gl.GL_MODELVIEW);
    Gl.glLoadIdentity();

    Gl.glEnable(Gl.GL_DEPTH_TEST);
    Gl.glEnable( Gl.GL_LIGHTING);
    Gl.glEnable( Gl.GL_LIGHT0);

    comboBox1.SelectedIndex = 0;
    comboBox2.SelectedIndex = 0;

    RenderTimer.Start();
}
```

Combobox1 komponentası jazılatuǵın kodlar:

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    switch (comboBox1.SelectedIndex)
    {
        case 0:
            { os_x = 1;
              os_y = 0;
              os_z = 0;
              break; }
        case 1:
            { os_x = 0;
              os_y = 1;
              os_z = 0;
              break; }
        case 2:
            { os_x = 0;
              os_y = 0;
              os_z = 1;
              break; }
    }
}
```

Keyin Draw yaǵnıy úsh ólshemli ob'ektlardi jaratıw funkciyası kiritiledi:

```
private void Draw()
{
    Gl.glClear( Gl.GL_COLOR_BUFFER_BIT | Gl.GL_DEPTH_BUFFER_BIT);
    Gl.glClearColor(255, 255, 255, 1);
    Gl.glLoadIdentity();
```

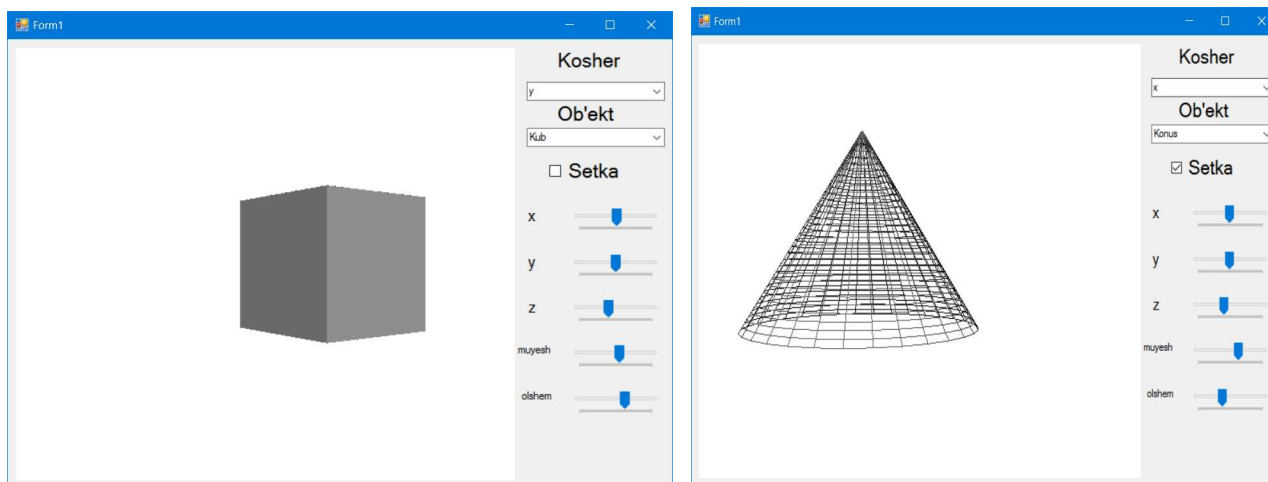
```

Gl.glPushMatrix();
Gl.glTranslated(a, b, c);
Gl.glRotated(d, os_x, os_y, os_z);
Gl.glScaled(zoom, zoom, zoom);
switch (comboBox2.SelectedIndex)
{
    case 0:
        { if(Wire)
            Glut.glutWireCylinder(1, 2, 32, 32);
          else Glut.glutSolidCylinder(1, 2, 32, 32);
          break ;
        }
    case 1:
        { if(Wire)
            Glut.glutWireSphere(2, 16, 16);
          else Glut.glutSolidSphere(2, 16, 16);
          break; }
    case 2: { if(Wire)
            Glut.glutWireCube(2);
          else Glut.glutSolidCube(2);
          break; }
    case 3: { if (Wire)
            Glut.glutWireCone(2, 3, 32, 32);
          else Glut.glutSolidCone(2, 3, 32, 32);
          break; }
    }
Gl.glPopMatrix();
Gl.glFlush();
AnT.Invalidate(); }

```

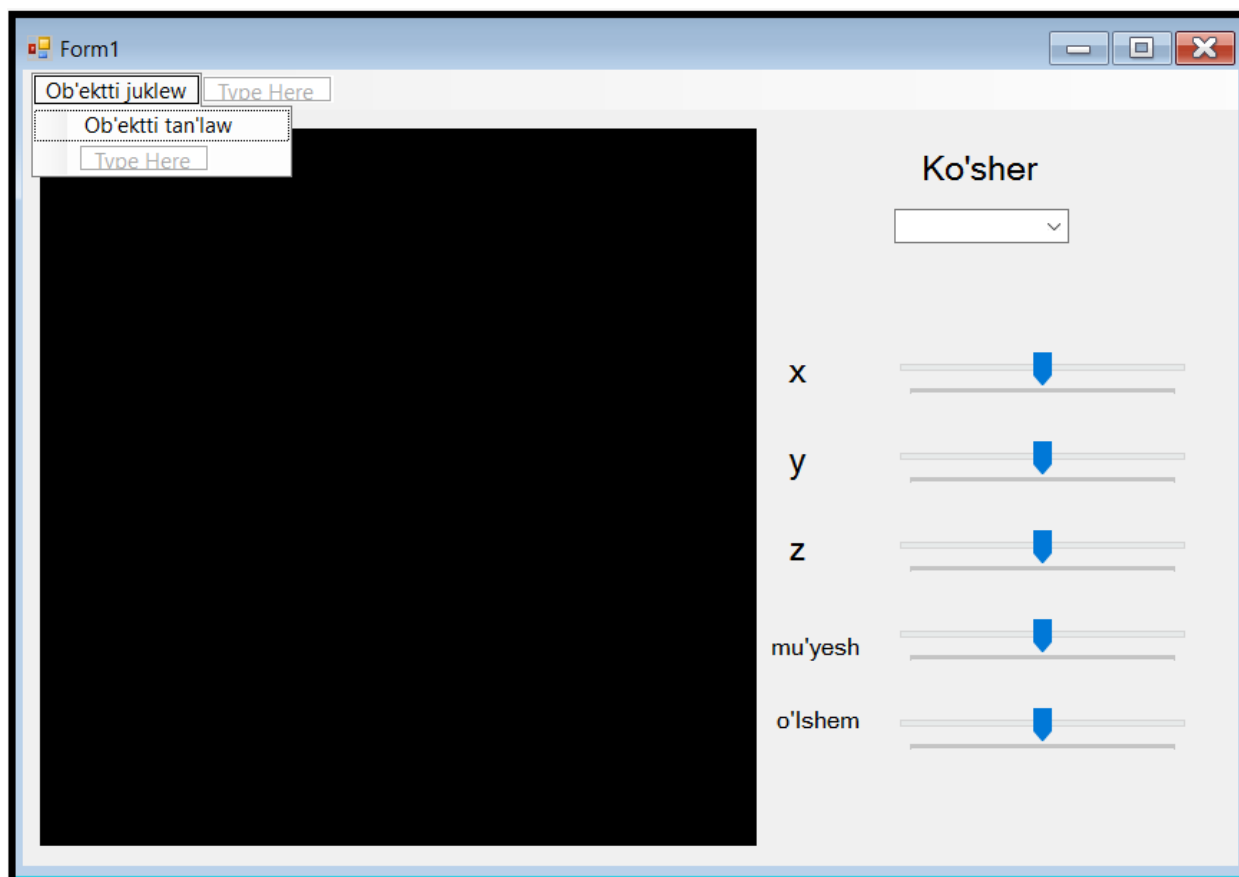


Bul programma bóleginde combobox2 komponentasındaǵı hár bir element (Cilindr, Konus, Kub, Sfera) ushın programma kodları switch saylaw operatorı menen tańlanadı. Nátiyjen F5 túymesin basıw arqalı kóremiz:



### 3.3. C# ob'ektke baǵdarlangan programmalaştırıw tiline úsh ólshemli ob'ektti shaqırıw hám programmalaştırıw

Bunın ushın bizlerge *Simpleopenglcontrol* 500x500 ólsheminde hám AnT atı menen formamızǵa jaylastıramız. III-baptıń 3.2 jaratılǵan programmadaǵı *trackbar*, *combobox*, *label*, *openfiledialog*, *timer* hám taza *menustrip* komponentaların tómendegishe jaylastıramız:



Dáslep ózgeriwshilerdi daǵazalaymız:

```
double a = 0, b = 0, c = -5, d = 0, zoom = 1;
```

```
int os_x = 1, os_y = 0, os_z = 0;
```

```
anModelLoader Model = null;
```

*Draw* funkciyasın *form1\_load* tan keyin yamas aldın kiritip alamız, bul funkciya sızıw ámelin orınlaydı.

```
private void Draw()
{
    Gl.glClear(Gl.GL_COLOR_BUFFER_BIT | Gl.GL_DEPTH_BUFFER_BIT);
    Gl.glLoadIdentity();
    Gl.glColor3i(255, 0, 0);
    Gl.glPushMatrix();
    Gl.glTranslated(a, b, c);
    Gl.glRotated(d, os_x, os_y, os_z);
    Gl.glScaled(zoom, zoom, zoom);
    if (Model != null)
        Model.DrawModel();
    Gl.glPopMatrix();
    Gl.glFlush();
    AnT.Invalidate();
}
```

Formamız ústine mishka shep tárepin eki márte tez-tez basamız payda bolǵan aynaǵa tómendegi programma kodın kiritemiz:

```
private void Form1_Load(object sender, EventArgs e)
{
    Glut.glutInit();
    Glut.glutInitDisplayMode(Glut.GLUT_RGB | Glut.GLUT_DOUBLE | Glut.GLUT_DEPTH);

    Il.ilInit();
    Il.ilEnable(Il.IL_ORIGIN_SET);

    Gl.glClearColor(255, 255, 255, 1);

    Gl.glViewport(0, 0, AnT.Width, AnT.Height);

    Gl.glMatrixMode(Gl.GL_PROJECTION);
    Gl.glLoadIdentity();

    Glu.gluPerspective(45, (float)AnT.Width / (float)AnT.Height, 0.1, 200);

    Gl.glMatrixMode(Gl.GL_MODELVIEW);
    Gl.glLoadIdentity();
}
```

```

Gl.glEnable(Gl.GL_DEPTH_TEST);
Gl.glEnable(Gl.GL_LIGHTING);
Gl.glEnable(Gl.GL_LIGHT0);

Gl.glBlendFunc(Gl.GL_SRC_ALPHA, Gl.GL_ONE_MINUS_SRC_ALPHA);
Gl.glEnable(Gl.GL_BLEND);
Gl.glEnable(Gl.GL_LINE_SMOOTH);
Gl.glLineWidth(1.0f);

comboBox1.SelectedIndex = 0;

openFileDialog1.Filter = "ase files (*.ase)|*.ase|All files (*.*)|*.*";

}

```

*OpenFileDialog1* komponentası filterine (ase files)ASE formatın hám (All files) barlıq fayllardı kiritemiz. Tao biblioteka elementleri qosıp alamız:

```

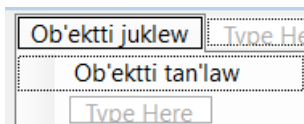
using Tao.OpenGl;
using Tao.FreeGlut;
using Tao.Platform.Windows;
using Tao.DevIL;

```

Keyin Timer komponentası qásiyeti name di RenderTimer atına ózgertemiz, hám interval qásiyetin 30 millisekundqa qoyamız:



Endi ob'ektti tańlaw bóliimine ótemiz:



Bul menustrip komponentası elementin basqanda orınlanatuǵın kodlar jazıladı:

```

private void obekttiTanlawToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        Model = new anModelLoader();
        Model.LoadModel(openFileDialog1.FileName);
        RenderTimer.Start();
    }
}

```

```

    }

}

```

Endigi náwbette combobox1 komponentasına jazılatuğın programma kodı.

```

private void comboBox1_SelectedIndexChanged_1(object sender, EventArgs e)
{
    switch (comboBox1.SelectedIndex)
    {
        case 0:
            {
                os_x = 1;
                os_y = 0;
                os_z = 0;
                break; }

        case 1:
            {
                os_x = 0;
                os_y = 1;
                os_z = 0;
                break;
            }

        case 2:
            {
                os_x = 0;
                os_y = 0;
                os_z = 1;
                break; }

    }
    Draw();
}

```

Hár bir trackBarğa óz ózgeriwshileri kodları jazıladı:

```

private void trackBar1_Scroll(object sender, EventArgs e)
{
    a = (double)trackBar1.Value / 1000.0;
    label4.Text = a.ToString();
    Draw();
}

```

```

private void trackBar2_Scroll(object sender, EventArgs e)
{
    b = (double)trackBar2.Value / 1000.0;
    label5.Text = b.ToString();
    Draw();
}

```

```

private void trackBar3_Scroll(object sender, EventArgs e)
{
    c = (double)trackBar3.Value / 1000.0;
    label6.Text = c.ToString();
}

```

```

        Draw();
    }

```

```

private void trackBar4_Scroll(object sender, EventArgs e)
{
    d = (double)trackBar4.Value;
    label6.Text = d.ToString();
    Draw();
}

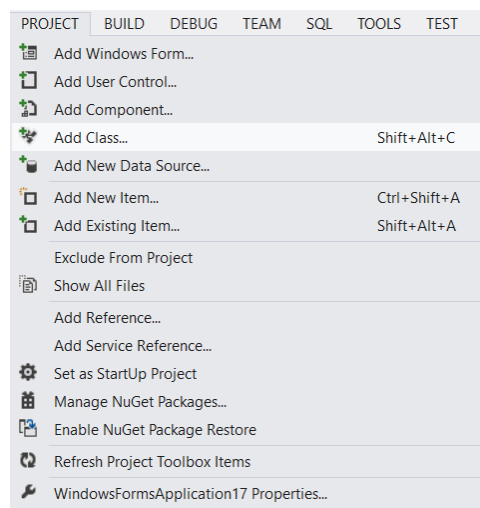
```

```

private void trackBar5_Scroll(object sender, EventArgs e)
{
    zoom = (double)trackBar5.Value / 1000.0;
    label6.Text = zoom.ToString();
    Draw();
}

```

Endi anModelLoader funkciyası ushın taza class jaratamız. Bunıń ushın menyular qatarınan PROJECT menyusına saylaymız hám ADD CLASS buyrığın saylaymız: Payda bolğan aynadan name qatarına AnModelLoader.cs atın beremiz hám Add túymesin basamız.



Name:

Keyin bizlerde kodlar aynası payda boladı. Ol jerge bizler Class LIMB ti payda etemiz. Bul Class bizlerge úsh ólshemli ob'ektlerdi ASE formattan shaqırıw ushın kerek boladı.

```

class LIMB
{
    public LIMB(int a, int b)
    {
        if (temp[0] == 0)
            temp[0] = 1;
        VandF[0] = a;
        VandF[1] = b;
        memcompl();
    }
    public float[,] vert;
    public int[,] face;
    public float[,] t_vert;
    public int[,] t_face;


    private int MaterialNom = -1;

    public int[] VandF = new int[4];
    private int[] temp = new int[2];

    private bool ModelHasTexture = false;

    public bool NeedTexture()
    {
        return ModelHasTexture;
    }

    public void SetMaterialNom(int new_nom)
    {
        MaterialNom = new_nom;
        if (MaterialNom > -1)
            ModelHasTexture = true;
    }

    public void createTextureVertexMem(int a)
    {
        VandF[2] = a;
        t_vert = new float[3, VandF[2]];
    }
}

```

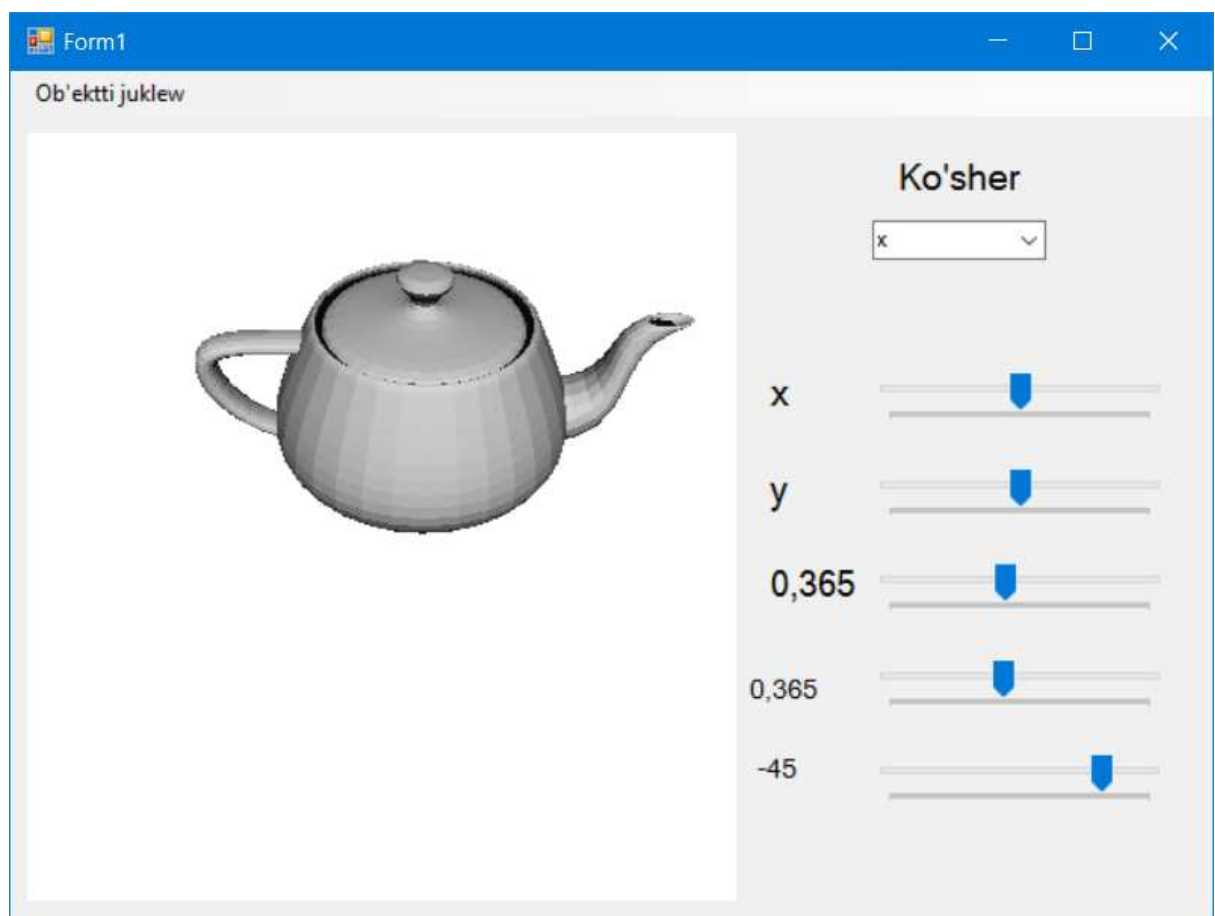
```

public void createTextureFaceMem(int b)
{
    VandF[3] = b;
    t_face = new int[3, VandF[3]];
}

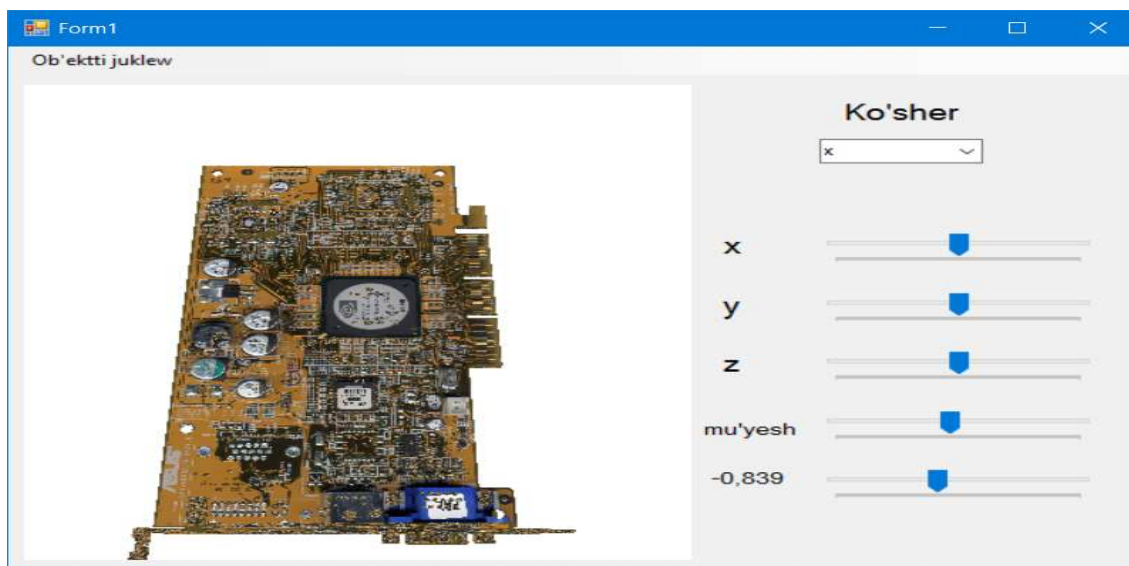
private void memcompl()
{
    vert = new float[3, VandF[0]];
    face = new int[3, VandF[1]];
}
public int GetTextureNom()
{
    return MaterialNom;
}
}

```

Nátiyjen F5 túymesin basıw arqalı kóremiz:



Basqada ASE modellerdi júklewge boladı:

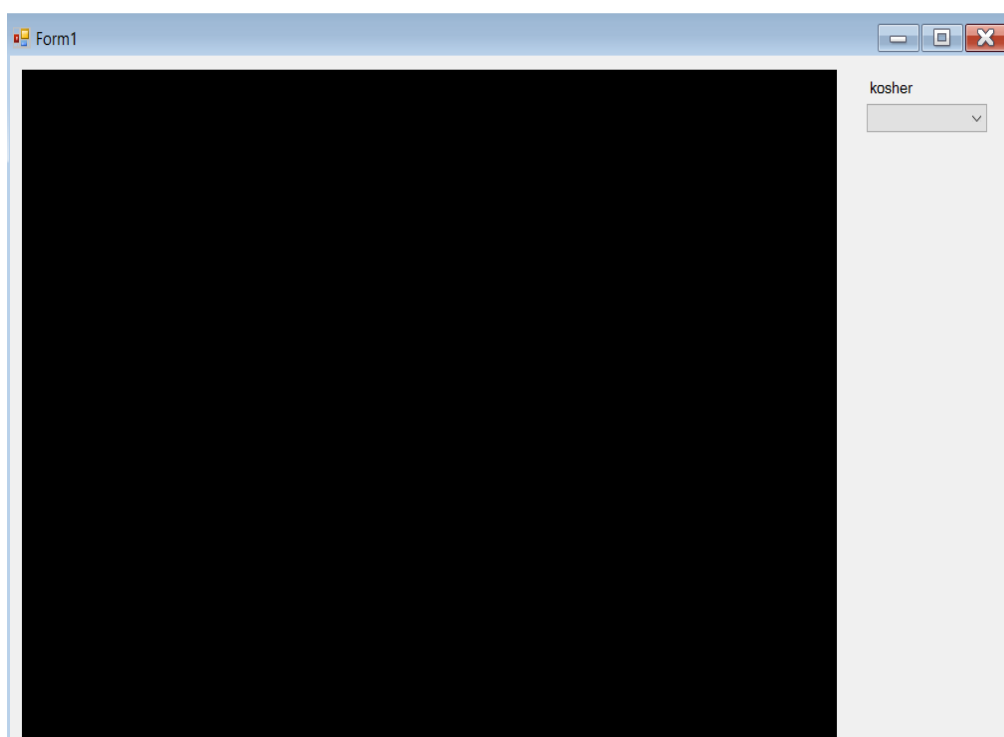


Bul programmağa sapalı úsh ólshemlerdi ákeliw ushın onı dáslep 3ds Max programmasında hár qıylı modifikatorlar ornatıp, sapalı kóriniske keltiriw kerek.

### Qosımshalar

C# tilinde OpenGL paydalanıp úsh ólshemli ob'ekt jaratıw hám onı x,z,a,s,d,w túymeleri járdeminde ólshemin ózgertiw, x kósheri, y kósheri boyınsha aylandırıw kórsetilgen.

Programmağa kerek elementler:





Programma kodı:

```
private float[,] GeomObject = new float[32, 3];  
  
private int count_elements = 0;  
  
private void AnT_KeyDown(object sender, KeyEventArgs e)  
{  
    if (e.KeyCode == Keys.Z)  
    {  
        CreateZoom(1.05f, comboBox1.SelectedIndex);  
    }  
    if (e.KeyCode == Keys.X)  
    {  
        CreateZoom(0.95f, comboBox1.SelectedIndex);  
    }  
    if (e.KeyCode == Keys.W)  
    {  
        CreateTranslate(0.5f, comboBox1.SelectedIndex);  
    }  
    if (e.KeyCode == Keys.S)  
    {  
        CreateTranslate(-0.5f, comboBox1.SelectedIndex);  
    }  
    if (e.KeyCode == Keys.A)  
    {  
        CreateRotate(0.5f, comboBox1.SelectedIndex);  
    }  
    if (e.KeyCode == Keys.D)  
    {
```

```

        CreateRotate(-0.5f, comboBox1.SelectedIndex);
    }
}

```

```

private void Form1_Load(object sender, EventArgs e)
{
    Glut.glutInit();

    Glut.glutInitDisplayMode(Glut.GLUT_RGB | Glut.GLUT_DOUBLE |
    Glut.GLUT_DEPTH);

    Gl.glClearColor(255, 255, 255, 1);

    Gl.glViewport(0, 0, AnT.Width, AnT.Height);

    Gl.glMatrixMode(Gl.GL_PROJECTION);

    Gl.glLoadIdentity();

    Glu.gluPerspective(45, (float)AnT.Width / (float)AnT.Height, 0.1, 200);

    Gl.glMatrixMode(Gl.GL_MODELVIEW);

    Gl.glLoadIdentity();

    Gl.glEnable(Gl.GL_DEPTH_TEST);

```

```

GeomObject[0, 0] = -0.7f;
GeomObject[0, 1] = 0;
GeomObject[0, 2] = 0;

```

```

GeomObject[1, 0] = 0.7f;
GeomObject[1, 1] = 0;
GeomObject[1, 2] = 0;

```

```

GeomObject[2, 0] = 0.0f;
GeomObject[2, 1] = 0;
GeomObject[2, 2] = 1.0f;

```

```

GeomObject[3, 0] = 0;
GeomObject[3, 1] = 0.7f;
GeomObject[3, 2] = 0.3f;

count_elements = 4;

comboBox1.SelectedIndex = 0;
    RenderTimer.Start();
}
private void CreateZoom( float coef, int os)
{
    float[,] Zoom3D = new float[3, 3];
    Zoom3D[0, 0] = 1;
    Zoom3D[1, 0] = 0;
    Zoom3D[2, 0] = 0;
        Zoom3D[0, 1] = 0;
    Zoom3D[1, 1] = 1;
    Zoom3D[2, 1] = 0;
        Zoom3D[0, 2] = 0;
    Zoom3D[1, 2] = 0;
    Zoom3D[2, 2] = 1;
        Zoom3D[os, os] = coef;

    multiply(GeomObject, Zoom3D);
}
private void CreateTranslate( float translate, int os)
{

```

```

for ( int ax = 0; ax < count_elements; ax++)
{
    GeomObject[ax, os] += translate;
}
}

private void CreateRotate( float angle, int os)
{
    float[,] Rotate3D = new float[3, 3];
    switch (os)
    {
        case 0:
        {
            Rotate3D[0, 0] = 1;
            Rotate3D[1, 0] = 0;
            Rotate3D[2, 0] = 0;
            Rotate3D[0, 1] = 0;
            Rotate3D[1, 1] = (float)Math.Cos(angle);
            Rotate3D[2, 1] = (float)-Math.Sin(angle);
            Rotate3D[0, 2] = 0;
            Rotate3D[1, 2] = (float)Math.Sin(angle);
            Rotate3D[2, 2] = (float)Math.Cos(angle);
            break;
        }
        case 1:
        {
            Rotate3D[0, 0] = (float)Math.Cos(angle);
            Rotate3D[1, 0] = 0;
            Rotate3D[2, 0] = (float)Math.Sin(angle);
            Rotate3D[0, 1] = 0;
            Rotate3D[1, 1] = 1;
            Rotate3D[2, 1] = 0;
            Rotate3D[0, 2] = (float)-Math.Sin(angle);

```

```

        Rotate3D[1, 2] = 0; Rotate3D[2, 2] = (float)Math.Cos(angle);
        break;
    }          case 2:          {
        Rotate3D[0, 0] = (float)Math.Cos(angle);
        Rotate3D[1, 0] = (float)-Math.Sin(angle);
        Rotate3D[2, 0] = 0;
        Rotate3D[0, 1] = (float)Math.Sin(angle);
        Rotate3D[1, 1] = (float)Math.Cos(angle);
        Rotate3D[2, 1] = 0;
        Rotate3D[0, 2] = 0;
        Rotate3D[1, 2] = 0;
        Rotate3D[2, 2] = 1;
        break;          }          }
    multiply(GeomObject, Rotate3D);
}

private void RenderTimer_Tick(object sender, EventArgs e)
{
    Draw();
}

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    AnT.Focus();
}

private void multiply(float[,] obj, float[,] matrix)
{
    float res_1, res_2, res_3;
    for (int ax = 0; ax < count_elements; ax++)
        { res_1 = (obj[ax, 0] * matrix[0, 0] + obj[ax, 1] * matrix[0, 1] + obj[ax, 2] *
matrix[0, 2]);

```

```
        res_2 = (obj[ax, 0] * matrix[1, 0] + obj[ax, 1] * matrix[1, 1] + obj[ax, 2] *  
matrix[1, 2]);
```

```
        res_3 = (obj[ax, 0] * matrix[2, 0] + obj[ax, 1] * matrix[2, 1] + obj[ax, 2] *  
matrix[2, 2]);
```

```
        obj[ax, 0] = res_1;
```

```
        obj[ax, 1] = res_2;
```

```
        obj[ax, 2] = res_3;
```

```
    } }
```

```
private void Draw()
```

```
{
```

```
    Gl.glClear( Gl.GL_COLOR_BUFFER_BIT | Gl.GL_DEPTH_BUFFER_BIT);
```

```
    Gl.glClearColor(255, 255, 255, 1);
```

```
    Gl.glLoadIdentity();
```

```
    Gl.glColor3f(0, 0, 0);
```

```
    Gl.glPushMatrix();
```

```
    Gl.glTranslated(0, 0, -7);
```

```
    Gl.glRotated(15, 1, 1, 0);
```

```
    Gl.glPushMatrix();
```

```
    Gl.glBegin( Gl.GL_LINE_LOOP);
```

```
    Gl.glVertex3d(GeomObject[0, 0],
```

```
        GeomObject[0, 1],
```

```
        GeomObject[0, 2]);
```

```
    Gl.glVertex3d(GeomObject[1, 0],
```

```
        GeomObject[1, 1],
```

```
        GeomObject[1, 2]);
```

```
    Gl.glVertex3d(GeomObject[2, 0],
```

```
        GeomObject[2, 1],
```

```

    GeomObject[2, 2]);
Gl.glEnd();
Gl.glBegin( Gl.GL_LINES);

Gl.glVertex3d(GeomObject[0, 0],
    GeomObject[0, 1],
    GeomObject[0, 2]);

Gl.glVertex3d(GeomObject[3, 0],
    GeomObject[3, 1],
    GeomObject[3, 2]);

Gl.glVertex3d(GeomObject[1, 0],
    GeomObject[1, 1],
    GeomObject[1, 2]);
Gl.glVertex3d(GeomObject[3, 0],
    GeomObject[3, 1],
    GeomObject[3, 2]);
Gl.glVertex3d(GeomObject[2, 0],
    GeomObject[2, 1],
    GeomObject[2, 2]);

Gl.glVertex3d(GeomObject[3, 0],
    GeomObject[3, 1],
    GeomObject[3, 2]);
Gl.glEnd();
Gl.glPopMatrix();
Gl.glPopMatrix();

```

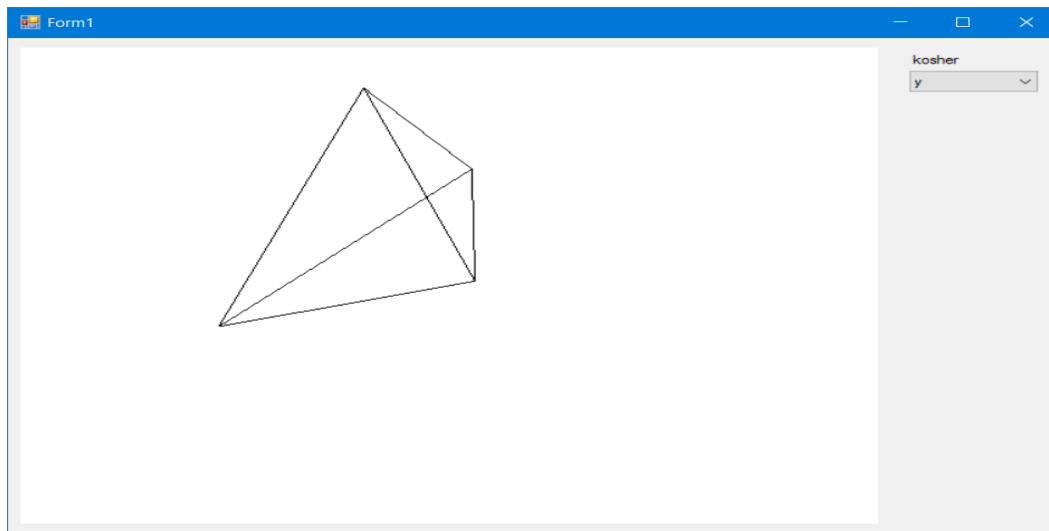
```

Gl.glFlush();
AnT.Invalidate();
}

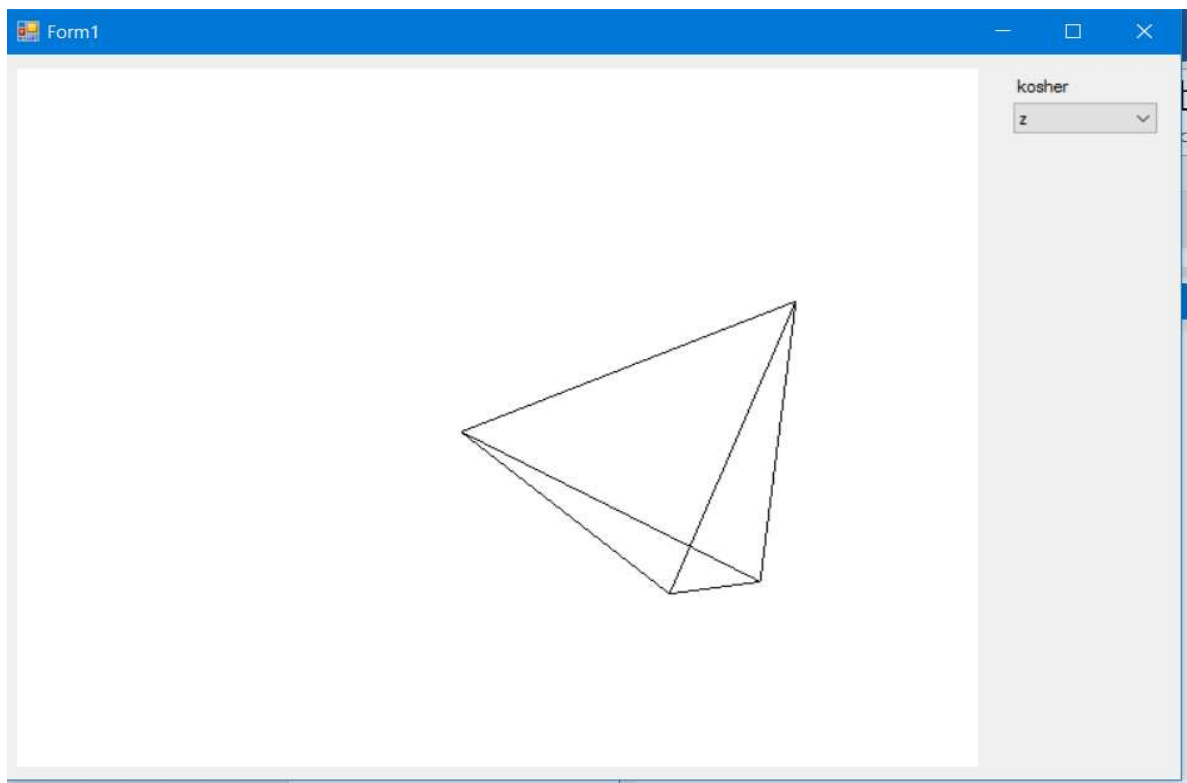
}

```

Nátiyje: w,a,s,d,z,x túymelerin basqan waqıtta: y kóshetine qoyğan waqıtta:



Z kósherin qoyğan waqıtta:





## JUWMAQLAW

Úsh ólsheмли ob'ektlerdi programmalastırıwda 3DS max hám C# ob'ektke baǵdarlangan programmalastırıw tilleri úlken ról oynaydı. Eger 3DS max programmasında qandayda bir model jaratılsa, onda onı C# ob'ektke baǵdarlangan programmalastırıw tilinde onı programmalastırıw da qıyınshılıq bolmaydı. Biraq C# ob'ektke baǵdarlangan programmalastırıw tilindegi bibliotekalar OpenGL, Tao Framework lardı izertlew talap etiledi.

Dissertaciyanıń birinshi babında Úsh ólsheмли ob'ektlerdi AutoDesk 3DS max programmasında úsh ólsheмли ob'ektlerdi jaratıw hám olardı eksport etiwge arnalǵan. Bul bapta sistemalı blok modeli mısál etip beriledi.

Jumıstıń ekinshi babı. C# ob'ektke baǵdarlangan programmalastırıw tilinde paydalanılatuǵın operatorlar, class lar , Windows Form da programmalar jatıw mısallar menen berilgen.

Úshinshi bap C# tiline TaoFrameWork bibliotekası járdeminde úsh ólsheмли ob'ektlerdi programmalastırıw járdeminde jaratıw, vizuallastırıw hám shaqırıw dep ataladı, onda C# ob'ektke baǵdarlangan programmalastırıw tilinde Tao Framework, Opengl járdeminde úsh ólsheмли ob'ektlerdi programmalastırıw hám vizualizaciya kórsetilgen hám bunda ápiwayıdan quramalıǵa metodı boyınsha programma jaratıw metodıkası islep shıǵılǵan.

# PAYDALANÍLGAN ÁDEBIYATLAR DIZIMI

## I. Ózbekistan Respublikasınıń nızamları

1. Elektron hujjat aylanishi to'g'risida, 29 – aprel 2004 – yil.
2. Avtomatlashtirilgan bank tizimida axborotni muhofaza qilish to'g'risida, 16 – noyabr 2005 – yil.

## II. Ózbekistan Respublikası Prezidenti pármanları hám qararlari

1. Oliy talim muassasalarining moddiy – texnik bazasini mustahkamlash va yuqori malakali mutaxassislar tayyorlash sifatini tubdan yaxshilash chora – tadbirlari to'g'risida, № PQ – 1533, 20 – may 2011 – yil.
2. Ózbekistan Respublikasın bunnan bılay da rawajlandırıw boyınsha háreketler strategiyası haqqında Ózbekistan Respublikası Prezidentiniń pármanı, Tashkent qalası, 2017-jıl 7-fevral.
3. Ilimler Akademiyasınıń jumısın, ilimiy izertlew jumısların shólkemlestiriw, basqarıw hám qarjılandırıwdı bunnan bılay da jetilistiriw ilajları haqqında Ózbekistan Respublikası Prezidentiniń qararı, Tashkent qalası, 2017-jıl 17-fevral.
4. Joqarı bilimlendiriw sistemasın bunnan bılay da rawajlandırıw ilajları haqqında Ózbekistan Respublikası Prezidentiniń qararı, Tashkent qalası, 2017-jıl 20-aprel.

## III. Ózbekistan Respublikası Prezidenti shıǵarmaları

1. Karimov I.A. Yuksak ma'naviyat engilmas kuch. – Toshkent: Ma'naviyat, 2008.
2. Karimov I.A. Vatanimiz bosqichma – bosqich va barqaror rivojlanishini tayinlash – bizning oliy maqsadimiz. – Toshkent: O'zbekiston, 2009.
3. Mirziyoev Sh.M. Sın talqılaw, qatań tártip – intizam hám jeke juwapkershilik – hár bir basshı jumısınıń kúndelikli qaǵıyası bolıwı kerek. – Tashkent: Ózbekistan, 2017.

## IV. Tiykargı ádebiyatlar

1. А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. [Язык программирования C#. Классика Computers Science. 4-е издание](#) = C# Programming Language

- (Covering C# 4.0), 4th Ed. — СПб.: [«Питер»](#), 2012.
2. Э. Стилмен, Дж. Грин. [Изучаем C#. 2-е издание](#) = Head First C#, 2ed. — СПб.: [«Питер»](#), 2012.
  3. Келли Мэрдок. Autodesk 3ds Max 2013. Библия пользователя = Autodesk 3ds Max 2013 Bible. — М.: [«Диалектика»](#), 2013.
  4. Бондаренко М. Ю., Бондаренко С. В., 3ds Max 2008 за 26 уроков (+CD), 1-е издание, Издательский дом «Диалектика», 2008.
  5. И. Е. ЖИГАЛОВ., И. А. НОВИКОВ., ПРОГРАММИРОВАНИЕ ТРЕХМЕРНОЙ КОМПЬЮТЕРНОЙ ГРАФИКИ владимир, 2016.

#### V. INTERNET SAYTLARI

1. <http://www.Ziyonet.uz>
2. <http://www.Lex.uz>
3. <http://www.Esate.ru>
4. <http://www.youtube.com>
5. <http://www.ru.wikipedia.org>