



TALLINNA TEHNIKAÜLIKOO
INSENERITEADUSKOND
Virumaa kolledž

**PYTHONI VÕIMALUSED BLENDERIS.
ÕPPEMATERJALIDE JA KURSUSE ARENDAMINE**
**Python capabilities in Blender. Educational materials and course
development**

ARUKAD SÜSTEEMID JA RAKENDUSINFOTEHNOLOOGIA ÕPPEKAVA
LÕPUTÖÖ

Üliõpilane: Katri Laanisto

Üliõpilaskood: 212634

Juhendaja: Valeria Juštšenko,
Haridustehnoloog/lektor

AUTORIDEKLARATSIOON

Olen koostanud lõputöö iseseisvalt.

Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneriplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

LIHTLITSENTS LÕPUTÖÖ ÜLDSUSELE KÄTTESAADAVAKS TEGEMISEKS JA REPRODUTSEERIMISEKS¹

Mina Katri Laanisto (sünnikuupäev: 20.02.1989)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose PYTHONI VÕIMALUSED BLENDERIS. ÕPPEMATERJALIDE JA KURSUSE ARENDAMINE, mille juhendaja on Valeria Juštšenko,
 - 1.1. reprodutseerimiseks säilitamise ja elektroonilise avaldamise eesmärgil, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta kolmandate isikute intellektuaalomandi ega isikuandmete kaitse seadusest ja teistest õigusaktidest tulenevaid õigusi.

¹ *Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingulise tegevusega ning lõputöö kaas- või ühisautori(d) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.*

SISUKORD

SISSEJUHATUS	5
1. METOODIKA.....	7
1.1 Moodle.....	7
1.2 Blender.....	10
1.3 Python	13
2. ÕPPEMATERJALIDE LOOMINE	14
2.1 Kursuse arendamine	15
2.2 Kursuse struktuur	16
2.2.1 Keskkonna seadistamine.....	16
2.2.2 Blender pythoni skriptimise põhialused	18
2.2.3 Objektide töötlemine Pythoniga	19
2.2.4 Funktsioonide ja tsükelite abil automatiseerimine	20
2.2.5 Uue tööriista loomine pythoni abil	22
2.2.6 Python ja animatsioonid	24
2.2.7 Pythoni skriptide eksportimine ja salvestamine	26
2.2.8 Lisamoodulid(add-ons)	29
2.2.9 Kasulikud lingid	31
2.3 Moodlesse loodud testid	31
KOKKUVÕTE	33
SUMMARY.....	34
KASUTATUD KIRJANDUSE LOETELU	35

SISSEJUHATUS

Tänapäeval on arvutid ja tarkvarad asendamatud paljudes eluvaldkonnades, sealhulgas 3D-modelleerimises. 3D-mudelite loomise tähtsus on pidevalt kasvanud, seda nii ehitusvaldkonnas, disainis, kunstis, arhitektuuris ja ka mööblitootmises. Inimesed loovad 3D mudelid nii hobi korras kui ka professionaalselt. Kõige selle tulemusena on kasvanud ka vajadus automatiseerida keerulisi 3D-modelleerimis ülesandeid ja suurenenud vajadus luua kohandatud tööriistu. Blender on üks populaarsemaid vabavaralisi 3D-modelleerimistarkvarasid, mida iseloomustab ulatuslik funktsionaalsus ja tasuta kasutamine. Lisaks on Blenderis Pythoni kasutamise võimalus, mis on tugevalt alahinnatud.

Käesolev töö keskendub Blender Python API-le. Probleem on selles, et puuduvad eesti keelsed Pythoni ja Blenderi kombineeritud kasutamise kohta õppematerjalid, mis sobiks väheste kogemustega inimestele. Lõputöö eesmärk on luua väheste kogemustega inimestele mõeldud e-kursus, mis tutvustab Pythoni kasutamist Blenderis praktiliste ülesannete kaudu. Uurimisobjektiks on Blenderi Python API ja selle rakendamisvõimalused 3D-modelleerimises ja animatsioonis ning õppes loomiseks on kasutatud e-kursuse formaati.

Lõputöö ülesanneteks on: 1) Blender python API põhjalik uurimine, rõhuga peamistel funktsioonidel, mis on olulised 3D mudelite loomise ja manipuleerimise automatiseerimisel. 2) Moodle'i keskkonnas interaktiivsete õppematerjalide koostamine, mis hõlmavad nii teoreetilist baasi kui ka praktilisi näiteid ja harjutusi. 3) Erineva raskusastmega ülesannete ja testide loomine, mis võimaldavad õppijatel hinnata oma teadmisi ja oskusi.

Lõputöö koostamisel kasutas autor dokumentatsiooni analüüsi, õppevideote vaatamist, erinevaid veebipõhiseid ressursse ja ka katse-eksituse meetodil skriptide loomist. Lisaks kasutas autor Blenderi 3D-modelleerimistarkvara, Pythoni programmeerimiskeelt ning Moodle õpikeskkonda kursuse loomiseks.

Töö uudsus seisneb e-kursuse loomises, mis keskendub praktilisele lähenemisele Pythoni kasutamisel Blenderis, pakkudes algajatele harjutusi ja ülesandeid, mis on otseselt seotud reaalsete 3D modelleerimise vajadustega, ning kombineerides need interaktiivses moodle'i keskkonnas.

Lõputöö metoodika peatükis annab autor esmalt ülevaate Moodle e-õppe keskkonnast. Kirjeldab selle peamisi omadusi, toob välja tugevused ja puudujäägid ning võrdleb Moodle'it teiste sarnaste keskkondadega. Seejärel tutvustab Blenderi 3D-modelleerimistarkvara, analüüsides selle tugevusi ja nõrkusi, ning annab ülevaate Blenderi Python API-st. Lisaks võrdleb Blenderit selle peamiste alternatiividega. Viimaks

annab lühikese ülevaate Pythoni programmeerimiskeelest, mille baasteadmised on e-kursuse edukaks läbimiseks vajalikud.

Võtmesõnad: Blender, Python, 3D modelleerimine, e-kursus, diplomitöö

1. METOODIKA

Veebikursuse loomisel lähtus autor sellest, et kursusel osaleja omab vähest kokkupuudet nii Blenderi kui ka pythoniga ning soovib, et saaks neid kahte omavahel ühendada. Kuid just see asjaolu toob esile probleemi: kuigi nii Blender kui ka Python on populaarsed ja võimekad tööriistad, on nende kooskasutamise õppimine väheste kogemustega inimestele sageli keeruline. Puudub terviklik õppematerjal, mis seoks praktilisi ülesandeid ja teooriat algajasõbralikul viisil. Kursuse läbimisega saab kindlasti hakkama ka inimene, kes ei ole kumbagi kasutanud, kuid see eeldab palju tööd ka dokumentatsiooniga. See iseseisev dokumentatsiooniga töötamine võib olla ajamahukas ja heidutav, eriti algajatele, kes vajavad selget ja struktureeritud juhendamist. Olemasolevad materjalid on tihti killustatud ja ei paku praktilist lähenemist, mis aitaks õppijal realselt teadmisi rakendada. Loodud materjalide juurde on lisatud ka dokumentatsiooni failid ning lisaks on Moodlesse tehtud ka kasulike linkide osa, kust on võimalik leida abistavaid materjale ja videoid.

Kursuse läbimise korral oskavad õpilased automatiseerida ja lahendada kergemaid ülesandeid Blenderis pythoni skriptide abil. Veebikursus sai loodud Moodle keskkonda just selle laia kasutuse tõttu. Kursus on jagatud kaheksasse blokki. Igal blokil on olemas pdf juhendid ja kui seal on lisatud ka harjutused, siis on peatükkide juurde lisatud ka pythoni failid võimalike lahendustega. Lisaks on lisatud ka mõned testid. Käesolev kursus püüab lahendada õppimise probleemi, pakkudes struktureeritud ja praktilist lähenemist Blenderi ja Pythoni kooskasutamise õppimisel. Kursuse ülesehitus ja materjalid on koostatud nii, et need toetaksid algajaid samm-sammult, pakkudes selgeid juhiseid ja lahendusi, et vähendada iseseisva õppimise raskust.

1.1 Moodle

Kursuse loomise keskkonnaks sai valitud Moodle (Modular Object-Oriented Dynamic Learning Environment) keskkond, sest seda kasutavad väga paljud õppeasutused. Moodle rahvusvahelise lehe järgi on Eestis 186 [1] kasutajat. Kokku on hetkel registreeritud 154221 aktiivset saiti 239 riigist[1]. Samuti kasutab moodle keskkonda ka Tallinna Tehnikaülikool.

Moodle on maailma kõige kohandatavam ja usaldusväärseim e-õppe lahendus, mis annab õpetajatele võimaluse meie maailma paremaks muuta[2]. Moodle ütleb ise enda kohta, et nad on avatud lähtekoodiga õppehaldussüsteem, mida saab tasuta alla laadida, muuta ja teistega jagada, mis on Moodle LMS väärtuste ülim väljendus, mis ühendab nende arendajate, süsteemiadministraatorite, koolitajate ja õppijate

kogukonda [2].

Moodle omadused [3]

- Tööriistad hindeliste ülesannete, viktoriinide ja õppetundide jaoks[3].
- Koostöö aruteluforumite, vestluste ja töötubade kaudu[3].
- Ühilduvus e-õppe sisu IMS/SCORM standarditega[3].
- Suurepärane dokumentatsioon ja tugevad turvaelemendid[3].

Moodle piirangud [3]

- Mittetehnilistel kasutajatel võib selle installimine ja haldamine olla keeruline[3].
- Algajatele võib selle liides ilma juhendamisetä üle jõu käia[3].
- Tuginemine seadistamisel ja sisu loomisel kursuse administraatorile[3].

Ideaalne isikupärastatud õppimiseks, toetades adaptiivseid funktsioone, nagu õpilaste vastustele kohandatud õppetunnid [3].

Moodle on ka populaarne see tõttu, et on piisavalt paindlik, et toetada väikesemahuliste (individuaalsed koolitajad) kuni suuremahuliste (ülikoolide) asutuste kasutuselevõttu [3].

Moodle peamisteks konkurentideks võiks nimetada TalentLMS ja Docebo. Nende mõlema kohta on kasutajad ise toonud välja eeliseid ja puuduseid, mis autor siinkohal ka välja toob.

TalentLMS puudused ja eelised on välja toodud kasutajate tagasisidest [4].

Eelised

- Administreerimine ja kasutajahaldus on lihtsad
- Paindlik
- Head aruanded
- Testide tegemise võimalus

Puudused

- Mõned lihtsad funktsioonid nõuavad käsitsi seadistamist(liiga konfigureeritav)
- Teavitussüsteemil on piirangud
- Kalendri integreerimine Outlooki/teamsiga on puudulik
- Arutelu ja koostöö vajab arendamist.

Docebo puudused ja eelised on samuti välja toodud lähtuvalt kasutajate tagasisidest

[5].

Eelised

- Hea kasutajaliides
- Analüütika pool on väga hea
- Kursuse sisu haldamine on väga kerge
- Võimalus hankida sisu erinevatest allikatest

Puudused

- Liiga kallis väiksematele organisatsioonidele
- Videokonverentsid vajaksid parendamist
- Puuduvad reaajas tehtavad treeningud
- Kehv kasutajatugi

Järgnev võrdlus (Tabel 1.1) on tehtud lehe capterra.com andmete põhjal, mis aitab ettevõtetel valida sobivat tarkvara[6].

	TalentLMS	Moodle	Docebo
maksumus	Olemas tasuta versioon	Olemas tasuta versioon	tasuline
Parim kasutus	sobib iga ettevõtte vajadustega, olenemata selle suurusest või koolituse tüübist.	Parim mittetulundusühingutele, erinevatele haridusasutustele (üldhariduskoolid, kutsekoolid, kõrgkoolid)	sobib eriti hästi keskmistele kuni suurtele organisatsioonidele, kes soovivad koolitada mitut õppivat sihtrühma ühelt platvormilt.
Kasutajate rahulolu	4.7 (541 tagasisidet)	4.3 (3336 tagasisidet)	4.5 (214 tagasisidet)
Parim sobivus	82%	91%	80%

Tabel 1.1 e-õppe keskkondade võrdlus[6]

Moodle kasuks räägib suuresti see, et seda on arendatud väga kaua ja paljud kunagised puudujäägid on nüüdseks parandatud. Lisaks on suur kogukond, kellelt saab abi küsida, kui isegi mõni asi tundub liiga keeruline.

Töö autoril puudub varasem kokkupuude e-kursuste loomisega, mistõttu on kindlat seisukohta keeruline võtta. Kursusi luues ja probleemide otsa takerdudes leidis kergelt abi erinevatest internetifoorumitest ja *youtube*'i videotest. See näitab, et tegemist on laialt kasutatud ja suure kogukonnaga e-õppe keskkond.

1.2 Blender

Blenderi tarkvara kodulehekülje andmetel on Blender tasuta avatud lähtekoodiga 3D loomise komplekt. See toetab kogu 3D-konveieri – modelleerimist, animatsioone, simulatsioone, renderdamist, koostamist ja liikumise jälgimist, isegi videotöötlust ja mängude loomist. Kogenud kasutajad kasutavad rakenduse kohandamiseks ja spetsiaalsete tööriistade kirjutamiseks Pythoni skriptimiseks mõeldud Blenderi API-d. Blender sobib hästi üksikisikutele ja väikestele stuudiotele. [7]

Käesolev töö keskendub ennekõike Blender API kasutamisele, mis võimaldab kasutajal kirjutada erinevad pythoni skripte, kas näiteks luua mõni uus funktsionaalsus või tööriist või automatiseerida mõnda protsessi.

Blenderi Python API võimalused[8]:

- redigeerida kõiki andmeid, mis kasutajaliides võimaldab (stseenid, võrgud, osakesed jne) [8].
- muuta kasutaja eelistusi, klahvikaarte ja teemasid[8].
- käivitada tööriistu koos enda seadistustega[8].
- luua kasutajaliidese elemente, nagu menüüd, päised ja paneelid[8].
- luua uusi tööriistu[8].
- luua interaktiivseid tööriistu[8].
- luua uusi renderdusmootoreid, mis integreeruvad Blenderiga[8].
- jälgida andmete ja nende omaduste muudatusi[8].
- määrata uusi sätteid olemasolevatele Blenderi andmetele[8].
- joonistada 3D-vaates Pythonit kasutades[8].

Blender on väga hea 3D modelleerimistarkvara, kuid on olemas ka teisi, alternatiivseid lahendusi. Siin kohal toob autor välja lehe geeksforgeeks.org andmetel 5 parimat Blenderi alternatiivi 3D-modelleerimiseks 2024. aastal koos nende plusside ja miinustega[9]. Nendeks viieks tarkvaraks on Autodesk Maya, SketchUp, ZBrush, Kino 4D, SolidWorks.

Autodesk Maya on professionaalne 3D-animatsiooni, modelleerimise, simulatsiooni ja renderdamise tööriistakomplekt, mis on loodud realistlike tegelaste ja kassahittide vääriliste efektide loomiseks. Fantastilistest olenditest laiaulatuslike maastike ja plahvatusohtlike lahingutekstideni – tippkunstnikud, modelleerijad ja animaatorid toetuvad Maya auhinnatud tööriistakomplektile, et äratada tänapäeva armastatuimad anima- ja live-action filmid, telesaated ja videomängud ellu. [10]

SketchUp on lihtne ja elegantne 3D-modelleerimisprogramm. Alustades põhikujudest ja vormidest, saab ehitada kõike alates pilvelõhkujatest kuni 3D-prinditavate tootemakettideni. [11]

Maxon ZBrush on Akadeemia auhinna võitnud ja tööstusharu juhtiv tööriist digitaalseks skulptuuriks, modelleerimiseks ja maalimiseks. Selle kohandatavad tööriistad ja funktsioonid, sealhulgas enam kui 200 patenteeritud pintslit, võimaldavad hulknurkadega töötada samamoodi nagu tegeliku saviga.[12]

Cinema 4D on professionaalne 3D-modelleerimise, animatsiooni, simulatsiooni ja renderdamise tarkvaralahendus. Selle kiire, võimas, paindlik ja stabiilne tööriistakomplekt muudab 3D-töövood disaini, liikuva graafika, VFX-i, AR/MR/VR-i, mänguarenduse ja igat tüüpi visualiseerimisprofessionaalide jaoks kättesaadavamaks ja tõhusamaks. [13]

SOLIDWORKS 3D CAD on võimas ja kasutajamugav projekteerimistarkvara. Selles sisalduvad funktsionaalsused võimaldavad lühendada arenduse aega ning tuua tooted kiiremini turule – vähemate kulude ja kõrgema kvaliteediga. [14]

All toodud tabelis (Tabel1.2) on näidatud nende tarkvarade peamisi plusse ja miinuseid. Tabel on koostatud geeksforgeeks.org info põhjal [9].

	plussid	miinused
Maya	Animatsiooni ja VFX-i poolest tuntud tööstusharu filmides, televisioonis ja mängudes. Võimas individuaalne tutvustusvarustus täiustatud taglase ja animatsioonifunktsioonidega. Suurepärased renderdusvõimalused koos Arnoldi renderdaja integratsiooniga mõistlike visuaalide jaoks.	Tellimuspõhine hinnakujundus võib olla kulukas, eriti stuudiote või vabakutseliste jaoks. Keeruline liides, mille kõverateadmised on amatöörisõbraliku tarkvaraga võrreldes järsemad. Pole parim lihtsate modelleerimisülesannete jaoks, kuna see on rikkalik
SketchUp	Äärmiselt kasutajasõbralik puhta ja intuitiivse liidesega. Kiire ja sujuv modelleerimisseade koos Push/Pull võimalusega lühiajaliseks 3D loomiseks. Suur laienduste kogu (tasuta ja tasuline) spetsiaalseks kasutuseks, näiteks sisustuse modelleerimiseks või renderdamiseks.	Pole parim kõrge eraldusvõimega orgaanilise moe kujundamiseks (vaoshoitud skulptuuriseadmed). Puuduvad täiustatud animatsioonivõimed, mistõttu on see keerukate animatsioonide jaoks palju vähem sobiv. Peamiselt suunatud arhitektuuri- ja sisekujunduse töövoogudele tasuline
ZBrush	Disainiplatvorm laia valiku võimsate pintslitega vormimiseks ja detailide kujundamiseks. Loob eriti selgelt eristuvaid orgaanilisi moodi, nagu tegelased, olendid ja anatoomia. Dünaamiline võrguotsus võimaldab kujundada liigset otsustusvõimet ilma üldiste jõudlusprobleemideta.	Puuduvad traditsioonilised modelleerimisseadmed, nagu väljapressimine, kaldnurk jne. Kallis abonemendipõhine hinnakujundus. Pole ideaalne algajatele oma disainikeskse tähelepanu tõttu
Cinema 4D	Populaarne animatsioonide, teksti ja nähtavate tulemuste arendamiseks mõeldud liikumisgraafika jaoks.	Litsentsitasemele tuginedes võib see olla kõrge hinnaga (Standard vs stuudio).

	Tõhusad 3-D meeste või naiste animatsioonivõimalused koos tugeva tegelaste süsteemiga (Cinema 4D R26).	Võib nõuda unikaalsete soovide (nt kõrgema inimese animatsiooni või renderdamise) jaoks lisapluginaid.
SolidWorks	Pakub parameetrilise modelleerimise funktsioone. Sisaldab insenerianalüüsi seadmeid Tugev integratsioon muu inseneritöövoogudes kasutatava CAD-tarkvaraga	Tavalise 3-D modelleerimistarkvaraga võrreldes järsem tutvumiskõver. Kallis abonemendipõhine hinnakujundus, mis on suunatud asjatundlikule kasutamisele. Ei sobi leidlikuks skulptuuriks, animatsiooniks ega videomängude või meelelahutuse 3D-mudelite arendamiseks.

Tabel 1.2 3D-modeleerimistarkvarade võrdlus

Samuti on Blenderil omad miinused ja plussid. Plussidena toob Pixune.com järgnevas aspektid [15]:

- Avatud lähtekoodiga ja tasuta: Blenderi üks olulisemaid eeliseid on selle avatud lähtekoodiga olemus, mis tähendab, et see on kõigile kasutajatele tasuta. See juurdepääsetavus on aidanud kaasa edukale ja kirglikule kogukonnale, soodustades pidevat arengut ja hulgaliselt kasutajate loodud ressursse[15].
- Mitmekülgsus: Blenderi kõik-ühes lähenemisviis muudab selle mitmekülgselt tarkvaraks, mis sobib 3D-sisu loomise erinevate aspektide jaoks, sealhulgas 3D-tegelaste modelleerimisteenused, animatsioon, renderdus ja komposiit. Selle skulptuuri-, VFX- ja simulatsioonivõimalused laiendavad veelgi selle potentsiaali erinevate projektide jaoks[15].
- Aktiivne kogukond ja tugi: Blenderi kogukond on tuntud oma aktiivse ja abivalmis olemuse poolest. Kasutajad pääsevad juurde arvukatele õpetustele, foorumitele ja kaaskunstnike loodud lisandmoodulitele, mis aitab oluliselt õppimist[15].
- Kohandatavus: Blenderi kasutajaliidest ja töövoogu saab laialdaselt kohandada, võimaldades kasutajatel kohandada tarkvara vastavalt oma eelistustele ja vajadustele. Kohandatud otseteed, lisandmoodulid ja isikupärastatud paigutused suurendavad tootlikkust ja kasutuskogemust[15].

Peamise miinusena tuuakse välja õppimiskõver: kasutajate jaoks, kes siirduvad muult tarkvaralt, võib Blenderi kasutajaliidesel ja paremklõpsu valikusüsteemil olla järsem õppimiskõver. Uutel kasutajatel võib kuluda aega, et tutvuda selle ainulaadse lähenemisviisiga suhtlemisele [15]. Seda miinust võiks osaliselt lahendada ka pythoni programmeerimise kasutamise võimalus, millele suures osas see töö ka keskendub.

1.3 Python

Python on laialt levinud programmeerimiskeel. TIOBE indeksi alusel on python 2024 aasta oktoobris kõige populaarsem programmeerimise keel [16]. Selles mängib rolli kindlasti keele lihtsus ja selle kasutamise võimalused. JalaSoft toob välja peamised põhjused, miks Python on nii populaarne programmeerimise keel[17]:

- Õppimise ja kasutamise lihtsus: Pythoni õppimiseks on väga palju erinevaid veebikursusi, õpetusi ja koole. Kui põhitõed on omandatud, siis saab seda juba kohe rakendada ja programme kiiresti luua[17].
- Väga mitmekülgne: Pythoni kasutusjuhtude loend on peaaegu lõputu. See on abiks ka ülesannete skriptimisel ja automatiseerimisel, statistilisel analüüsil, tarkvara prototüüpimisel ja testimisel ning mujal[17].
- Raamistikud ja teegid – tasuta keelena on võrgus saadaval tuhandeid kasulikke teeke, ressursse, pakette ja raamistikke, mis muudavad kodeerimise sujuvamaks ja lihtsamaks[17].
- Täiustatud automatiseerimisvõimalused – Pythoni ulatuslikud moodulid ja teegid vähendavad vajadust kulutada automatiseerimisskriptide kirjutamisel palju aega ja vaeva[17].
- Asjade Interneti ühilduvus – asjade internet on tänapäeval üks kuumemaid tehnoloogilisi suundi. Lingitud tehnoloogiate ja seadmete jõulise võrguna võimaldab see seadmetel üksteise ja pilve vahel suhelda[17].
- Aktiivne võrgukogukond – 2024. aasta veebruari seisuga on üle 8 miljoni Pythoni arendaja kogu maailmas, ületades arvuliselt kõiki teisi programmeerimiskeeli[17].

Pythonit kasutatakse veebiarenduseks (serveripoolne), tarkvara arenduses, matemaatikas ja süsteemi skriptimises[18].

2. ÕPPEMATERJALIDE LOOMINE

Käesoleva lõputöö eesmärgiks oli luua e-kursus, mis tutvustaks Pythoni kasutamise võimalusi Blenderis algajatele. See peatükk kirjeldab detailselt õppematerjalide loomise protsessi, mis oli keskne selle eesmärgi saavutamisel. Õppematerjalide loomisel lähtusin alguses Blenderi ja Pythoni baasteadmiste tutvustamisest, seejärel liikudes edasi keerukamate API funktsioonide juurde. Valisin ülesanded nii, et need seoksid teoreetilisi teadmisi praktiliste näidetega, võimaldades õppijatel kohe kätt proovida. Eriti oluline oli selgete ja üheselt mõistetavate juhiste loomine, et õppijad saaksid iseseisvalt ülesannetega hakkama.

Tagantjärele vaadates tõdeb autor, et esimesed ülesanded võinuks olla veelgi lihtsamad, et õppijad saaksid kindlama alguse. Üks efektiivseim lähenemine oli luua ülesanded, mis on suunatud praktilistele 3D-modelleerimisvajadustele, mida õppijad saavad kohe oma projektides kasutada. Moodle'i valik osutus heaks, kuna see võimaldas integreerida nii õppematerjalid, ülesanded, testid kui ka foorumid, luues tervikliku õpikeskkonna. Lisaks võimaldas Moodle ka paindlikku ülesehitust, kus iga teema on eraldi moodulina, mis aitab õppijal paremini orienteeruda. Kursus jaguneb kaheksasse moodulisse, millest igaüks käsitleb teatud teemat Blenderi Python API-s. Iga mooduli juurde on lisatud nii teoreetilised materjalid kui ka praktilised ülesanded, mis võimaldavad õppijatel kohe teadmisi rakendada. Kursuse struktuuri töötas autor välja pärast suure hulga materjalide läbitöötamist, oma tunnetuse järgi (Pilt 2.1).



Pilt 2.1 Moodles kursuse kastike

2.1 Kursuse arendamine

Õppematerjalide loomine sai alguse sellest, et töö autor tegi ise läbi mitu videokursust, et teema endale selgemaks teha ja näha, millised on peamised võimalused Blenderis, varasemalt on ülikoolis ühe õppeaine raames tutvustatud Blenderi kasutamise võimalusi.

Edasi sai tehtud plaan, kuidas võiks hakata välja nägema kursuse struktuur (Pilt 2.2) ning mida õppetükid endas sisaldada võiks. Õppematerjali koostamisel on kasutatud palju allikaid, mis on ka materjalid lõppu välja toodud, kes õppija saaks ka ise vajadusel neid materjale kasutada. Materjalide loomisel on autor proovitud läheneda lihtsalt ja loovalt ning samm-sammult kõik kirja pannud. Enne iga uue teema alustamist otsis autor lisainfot ja vaatas antud teema kohta ka dokumentatsiooni.

Pythoni võimalused Blenderis

Kursus Sätted Osalejad Hinded Anuanded Rohkem ▾

Üldine

Teated õppijatele / News Forum FORUM

Lõpuprojekt Märgi tehtuks
OLESANNE

Avatud: neljapäev, 12. detsember 2024, 00:00 AM
Tähtaeg: neljapäev, 19. detsember 2024, 00:00 AM

Test 1: Teemad 2 ja 3 Märgi tehtuks

Test2: Teemad 4 ja 5 Märgi tehtuks

Test3: Teemad 6, 7 ja 8 Märgi tehtuks

1. Keskkonna seadist...
2. Blender pythoni s...
3. Objektide töötlemi...
4. Funktsioonide ja l...
5. Uue tööriista loom...
6. Python ja animatsi...
7. Pythoni skriptide e...
8. Lisamoodulid(add-...
- Kasulikud lingid

Pilt 2.2 Moodle kursuse vaade

Moodlesse sai loodud kaheksa moodulit. Lisaks on lisatud 3 testi, mis katavad erinevate moodulite teemasid. Lisatud sai ka lõpuülesanne, mis annab võimaluse loovalt rakendada kõiki läbitud teemasid. Moodulite piltideks on antud moodulis loodud materjali pilt või tulem.

2.2 Kursuse struktuur

Nagu eelnevalt sai mainitud, siis loodud e-õppekursus jaguneb kaheksasse moodulisse. Igas moodulid jagunevad veel omakorda alapeatükkideks. Iga mooduli juurde on lisatud pdf formaadis fail ning kui on olnud ka iseseisev töö, siis selle võimalikud lahendused on samuti välja toodud eraldi failidena.

Näide kuuenda mooduli sisust(Pilt 2.3).

The screenshot shows a Moodle course page for 'Pythoni võimalused Blenderis'. The page is titled '6. Python ja animatsioonid'. It lists five modules, each with a document icon, a title, a file type, and a 'Märgi tehtuks' button. The modules are: 1. 'Python ja animatsioonid' (PDF-dokument), 2. 'ruudustiku laine lahendus' (Tekstifail), 3. 'tähiobjekti jälgine kuus suumiga kood' (Tekstifail), 4. 'kuubiku värvi muutmise ja paneeli kood' (FAIL), and 5. 'Kuubirvi kood, kui kaamera liigub ringselt' (Tekstifail). The page also shows a navigation menu at the top with 'Kursus', 'Sätted', 'Osalejad', 'Hinded', 'Aruanded', and 'Rohkem'. A breadcrumb trail at the bottom shows '5. Uue tööriista loomine pythoni abil' and '7. Pythoni skriptide eksportimine ja salvestamine'.

Pilt 2.3 Näide Moodles olevast moodulist

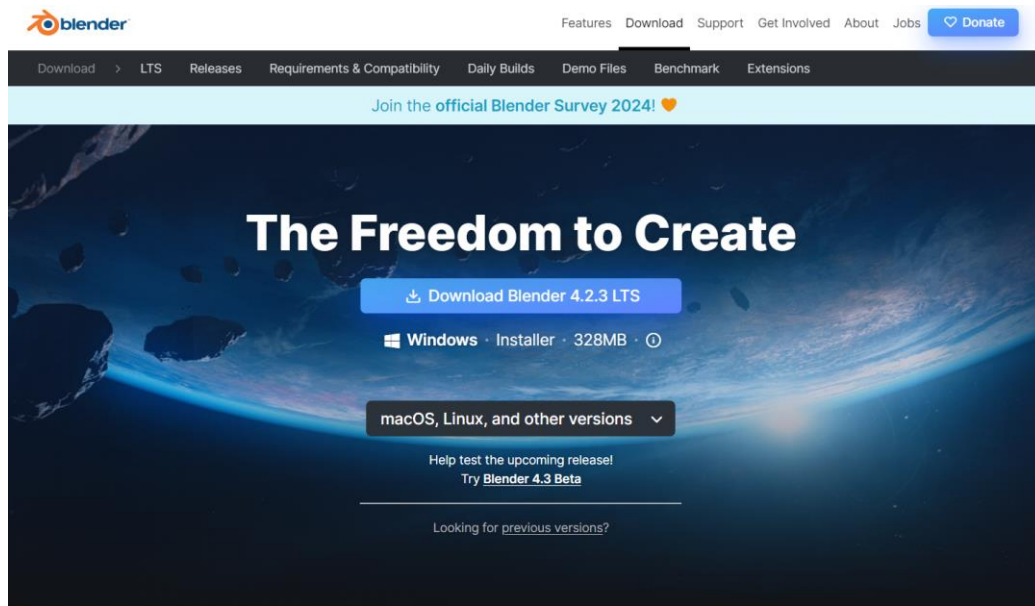
Siinkohal toob autor välja kõik kaheksa teemat ning kokkuvõtvalt nende õppetükkide peamise sisu.

2.2.1 Keskkonna seadistamine

Esimeseks teemaks valis autor keskkonna seadistamise. See teema on suuresti eelduseks selleks, et üldse saaks edasi kurust läbida. Üks olulisemaid asju selle teema

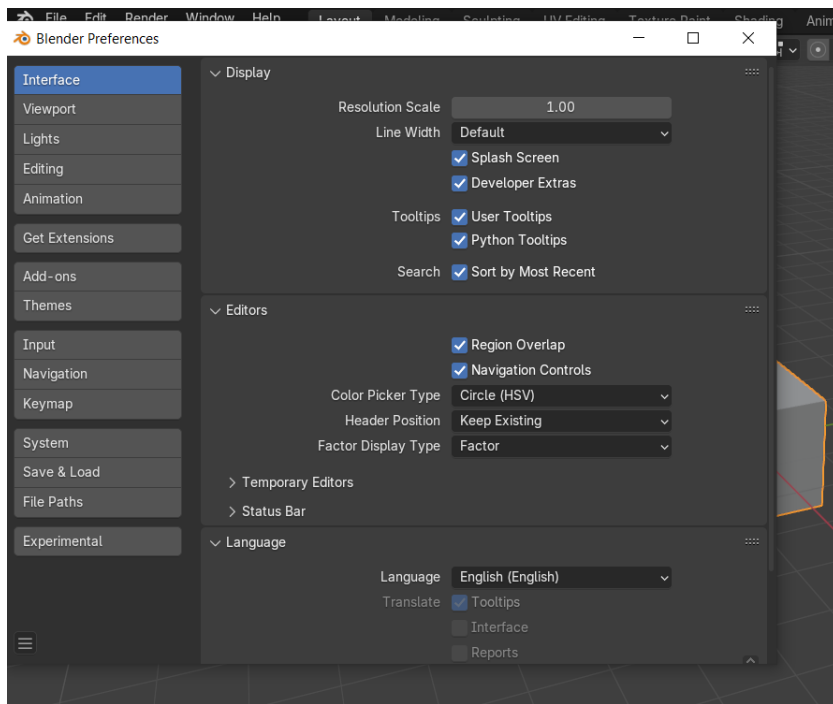
juures on välja tuua Blenderi versioon. Ka töö autor sattus paaril korral hätta sellega, et kasutatud materjalides oli kasutatud teist versiooni Blenderis ning kõik ettenähtud toiminguid ei toimunud ootuspäraselt ning tuli teha otsida dokumentatsioonist olemasolevale versioonile sobilik variant.

Seega peab arvestama, et see materjal on loodud kasutades Blenderis versiooni 4.2.3 (Pilt 2.4), vanematel versioonidel ei pruugi olla kõiki funktsionaalsusi, mida selle kursuse raames kasutatakse ning uuematel versioonidel võivad olla muutunud mõned käsud või täiustunud mõned moodulid.



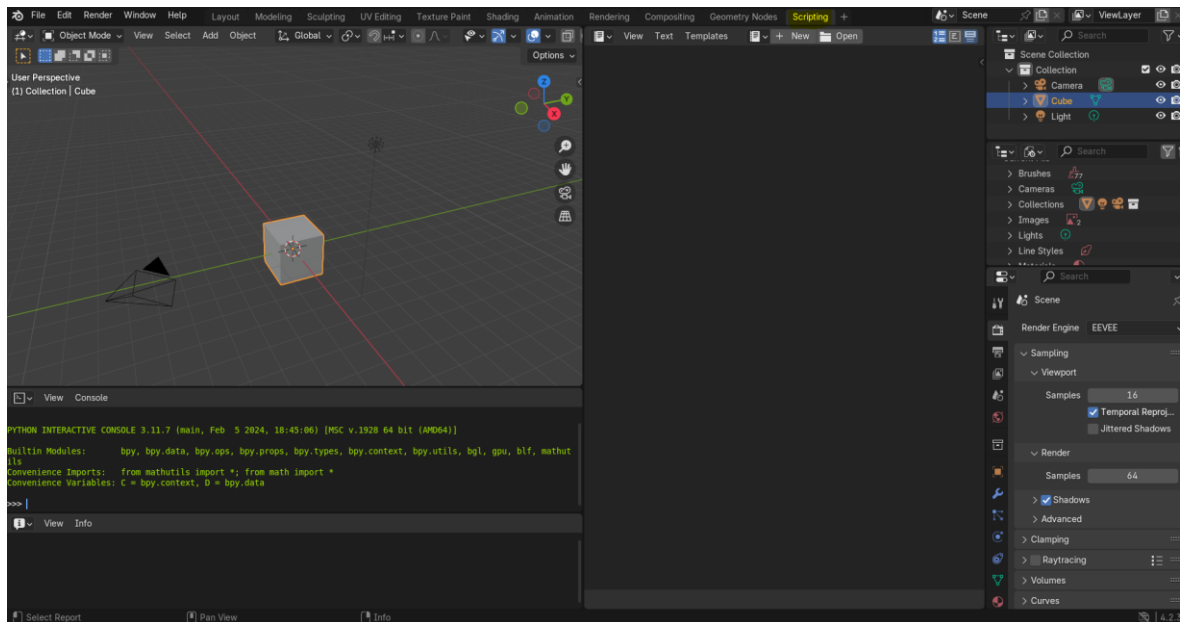
Pilt 2.4 Blenderi veebilehel allalaadimise koht

Teise osana selle õppetüki juures on välja toodud, kuidas ette valmistada Blender skriptide loomiseks, näiteks tuleb sisse lülitada Developer Extras ja Python tooltips (Pilt 2.5).



Pilt 2.5 Blender preferences valikute sisse lülitamine

Viimaseks on välja toodud vajalikud vaated, et oleks mugav kasutada pythoni koodi. Nendeks on text editor, pythoni konsool ja info editor (Pilt 2.6). Neid kõiki on natukene tutvustatud ja lisatud dokumentatsiooni viited.



Pilt 2.6 Blenderi vaade Pythoni kasutamiseks

2.2.2 Blender pythoni skriptimise põhialused

Selle peatüki eesmärgiks on õppijale meelde tuletada või õpetada pythoni kasutamise alustõdesid ja tutvustada bpy moodulit. See moodul ei lähe süvitsi vaid peamiselt annab

lihtsalt ülevaate mõne kerge harjutusega.

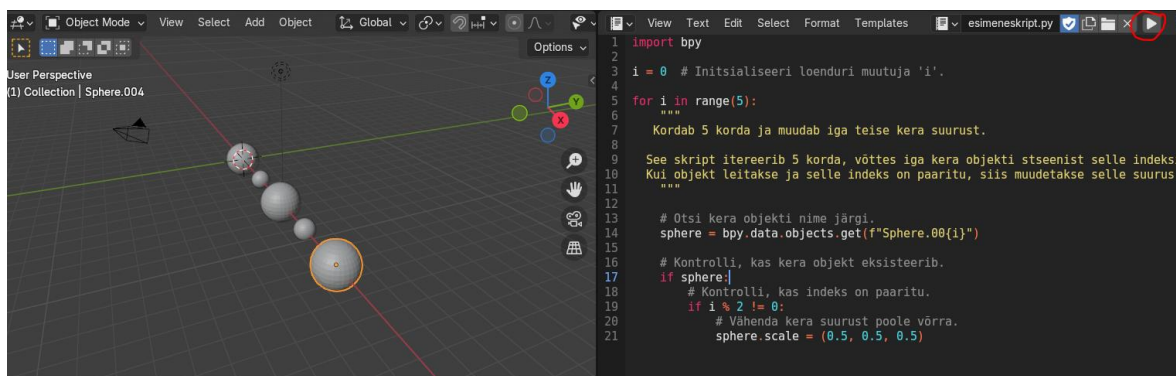
Pythoni skripte tuleb kirjutada pythoni konsooli (Pilt 2.7). Katsetada saab muutujate defineerimist ja erinevaid andmetüüpe. Lisaks meenutatakse nii while kui ka for tsükleid ning viimaks saab katsetada ka funktsiooni kirjutamist ja välja kutsumist.

```
>>> def regist(name, course="Blender"):  
...     print(f"{name}, olete registreerinud kususele {course}")  
...  
>>> regist("Mari")  
Mari, olete registreerinud kususele Blender  
  
>>> regist("Mari","Python")  
Mari, olete registreerinud kususele Python  
  
>>> |
```

Pilt 2.7 Funktsiooni kirjutamine ja välja kutsumine konsoolis

Bpy mooduli abi saab Blenderis juhtida põhimõtteliselt kõike. Selles osas tehakse ülevaade, kuidas käivitada operaatoreid, kuidas pääseda ligi objekti andmetele ja kuidas neid mõjutada. Näidatud on ka see, kuidas tsükli abi luua mitu objekti ja kuidas muuta ka objektide suurust.

Viimaseks õpetatakse õpilast kasutama text editorit (Pilt 2.8). Kuidas salvestada skripte ja kuidas saab koodi käivitada läbi text editorit.



Pilt 2.8 Koodi näidis text editoris

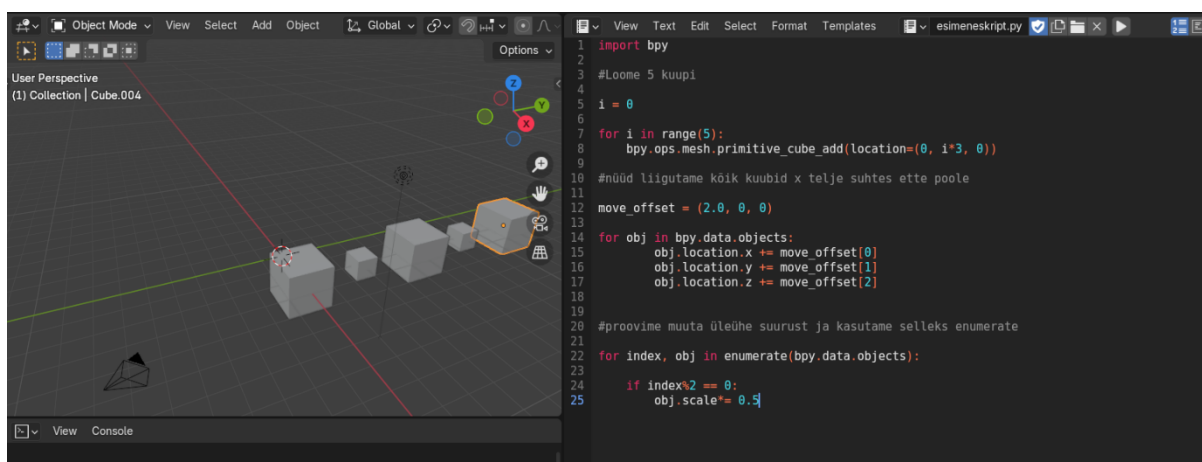
Kui bpy mooduli osa alguses sai tehtud kõik ülesanded konsoolis, siis peavad õpilased peatüki lõpus iseseisvalt kõik koodid kirjutama ka text editoris failina, et kinnistada muuhulgas ka see, et iga pythoni faili alguses peab olema imporditud bpy moodul.

2.2.3 Objektide töötlemine Pythoniga

Kui eelmises peatükis anti pealiskaudne ülevaade, siis kolmandas peatükis on eesmärgiks põhjalikumalt näidata objektide töötlemise võimalusi ja õppija saab ka kolm ülesannet, mille peab proovima iseseisvalt ära lahendada.

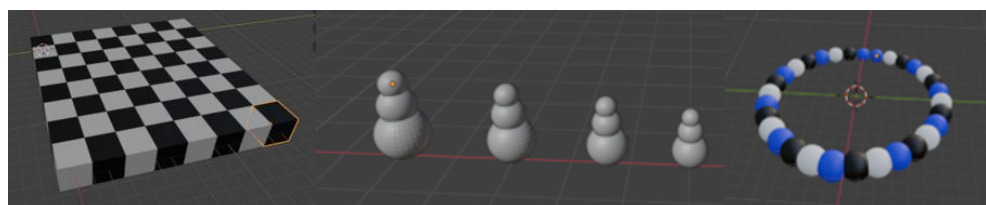
Peatükk on jagatud kolmeks. Esimene osa kajastab objektidele ligipääsemist, kus tuuakse välja, mis on erinevus bpy.data ja bpy.context-i vahe. Lisaks on näidatud, kuidas enda soovitud objekti muuta aktiivseks objektiks. See on oluline, sest see vähendab võimalust, et koodist saaks mõjutatud mõni vale objekt.

Teise osana on välja toodud objektide omadused ja teisendused. Kuidas saab näiteks objekti asukohta muuta, kuidas saab suurust muuta või kuidas saab pöörata kolme telje suhtes. Selles osas on ka välja toodud võimalus, kuidas objekti peita renderdamise eest, mis aitab suurte projektide puhul võita renderdamise aega. Lisaks on välja toodud ka võimalus, kuidas mõjutada mitut objekti korraga (Pilt 2.9). See on oluline, kui soovida sama või sarnast tegevust rakendada mitmele objektile.



Pilt 2.9 näidis funktsioon objektide töötlemise kohta

Peatüki viimases osas on õppijatele antud kolm ülesannet, kus saab rakendada eelnevalt läbi räägitut. Lisaks peab uurima ka natukene dokumentatsiooni, et leida võimalused värvi lisamiseks, millest tuleb alles edaspidi juttu. Ülesanded lahendamiseks nagid ette maleruudustiku loomist ja värvimist, lumememmede tegemist ning kaelakee tegemist ning ka selle värvimist (Pilt 2.10).



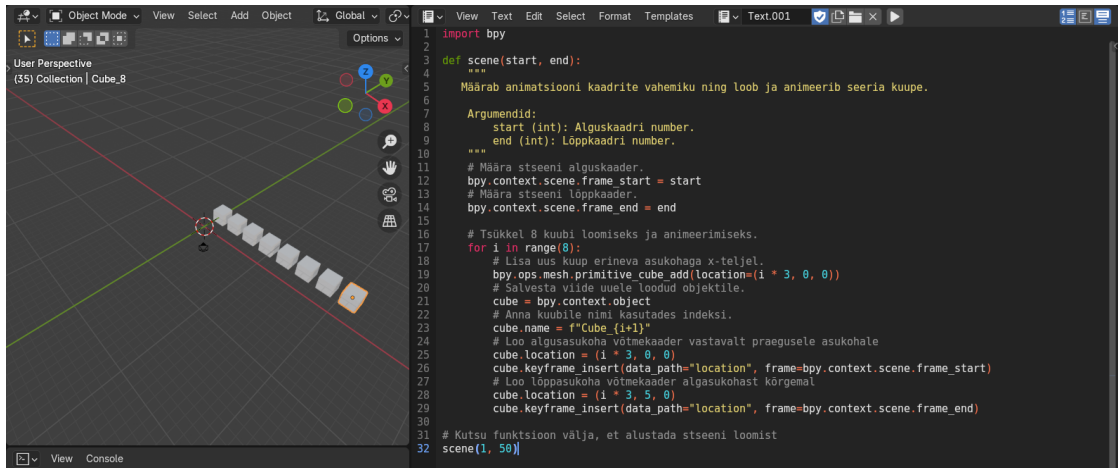
Pilt 2.10 Kolm iseseisva töö väljundit

2.2.4 Funktsioonide ja tsüklite abil automatiseerimine

Neljas peatükk jaguneb samuti kolmeks suuremaks teemaks, milleks on dünaamiline loomine ja manipuleerimine, korduvkasutatav automatiseerimise raamistik ning kogud

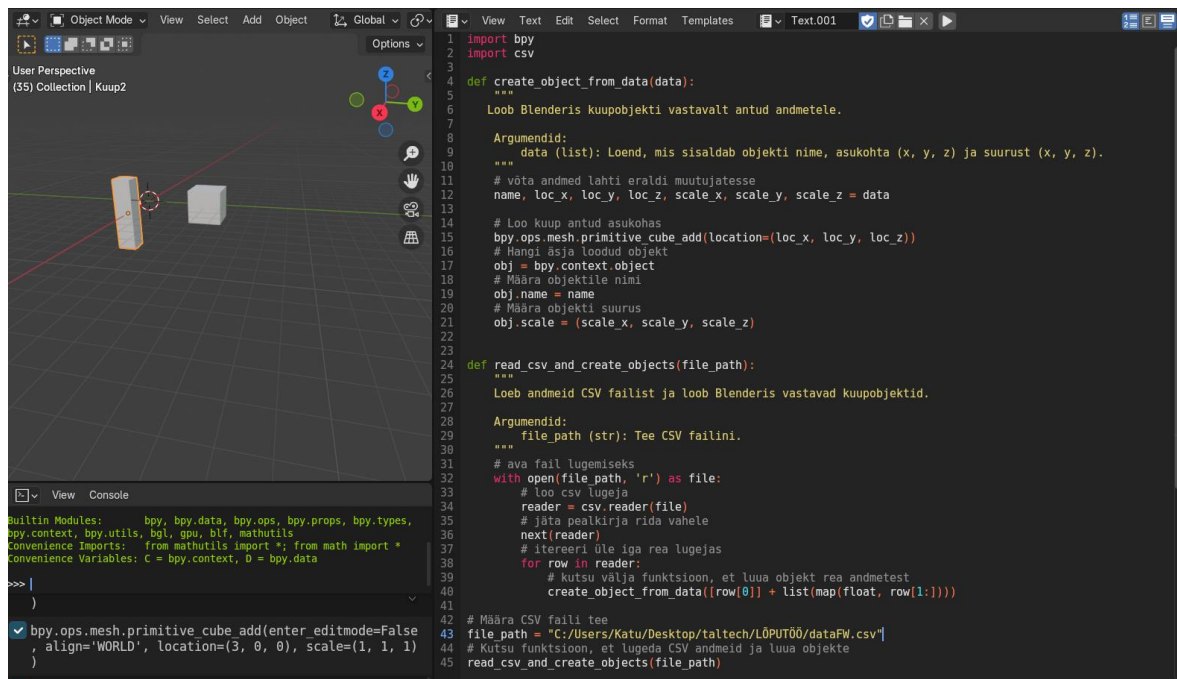
ja kihid. Selles osas läheb ka koodi kirjutamine natukene keerulisemaks ja võib vajada tuge pythoni dokumentatsioonist.

Esimeses osas on juttu hierarhiliste struktuuride loomist, mis see tähendab ja kuidas sõltuvus üksteisest mõjutab objekti omaduste muutmist. Lisaks kuidas automaatselt objekti omaduste põhjal määrata objekti materjal ning funktsioonide kasutamine (Pilt 2.11) animeerimisel.



Pilt 2.11 Näide tsükli kasutamisest funktsionis

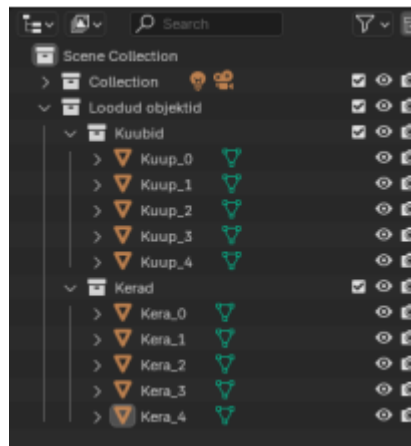
Järgmises osas saab õppija luua objektihalduse raamistikku, ehk luua pythonis klassi, mille abil saab objekte luua, muuta ja hallata. Samuti käsitletakse seda, kuidas csv failist infot lugeda ja seda kasutada (Pilt 2.12).



Pilt 2.12 csv failist info lugemine

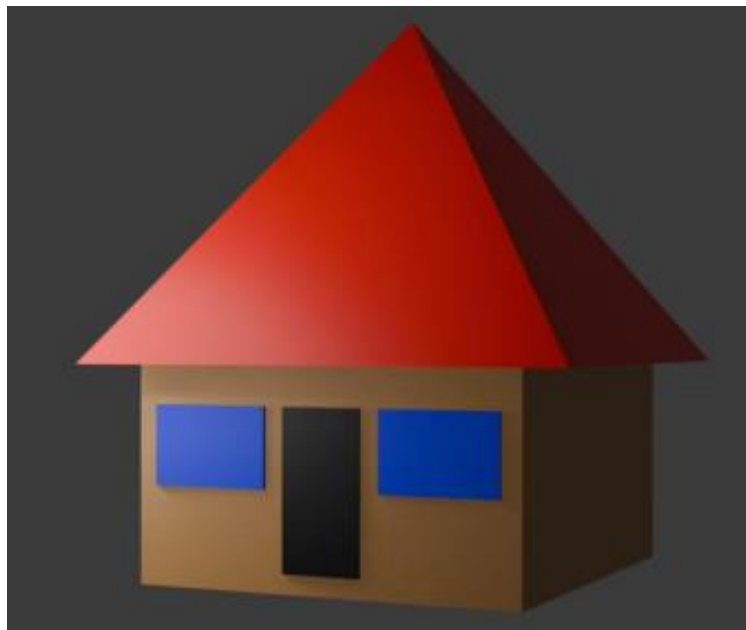
Kolmandas osas on juttu kogude haldamisest ja kihilisest automatiseerimisest

Näiteks kuidas teha erinevaid kollektsioonide kihte (Pilt 2.13).



Pilt 2.13 kollektsioonide kihtide näide

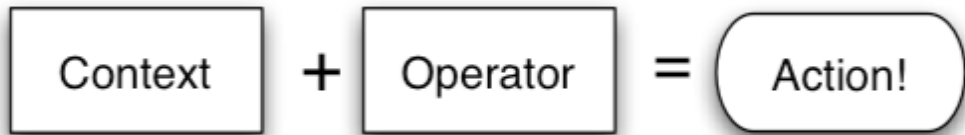
Peatüki lõpus on ülesanne, kus tuleb eelnevalt õpitud oskusi kasutada ja luua maja kujutis (Pilt 2.14) ette antud juhiste järgi.



Pilt 2.14 Maja loomise ülesande võimalik tulem

2.2.5 Uue tööriista loomine pythoni abil

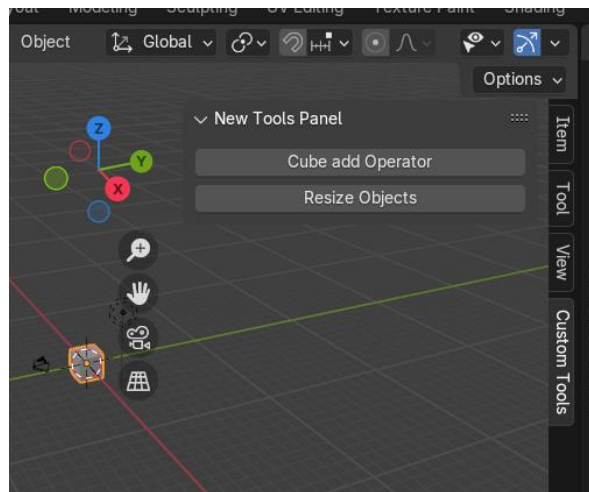
Uue tööriista loomise peatükis keskendutakse peamiselt kolmele punktile. Esimeseks punktiks seletatakse milline on Blenderi operaatorite süsteem (Pilt 2.15) ja miks see on oluline ja kuidas luua uus operaator.



Pilt 2.15 skeem operaatorite süsteemi kohta

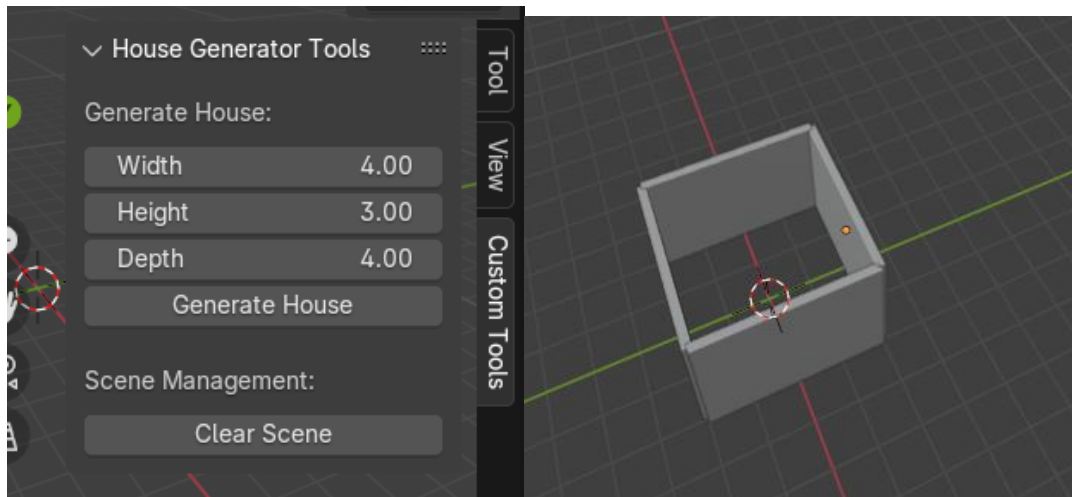
Teises osas vaadatakse, kuidas kohandatud operaatorit täiendada ja omadusi lisada. Õppija saab loodud funktsiooni välja kutsuda pythoni konsoolis ja vaadata kuidas kaasa antud parameetrit muutes muutub ka tulemus mida näeme kasutaja vaates.

Kui teises osas oli vaja funktsioon välja kutsuda konsoolist, siis kolmandas osas vaadatakse üle võimalused, kuidas lisada enda loodud tööriistad kasutajaliidesesse 3D vaatele (Pilt 2.16).



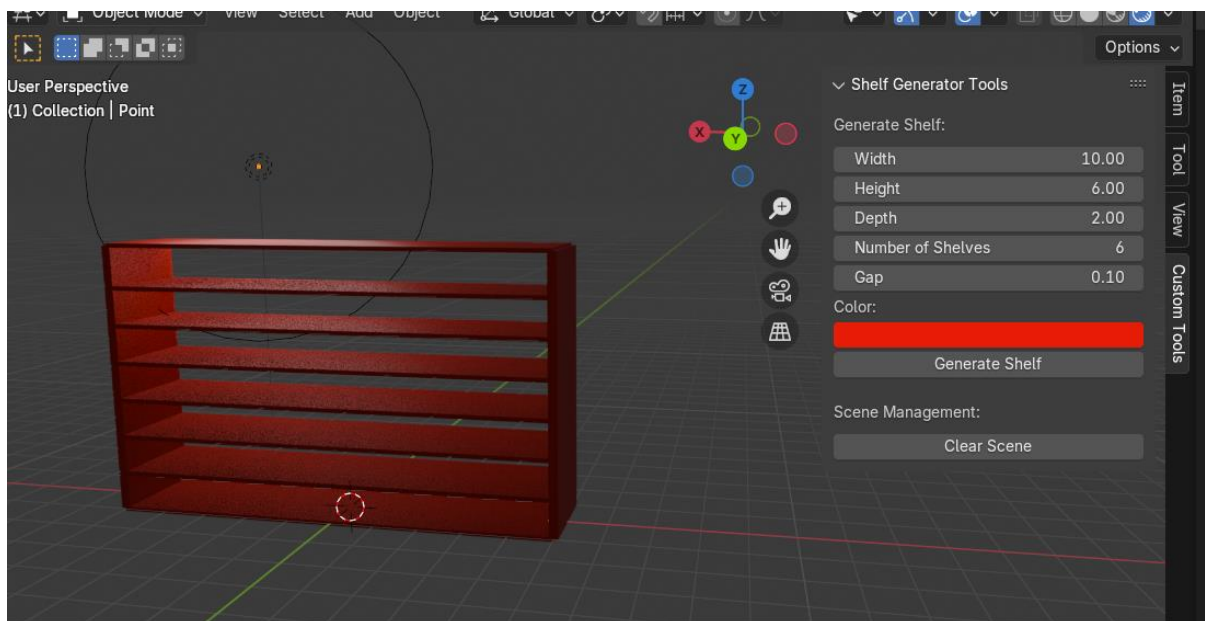
Pilt 2.16 Näide tööriista lisamisest paneelile

Selles peatükis arendatakse natukene edasi ka maja tegemise ülesannet, mis sai loodud eelmises peatükis. Juhendis on ära näidatud, kuidas luua majaseinte loomise jaoks uus töörist, kuidas see lisada kasutajaliidese paneeli. Oluline oli see, et kasutaja peaks saama määrata läbi kasutajaliidese seinte suurusi (Pilt 2.17). Õppijatele jäätakse võimalus ka iseseisvalt lisada sinna juurde ka katuse ja miks mitte ka akende loomise osa.



pilt 2.17 Näide majaseinte tööriista loomise tulemist

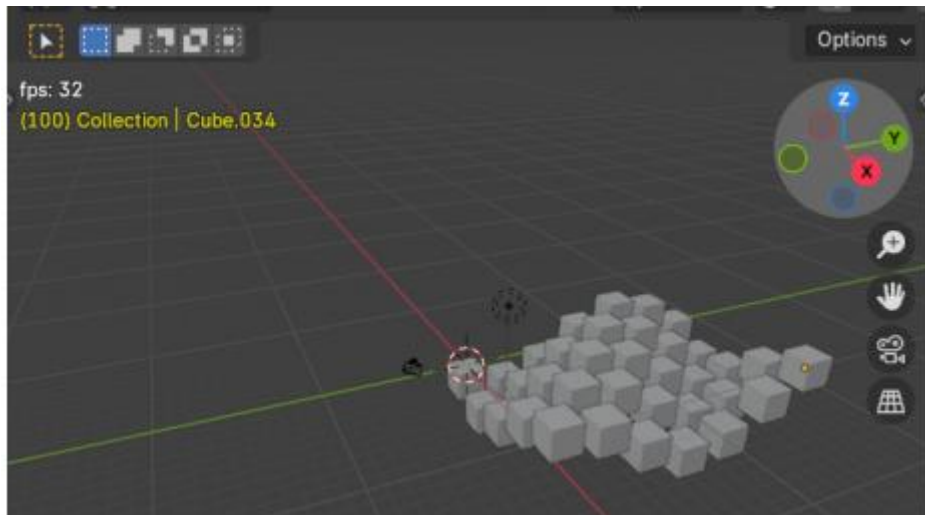
Kui majaseinte tegemise osa on juhendis ette antud, siis iseseisvaks harjutuseks tuli luua riuli tegemise operaator (Pilt 2.18) ja see lisada paneelile. Kasutaja pidi saama valida riuli suurust ja sahtlite arvu. Samuti ka muuta riuli värvi. See ülesanne on üsna keeruline ja vajab natuke suuremat pusimist ja ka lisamaterjali otsimist. Riuli ülesande lahendamine osutus ka üpris keeruliseks töö autorile, kuid lõpuks valminud võimalik lahendus on lisatud samuti Moodle keskkonda.



Pilt 2.18 Võimalik tulem riuli loomise ülesandele

2.2.6 Python ja animatsioonid

Animatsioonide osa on jagatud viieks alateemaks. Kõigepealt vaadatakse üle kuidas määrata pythoniga võtmekaadreid. Kuidas nende abil näiteks panna ruudustik lainetama (Pilt 2.19).



Pilt 2.19 lainetava ruudustiku tulem

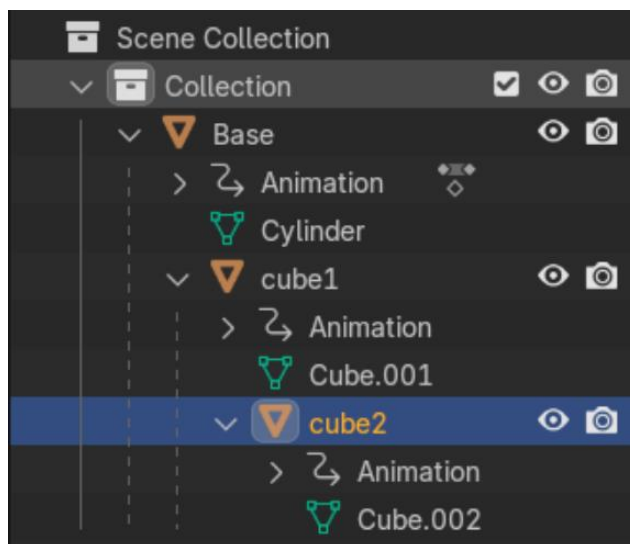
Teiseks vaadatakse, kuidas saab kaameraanimatsioone skriptida. Kirjutatakse koodi, kus kaamera jälgib tühiobjekti. Hiljem täiendatakse seda koodi, et kaamera suumiks vahepeal sisse ja muudaks ka fookust (Pilt 2.20).

```
# Sisse- ja väljasuunimine
camera.data.lens = 50
camera.data.keyframe_insert(data_path="lens", frame=1)
camera.data.lens = 200 # sissesuunimine
camera.data.keyframe_insert(data_path="lens", frame=frame_count // 2)
camera.data.lens = 20 # väljasuunimine
camera.data.keyframe_insert(data_path="lens", frame=frame_count)

# lubame sügavuse ja muudame fookuskaugust
camera.data.dof.use_dof = True
camera.data.dof.focus_distance = 20
camera.data.keyframe_insert(data_path="dof.focus_distance", frame=1)
camera.data.dof.focus_distance = 5 # fookus lähemale
camera.data.keyframe_insert(data_path="dof.focus_distance", frame=frame_count // 3)
camera.data.dof.focus_distance = 20 # tagasi algesse
camera.data.keyframe_insert(data_path="dof.focus_distance", frame=frame_count)
```

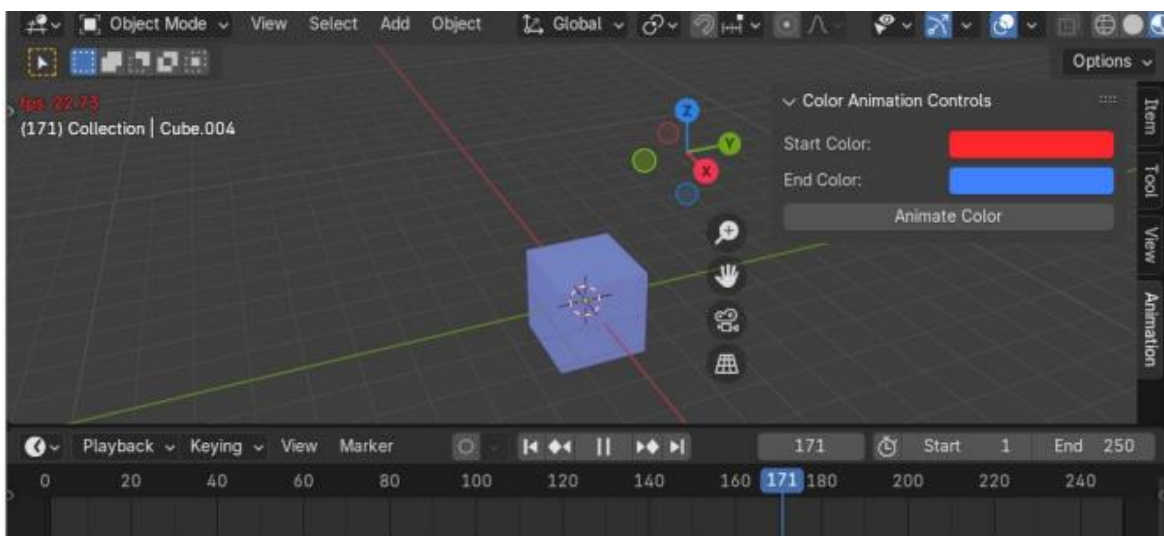
Pilt 2.20 näidis kaamera liikumise kohta

Kolmandas osas vaadatakse, kuidas luua hierarhilisi animatsioone. Uuritakse, kuidas laps-vanem seosed mõjutavad erinevate objektide liikumist(Pilt 2.21).



Pilt 2.21 hierarhia ja liikumise kombinatsioon

Kahes viimases osas luuakse materjalide animatsioone ja selle juhtimiseks ka nupud. Loodud sai kood, mis loob paneelile uue tööriista, mis võimaldab kasutajal valida kuubiku alguse värvi ja lõpuvärvi ning värv peab sujuvalt muutuma ühest teiseks (Pilt 2.22).



Pilt 2.22 Kuubiku värvimise ülesande võimalik tulem

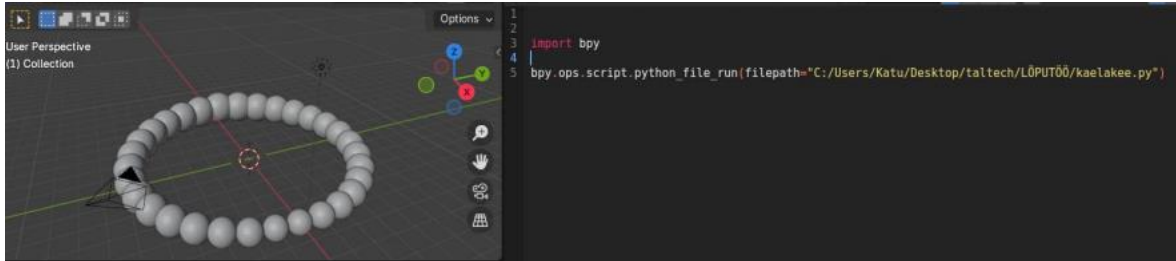
Lõpuks said õpilased ka ülesande, kus pidi kasutama kolme esimest punkti. Luua tuli robotkäsi (kasutades hierarhiat), mis koosneb mitmest osast ning lisada ka kaamera, mis järgib käe liikumist, huvitavamaks muutmiseks võis ka lisada suumimist ja ka materjali muutmist.

2.2.7 Pythoni skriptide eksportimine ja salvestamine

Selles osas keskendutakse Blenderis pythoni skriptide haldamise, eksportimise ja salvestamise praktilistele aspektidele ning nende skriptide integreerimisele erinevatesse

töövoogudesse.

Ühe osana vaadake üle, kuidas väliseid skripte saab käivitada `bpy.ops.script.python_file_run()` käsuga. Proovitakse laadida mõni varem koostatud skript sellisel moel ka oma programmi (Pilt 2.23).



Pilt 2.23 Failist koodi käivitamise näidis

Blenderis on võimalus animatsioone salvestada nii json kui ka csv failina. Seda sai katsetatud tühiobjekti jälgimise ülesandes.

Näiteks json faili salvestamine (Pilt 2.24):

```

def export_object_animation_to_json(obj, filepath):
    """
    Eksportib antud objekti animatsiooni andmed JSON faili.

    Argumendid:
    | obj (bpy.types.Object): Blenderi objekt, mille animatsiooni andmeid eksporditakse.
    | filepath (str): JSON faili tee, kuhu animatsiooni andmed salvestatakse.
    """
    # Kontrolli, kas objektil on animatsiooni andmed ja seostatud tegevus.
    if not obj.animation_data or not obj.animation_data.action:
        print(f"Object '{obj.name}' has no animation data.")
        return

    # Initsialiseeri tühi sõnastik animatsiooni andmete salvestamiseks.
    animation_data = {}

    # Juurdepääs objekti animatsioonitegevusele.
    action = obj.animation_data.action

    # Itereeri üle tegevuse iga F-kurvi (animatsioonikõvera).
    for fcurve in action.fcurves:
        # Hangi F-kurvi andmepunkt (nt. asukoht, pööramine).
        data_path = fcurve.data_path
        # Hangi F-kurvi massiivi indeks (nt. x, y, z asukoha jaoks).
        array_index = fcurve.array_index

        # Koosta string omaduse nimega, näiteks "location[0]", location[1]...
        property_name = f"{data_path}[{array_index}]"
        # Initsialiseeri tühi loend võtmekaadrite salvestamiseks
        keyframes = []

        # Itereeri üle F-kurvi iga võtmekaadripunkti.
        for keyframe in fcurve.keyframe_points:
            # Lisa võtmekaadrit esindav sõnastik võtmekaadrite loendisse.
            keyframes.append({
                "frame": keyframe.co[0], # kaadri number
                "value": keyframe.co[1], # selle kaadri väärtus
            })

        # Salvesta võtmekaadri andmed vastava omaduse jaoks animatsiooni sõnastikku.
        animation_data[property_name] = keyframes

    # Salvesta animatsiooni andmed JSON faili.
    with open(filepath, 'w') as file:
        json.dump(animation_data, file, indent=4) # Taane (indent) paremaks vorminduseks

    print(f"Animation data for '{obj.name}' exported to {filepath}")
    # Väljasta info animatsiooni ekspordi kohta konsooli

# Näide kasutamisest (asenda oma soovitud objekti ja faili teega)
export_object_animation_to_json(
    bpy.data.objects['Empty'],
    "C:/Users/Katu/Desktop/taltech/LÖPUTÖÖ/empty_animation.json"
)

```

pilt 2.24 Näidis liikumise salvestamiseks .json faili

Mille tulemusel tekkis ilus .json formaadis fail (Pilt 2.25).

```
1  "location[0]": [  
2  [  
3  {  
4    "frame": 1.0,  
5    "value": 0.0  
6  }],  
7  [  
8    "frame": 300.0,  
9    "value": 0.0  
10 }]  
11 ],  
12 "location[1]": [  
13 [  
14 {  
15   "frame": 1.0,  
16   "value": 0.0  
17 }],  
18 [  
19   "frame": 300.0,  
20   "value": 30.0  
21 }]  
22 ],  
23 "location[2]": [  
24 [  
25 {  
26   "frame": 1.0,  
27   "value": 0.0  
28 }],  
29 [  
30   "frame": 300.0,  
31   "value": 0.0  
32 }]  
33 ]
```

Pilt 2.25 .json faili näidis, kuhu on salvestatud liikumine

Lisaks sai vaadatud, kuidas laadida ja hallata erinevaid väliseid teeke (Libraries) ning kuidas korraldada versioonikontrolli ja korraldada koostööd. Lisaks on välja toodud ka erinevad olulised punktid koodi kirjutamise, nagu näiteks dokumenteerimise ja kommenteerimise vajadus.

Õppijad said ülesandeks lisada oma eelmistest osadest pärit vabalt valitud kood giti. Selleks anti soovitus jälgida arendaja juhendit ning kasutada enimlevinuid keskkondasi näiteks githubi või gitlabi

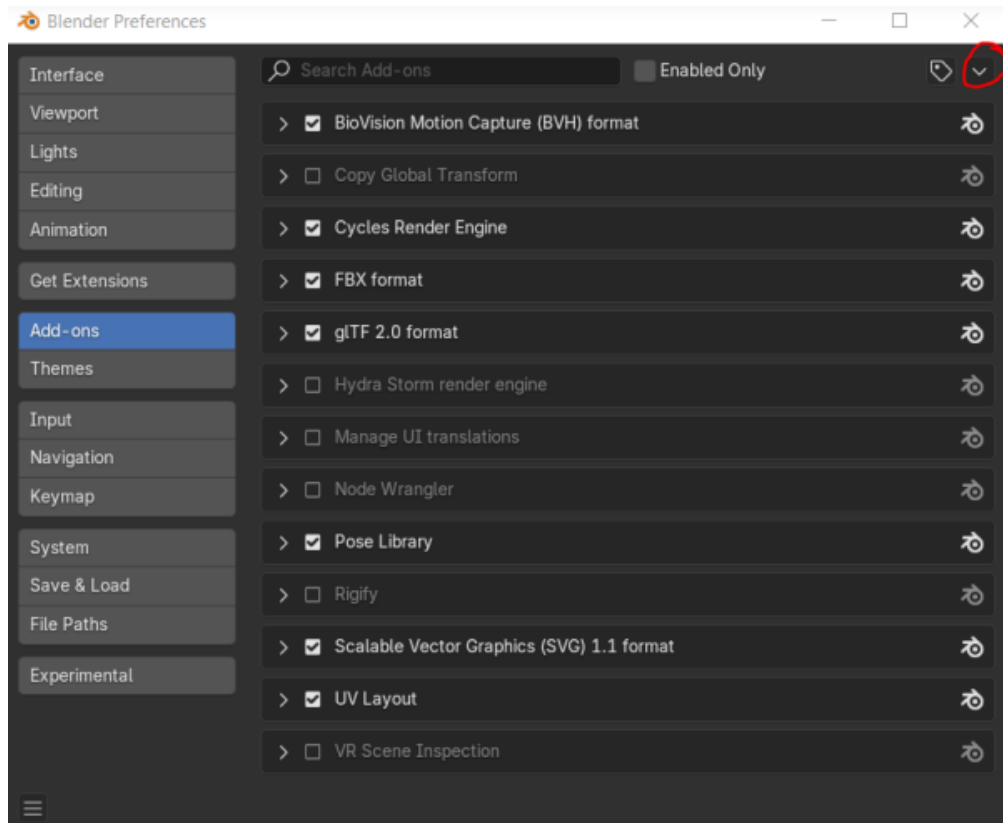
2.2.8 Lisamoodulid(add-ons)

Kõige viimases, ehk kaheksandas peatükis räägitakse Blenderi lisamoodulitest. Kõigepealt on tutvustatud Blenderi lisamoodulite loogikat (Pilt 2.26) ja kasutusvaldkondasid. Lisamooduli ehitus on väga oluline just selleks, et saaks ise luua lisamooduleid.

```
bl_info = {  
    "name": "My Test Add-on",  
    "blender": (2, 80, 0),  
    "category": "Object",  
}  
  
def register():  
    print("Hello World")  
  
def unregister():  
    print("Goodbye World")
```

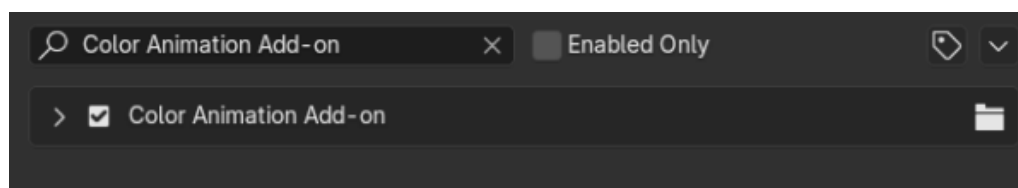
Pilt 2.26 Lisamooduli struktuur

Selles peatükis enam päris uut koodi kirjutama ei hakatud, vaid võetakse ette eelnevalt juba eelnevas peatükis loodud kuubi värvimuutmise kood, koos UI paneeli osaga. Seda koodi muudetakse nii, et sellest saaks Blenderi lisamoodul ja vastaks kõikidele nõuetele. Kui lisamoodul sai loodud, siis lisatakse see ka lisamoodulina endale, et näha, kuidas kohandatud lisamooduli lisamine ka praktikas välja näeb(Pilt 2.27).



Pilt 2.27 Lisamooduli lisamine aken

Katsetatakse ka seda, kui Blender taaskäivitada, kas siis lisatud lisamoodul on alles ja endiselt töökorras(Pilt 2.8).



Pilt 2.28 Lisatud lisamooduli näidis

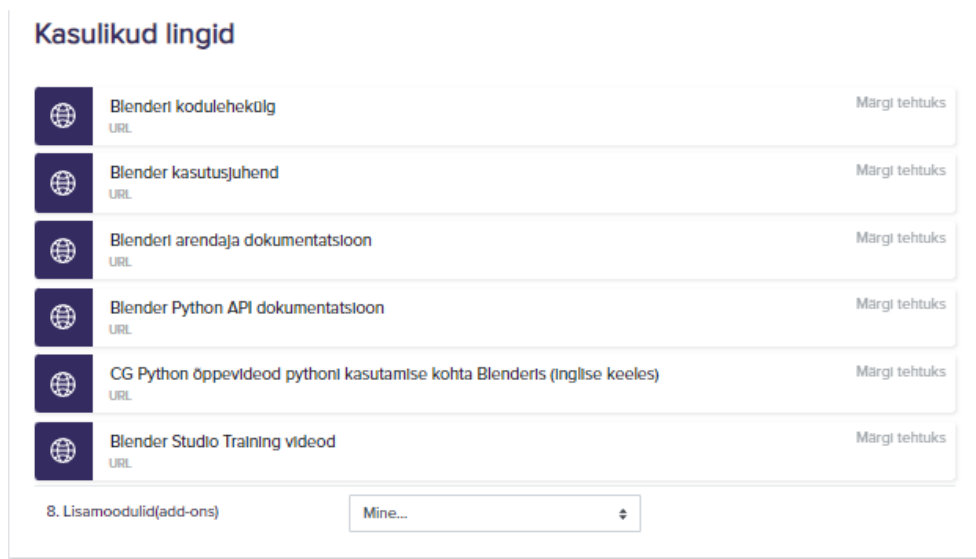
Viimase osana selles õppetükis räägitakse lisamooduli kokku pakkimise võimalustest ja jagamisest. Oluline on kaasa panna ka README fail, kuhu lisatakse info selle kohta, mida lisamoodul teeb, kuidas seda installida ja kasutada ning kindlasti ka Blenderi versiooni number, millega see on varasemalt töötanud(või millega see on loodud).

Ka selle peatüki lõpus antakse õppijatele ülesanne, milleks on majaseinte loomise ülesandest teha lisamoodul. Selleks tuleb muuta koodi. Kui see osa saab valmis, tuleb

kood kokku pakkida, lisada README fail ja üleslaadida git keskkonda.

2.2.9 Kasulikud lingid

Kõige viimases Moodle moodulisse on pandud kirja erinevad kasulikud lingid (Pilt 2.29), kust on töö autor infot saanud ja kust võiks saada ka infot õppija.



Pilt 2.29 Moodles olevad kasulikud lingid

2.3 Moodlesse loodud testid

Moodle keskkonda sai lisatud kolm testi. Need testid annavad hinnangu Blender pythoni skriptide loomise kohta, sest hõlmavad endas kõiki läbitud teemasid, alustades pythoni programmeerimiskeelest lõpetades animatsiooni ja kasutajaliidese loomisega.

Küsimusi on üldiselt kolme varianti, kas valikvastustega, tõene/väär või tühja koha täitmine. Valikvastusega küsimused testivad peamiselt pythoni süntaksi ja Blender API käskude teadmisi, lisaks ka natukene konkreetsemaid teadmisi näiteks bpy mooduli kohta. Tõene/väär küsimused keskenduvad sellele, et õppija mõistaks skriptimisega seotud põhikontseptsioone ja funktsionaalsusi. Viimaks lühivastused, need on ilmselt kõige keerulisemad küsimused, sest need eeldavad Blenderi API ja süntaksi natukene põhjalikumalt mõistmist. Seal on vaja lõpetada koodi või vastata juba konkreetsete terminitega.

Testid on jagatud kolme gruppi (Pilt 2.30). Esimene test kajastab endas teist ja kolmandat teemat ning kokku on küsimusi 10. Teine test sisaldab teemasid 4 ja 5, kuna need teemad olid natuke pikemad, siis selles testis on kokku 15 küsimust. Viimases testis, mis sisaldab endas viimase kolme teema küsimusi, on kokku 17 küsimust.



Pilt 2.30 Moodlesse lisatud testide vaade

Üldiselt on nende testide eesmärgiks hinnata teoreetilisi teadmisi aga ka pythoni praktilist rakendamist Blenderi keskkonnas, mis võimaldab õppijatel pythonit tõhusalt kasutada, et laiendada Blenderi võimalusi oma projektide jaoks. Need testid ei ole mõeldud hinnete andmiseks vaid õppijale enesetestimiseks, kuid vajadusel ja soovil saab kursuse korraldaja nende eest ka hindeid panna.

Viimasena on Moodlesse lisatud ka lõpuprojekti ülesanne (Pilt 2.31), mis kursuse lõpetamiseks tuleb esitada. Selles lõpuprojektis on kajastatud suurem osa läbivõetud teemasid.

Loo Blenderi projekti, kus kasutada ära võimalikult palju õpitud elemente. Lähene ülesandele loovalt ja leia ise juurde mõned uued nipid, mida saaks veel pythoniga kasutada.

Punktid, mis peavad olema kajastatud

- objekti töötlemine `bpy.data` abil
- automatiseeri tsükli abil
- loo funktsioone, et neid saaks korduvalt kasutada
- loo uus tööriist ja loo ka kasutajaliides
- loo hierarhia ja animatsioon
- tekita ise üks lisamoodul

Ideena võib edasi arendada näiteks maja, luua sinna ümber ka maastik ning näiteks tuulelipp, mis liigub. Kuid kui kõik kriteeriumid on täidetud, siis ei ole piirangut, mida töö endas kujutama peab.

Pilt 2.31 Moodlesse lisatud lõpuülesanne

KOKKUVÕTE

Pythoni kasutamine Blenderis võib keerukate objektide loomist oluliselt lihtsustada. Loodud kursus keskendus 3D-modelleerimisele ja tutvustas natukene ka animatsiooni kontseptsioone, kuid Pythoni skriptimise paindlikkus ja võimsus Blenderis muudavad selle kasulikuks mitmesuguste tegevuste jaoks, näiteks mängude ja videoklippide arendamiseks. Autori töötas välja metoodilise lähenemise, mis ühendab endas teoreetilised teadmised praktiliste ülesannetega, tagades õppijate aktiivse osaluse. Autor mõtles välja ja valis materjalidesse selliseid ülesandeid ja näiteid, mis toetaks koheselt teoreetilist osa.

Lõputöö eesmärk, luua e-kursus, mis tutvustaks Blenderi ja Pythoni kooskasutamise võimalusi väheste kogemustega inimestele, sai täidetud, sest kursuse läbinud õppijad suudab iseseisvalt luua ja manipuleerida 3D-objekte Pythoni skriptide abil. Kuigi kursus keskendus 3D-modelleerimisele, selgus töö käigus, et Pythoni skriptimine Blenderis avab palju võimalusi ka muudes valdkondades, nagu mängude arendus ja animatsioonide loomine. Moodle'i keskkonna intuiitivne kasutamine ja laialdased võimalused muudavad selle sobivaks interaktiivseks õppematerjaliks.

Kursuse loomine kinnitas, et hoolimata esialgsetest raskustest on võimalik luua algajasõbralik õppematerjal, mis aitab neil omandada Pythoni skriptimise oskused Blenderis. Kursuse peamiseks tugevuseks on selle praktiline lähenemine, mis julgustab õppijaid kohe teadmisi rakendama. Samuti osutus Moodle'i keskkonnas pakutavad harjutused ja testid tõhusaks viisiks õppimise kinnistamiseks. Arenguruumi on kursuse mitmekesistamisel, lisades rohkem interaktiivseid elemente ja kaasaegseid õppevorme. Samuti võiks kursust täiendada keerukamate ülesannetega, et rahuldada edasijõudnute vajadusi. Tulevikus tasuks kaaluda ka õpilaste tagasiside kogumist, et parandada õppematerjalide kvaliteeti ja struktuuri. Edasiste uuringute ja arendustöö raames võiks luua põhjalikuma õppekava, mis hõlmab kõiki Blenderi Python API aspekte.

Autor tunnustab, et töö piiranguks osutus ka ajapuudus, mis ei võimaldanud minna nii süvitsi, kui esialgu plaanis. Lisaks peaks looma testgrupi, kes katsetaks kursust ja annaks autorile tagasiside selle parendamisvõimaluste kohta. Seda tööd luues üllatas autorit, kui palju võimalusi pakub Blenderi Python API ja kuidas selle abil saab automatiseerida keerukaid 3D-modelleerimise ülesandeid. Loodud e-kursus aitab populariseerida Blenderit ja Pythonit laiemalt, andes algajatele juurdepääsu nende võimsate tööriistade kasutamisele ning panustab avatud lähtekoodiga haridusressursside loomisesse. Tulevikus sooviksin luua veelgi põhjalikuma õppekava, mis hõlmab kõiki Blenderi Python API aspekte. Samuti on autori sooviks arendada ka interaktiivset õppeplatvormi, kus oleks võimalik paremini õppijatelt tagasisidet saada.

SUMMARY

The title of this thesis is „Python capabilities in Blender. Educational materials and course development“ and the author is Katri Laanisto. This thesis deals with the growing importance of 3D modeling in various fields, including construction, design, art, architecture and manufacturing. There is an increasing need for 3D models created for both professional and hobbyist use. This has led to the need to automate complex modeling tasks and create customized tools. Blender, a well-known free 3D modeling software, offers extensive functionality and at the same time an underrated capability thanks to its Python API. This thesis focuses on Blender's Python API, because there are no educational materials in Estonian for using Python and Blender together, especially in a beginner-friendly approach. The main goal of the thesis was to create an e-course aimed at people with low skills through practical tasks, which would introduce the use of Python in Blender. The object of research was Blender's Python API and its application possibilities in 3D modeling and animation, and the e-course was chosen as the format for creating the educational material.

To achieve the goals of the work, three main tasks were set: 1) to thoroughly study the Blender Python API, focusing on important functions for automating the creation and manipulation of 3D models; 2) to create interactive learning materials in the Moodle environment that combine the theoretical base with practical examples and exercises; and 3) to create tasks and tests of varying difficulty so that learners can assess their knowledge and skills. The created course, which focuses on a practical approach to using Python in Blender, integrating them into an interactive Moodle environment.

The final thesis resulted in an e-course that introduces beginners to the possibilities of using Python and Blender together. Learners who have completed the course should be able to independently create and manipulate simpler 3D objects using Python scripts. The created Moodle course is intuitive and flexible, offering learners the opportunity to learn at a suitable pace. The work confirmed that even beginners can acquire Python scripting skills in Blender, using a practical approach that encourages immediate application of knowledge.

In the future, the author plans to diversify the course by adding more interactive elements and modern learning forms. There is also a need for a more comprehensive curriculum that would cover all aspects of the Blender Python API and the needs of advanced students. It is also advisable to collect student feedback to improve the quality and structure of the learning materials. This thesis will help to popularize Blender and Python more widely, providing beginners with access to these powerful tools. In the future, the author wants to create an even more comprehensive curriculum that would cover all aspects of the Blender Python API.

KASUTATUD KIRJANDUSE LOETELU

1. Registered Moodle sites, <https://stats.moodle.org/sites/index.php?country=EE>, vaadatud 13.11.2024
2. About us, <https://moodle.com/about/>, vaadatud 13.11.2024
3. A.Al-Ajlan, H. ZedanWhy Moodle, https://www.researchgate.net/publication/232615507_Why_Moodle, avaldatud 10.2008
4. What is TalentLMS?, <https://www.trustradius.com/products/epignosis-talentlms/reviews?q=pros-and-cons#reviews>, vaadatud 4.12.2024
5. What is Docebo Learning Platform?, <https://www.trustradius.com/products/docebo-learning-platform/reviews>, vaadatud 4.12.2024
6. TalentLMS vs Moodle vs Docebo: Which is a better fit?, <https://www.capterra.com/compare/80691-127213-132935/Moodle-vs-Docebo-vs-TalentLMS>
7. The Freedom to Create, <https://www.blender.org/about/>
8. Blender Python API, https://docs.blender.org/api/current/info_quickstart.html
9. 10 Best Blender Alternatives for 3D Modeling in 2024, <https://www.geeksforgeeks.org/blender-alternatives-for-3d-modeling/>, avaldatud 01.07.2024, vaadatud 10.10.2024
10. Autodesk Maya: Create expansive worlds, complex characters, and dazzling effects, <https://www.autodesk.com/products/maya/overview?term=1-YEAR&tab=subscription>, vaadatud 10.10.2024
11. SketchUp Tutorial, <https://guides.library.ucsc.edu/DS/Resources/SketchUp>, loodud sügis 2019, vaadatud 10.10.2024
12. ZBrush, <https://www.maxon.net/en/zbrush-digital-sculpting-software>, vaadatud 10.10.2024
13. Cinema 4D What is cinema 4D?, <https://www.maxon.net/en/cinema-4d>, vaadatud 10.10.2024
14. SOLIDWORKS 3D CAD, <https://plmgroup.ee/tarkvara/solidworks-3d-cad/>, vaadatud 10.10.2024
15. Comparison of Maya and Blender, Which One Is the Best for You?, <https://pixune.com/blog/maya-vs-blender/>, vaadatud 10.10.2024

16. TIOBE Index for October 2024, <https://www.tiobe.com/tiobe-index/>
17. Python Popularity: Why Is It The Most Used Programming Language, <https://www.jalasoft.com/blog/is-python-the-most-used-language>, avaldatud 02.04.2024
18. Python Introduction, https://www.w3schools.com/python/python_intro.asp