

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

К. О. Трапезон, Г. Г. Власюк, О. А. Батіна

ТЕХНІЧНІ ЗАСОБИ ВИРОБНИЦТВА МУЛЬТИМЕДІЙНОГО КОНТЕНТУ ЛАБОРАТОРНИЙ ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю 171 «Електроніка»,
спеціалізацією «Електронні та інформаційні технології кінематографії та
аудіовізуальних систем»*

Київ
КПІ ім. Ігоря Сікорського
2017

Рецензенти: *Михайлов С.Р.*, к.т.н., доцент

Відповідальний редактор *Савченко Ю.Г.*, д.т.н., професор

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 4 від 21.12.2017 р.)
за поданням Вченої ради факультету (протокол № 12/2017 від 18.12.2017 р.)*

Електронне мережне навчальне видання

Трапезон Кирило Олександрович, канд. техн. наук, доц.
Власюк Ганна Григорівна, д-р. техн. наук, проф.
Батіна Олена Анатоліївна

ТЕХНІЧНІ ЗАСОБИ ВИРОБНИЦТВА МУЛЬТИМЕДІЙНОГО КОНТЕНТУ ЛАБОРАТОРНИЙ ПРАКТИКУМ

Технічні засоби виробництва мультимедійного контенту: Лабораторний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні та інформаційні технології кінематографії та аудіовізуальних систем» / К. О. Трапезон, Г. Г. Власюк, О. А. Батіна ; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 13,23 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2017. – 101 с.

Лабораторний практикум надає змогу студентам отримати знання та практичні навички при роботі з сучасним спеціалізованим програмним забезпеченням, яке спрямоване на реалізацію повного технологічного циклу виробництва різнопланового аудіовізуального матеріалу для цифрового кіно-, радіо- та телебачення. Особливістю практикуму є те, що розглянуті положення можуть бути використані фахівцями і при створенні аудіовізуального контенту з залученням комп'ютерних технологій в режимі скриптингу, адже досить детально розглянуто базові положення об'єктно-орієнтованої мови програмування Python по відношенню до комплексу програм The Foundry Nuke.

© К. О. Трапезон, Г. Г. Власюк, О. А. Батіна, 2017
© КПІ ім. Ігоря Сікорського, 2017

ЗМІСТ

ВСТУП.....	4
Мета та завдання лабораторних робіт	5
Навчально-методичний опис робіт практикуму.....	7
Критерії оцінювання та вказівки про порядок захисту лабораторних робіт.	100
Список рекомендованої літератури.....	101

ВСТУП

Дисципліна “Технічні засоби виробництва мультимедійного контенту” вважається однією з основних в комплексі підготовки фахівців в галузі електроніки за спеціалізацією “Електронні та інформаційні технології кінематографії та аудіовізуальних систем” і ставить за мету детальне висвітлення з усіма технічними нюансами етапів, які використовуються в процесі створення, обробки вихідного аудіовізуального контенту за новою нодовою архітектурою для компаній та студій, які займаються не тільки виробництвом і підготовкою до ефіру телевізійних програм та інформаційного продукту різного жанру, але які є посередниками у процесі передавання інформаційного продукту до регіональних компаній та студій.

Разом з тим, названа дисципліна шляхом виконання комплексу лабораторних робіт дозволяють оволодіти базисними сучасними знаннями та вміннями з питань обробки, відновлення та аналізу мультимедійних даних різної структури (у тому числі і створення комп’ютерних ігор) і розкривають в практичному відношенні особливості роботи зі спеціалізованими програмними комплексами, які дозволяють, наприклад, проводити планарне трекування, ротоскопінг даних тощо.

МЕТА ТА ЗАВДАННЯ ЛАБОРАТОРНИХ РОБІТ

Метою лабораторних робіт з дисципліни “Технічні засоби виробництва мультимедійного контенту” є оволодіння студентами практичних навичок та вмінь при роботі з професіональними апаратно-програмними комплексами з підготовки аудіовізуального контенту. Так, на лабораторних заняттях студенти мають змогу ознайомитися з принципами налаштування, роботи та обробки вихідних аудіо та відеоматеріалів з кінцевою метою створення контенту для подальшого його використання в програмах телевізійних студій. Окремою задачею курсу лабораторних робіт можна вважати набуття студентами знань з принципів підготовки контенту у розрізі послідовності дій, які слід виконати перед експортом на локальні носії даних.

Під час виконання лабораторних робіт з названої дисципліни студенти:

- досліджують принципи організації структурованого зберігання мультимедійної інформації засобами програми iTunes;
- навчаються створювати найпростіші шаблони комп’ютерної гри на основі використання 2D-конструктора з залученням найбільш відомих мультимедійних платформ;
- отримують навички з виконання процедури налаштування 3D-трекінгу для відзнятого відеопотоку, як складової одиниці комп’ютерної графіки при створенні мультимедійного контенту;
- вивчають можливості налаштування елементів маскування в середовищі програми After Effects;
- навчаються виконувати планарне трекування при роботі з контентом, проводити планарне відслідковування та вміти розрізняти за яким алгоритмом проводиться програмне відслідковування окремих частин відеоряду;
- на основі правил та процедур програми Nuke вивчають основні інструменти програми, які дозволяють користувачу виконувати основні технічні процедури з різноплановим контентом. Отримують знання з організації нодової структури проекту програми;

– вивчають правила написання скриптів для створення, зміни та обробки аудіовізуального контенту (зображення та відеопослідовності) в спеціалізованому вузловому середовищі Nuke FX.

НАВЧАЛЬНО-МЕТОДИЧНИЙ ОПИС РОБІТ ПРАКТИКУМУ

Лабораторні заняття дозволяють оволодіти студентам практичними навичками та вміннями при роботі з професіональними апаратно-програмними комплексами з підготовки різнопланового аудіовізуального контенту. Окремою задачею курсу лабораторних робіт можна вважати набуття студентами знань з принципів підготовки контенту у розрізі послідовності дій, які слід виконати перед експортом на локальні носії даних.

Ваговий бал за виконання однієї лабораторної роботи – **3 бали**. Максимальна кількість балів за всі лабораторні роботи з дисципліни становить **24 бали**. Далі у виданні наведено навчально-методичний опис усіх лабораторних робіт з відповідними завданнями для студентів з розподілом по бригадах.

Лабораторна робота №1

Особливості організації мультимедійної інформації у програмі iTunes

Мета. Навчитись організовувати структуроване зберігання мультимедійної інформації засобами програми iTunes.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення iTunes на операційних системах Windows або Mac OS.

Завдання до лабораторної роботи.

1. Встановити програму iTunes та провести пошук усієї мультимедійної інформації на персональному комп'ютері.
2. Перенести медіатеку у нову папку на комп'ютері (C://Медіатека Бригада _).
3. Створити нову медіатеку.
4. Занотувати які формати з відповідними особливостями аудіо та відео файлів підтримує програма iTunes.
5. Додати у створену медіатеку декілька нових музикальних файлів (кількість файлів – номер бригади+1 для першої групи, +2 – для другої групи).
6. Змінити теги для 5 файлів на тег “номер бригади (число)23”.
7. Занотувати та визначити призначення основних елементів інтерфейсу програми. Визначити, як можна зберегти/змінити чи об'єднати створений перелік музикальних файлів.
8. Створити новий плейліст з 10 музикальних файлів (бригада 1), інші бригади – 10-номер бригади.
9. Сформулювати висновки з лабораторної роботи.

Теоретичні відомості

iTunes – програма для організації та упорядкування музикальних та відео файлів, яка дозволяє синхронізувати різноманітні складові мультимедійного контенту. Програма існує для різних операційних систем і відрізняється лише

функціоналом, який доступний користувачу і має до того ж відмінності залежно від ступеню інтеграції з системою та доступними додатками.

Програма iTunes (рис.1) організовує медіатеку відповідно до тегів, які повинні бути прописані у мультимедійних файлах. Через це для коректної та зручної роботи в програмі спочатку слід уявляти собі що таке медіатека і навести порядок у тегах медіафайлів.

Тег (tag) – текстова та графічна інформація, яка зберігається у мультимедійному файлі разом з відео/аудіо. В якості такої інформації може бути назви треків, виконавців, час виходу альбому, тексти пісень, зображення альбомів тощо.

Медіатека- база даних, яка містить різноманітну інформацію про усі проіндексовані медіафайли на ПК.

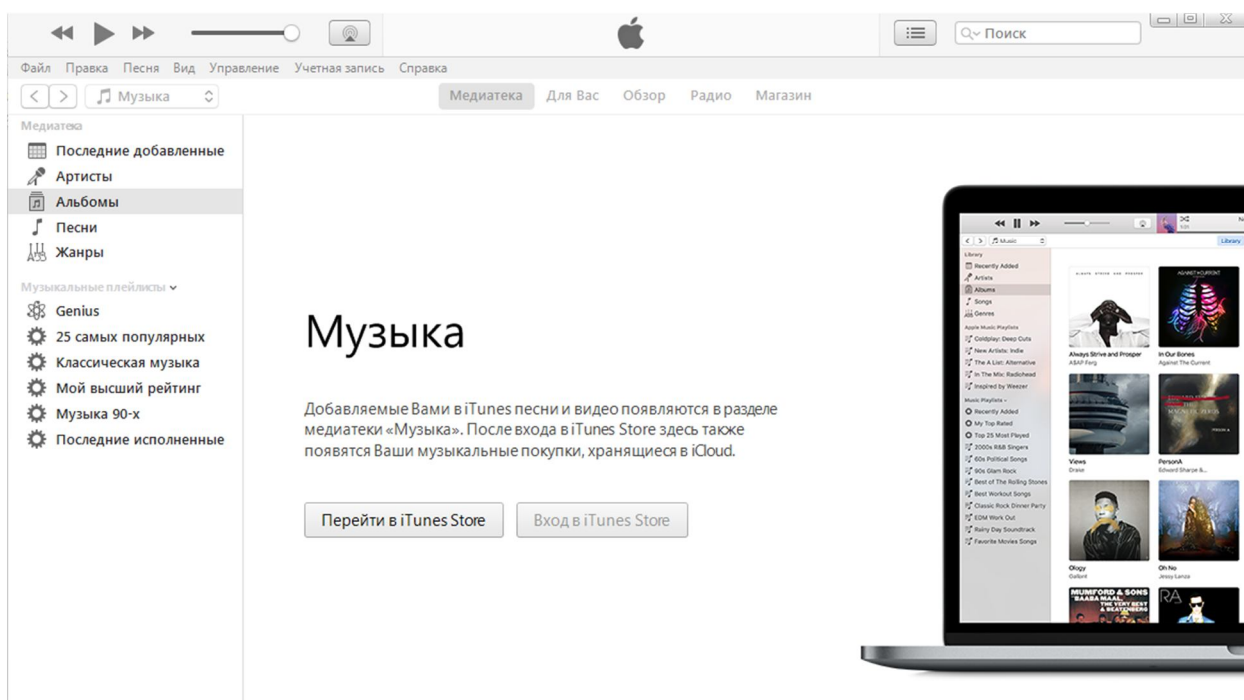


Рисунок 1 – Зовнішній інтерфейс програми

Такий підхід насамперед дозволяє створити Smart Playlists, тобто за ознаками жанру, років виходу пісень тощо.

Крім цього, інформація з тегів досить часто використовується у сторонніх додатках та на портативних девайсах.

1.1 Робота з бібліотекою.

Алгоритм роботи програми iTunes полягає у тому, що користувачу не потрібно проводити жодних маніпуляцій стосовно організації файлів через відомі файлові менеджери операційної системи чи на основі сторонніх встановлених програм. Цю процедуру програма виконує самостійно і дискові операції не має сенс виконувати.

Крім цього, програма iTunes працює з власною базою даних, де зберігаються шляхи до медіа файлів, їх рейтинги, кількість переглядів тощо. База даних за замовчуванням зберігається у папці користувача (Мої документи\Моя Музика\iTunes) і може мати назву *iTunes Library.itl*. До речі, усі файли у папці iTunes буде зведено до ієрархії виду /Артист/Альбом/Трек відповідно до тегів, які визначено у медіафайлах.

За замовчуванням усі файли, які Ви додаєте в iTunes будуть надіслані і в папку iTunes. Якщо ж відмовитись від цієї опції, то видалення файлу треба буде проводити два рази, спочатку з бібліотеки програми, потім – фізично з жорсткого диску. Якщо ж необхідно відключити цю функцію, то у налаштуваннях програми слід зняти відповідну позначку (рис.2). Інший спосіб відмінити цю опцію – при перенесенні файлу у вікно програми натиснути комбінацію клавіш ctrl+alt.

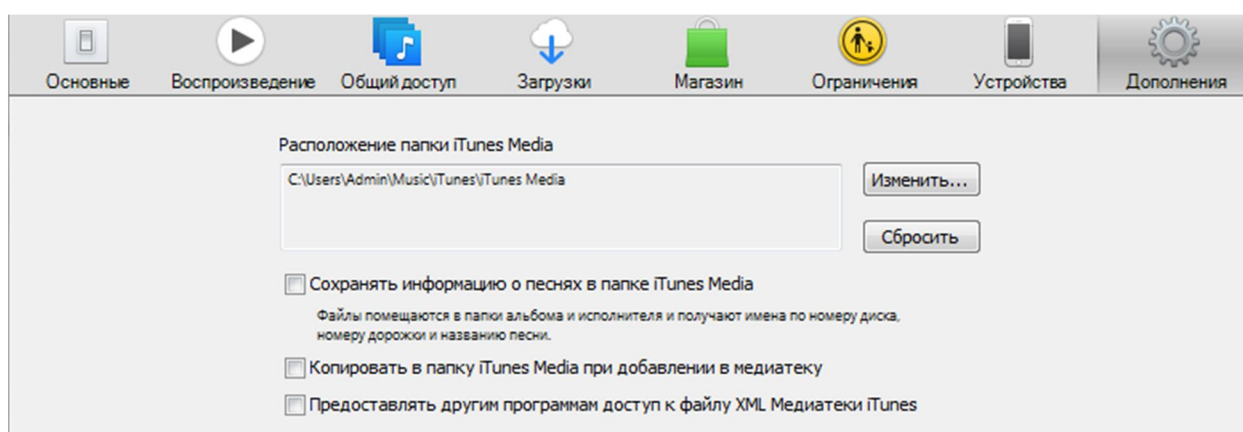


Рисунок 2 – Налаштування програми

1.2 Перенесення бібліотеки iTunes.

У випадку, коли виникає ситуація, що на жорсткому диску, де знаходиться бібліотека і на який копіюються файли може бути недостатньо місця, і в цьому випадку треба бібліотеку перенести на інший диск. Ця обставина може бути і коли необхідно зібрати усі імпортовані файли в iTunes в одне місце. Для цього слід відкрити налаштування програми і через кнопку “Изменить” вказати новий шлях бази (рис.3)

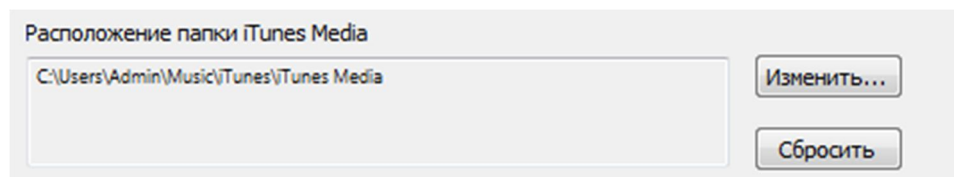


Рисунок 3 – Налаштування

І після цього треба через верхнє меню знову організувати медіатеку: Файл\Медіатека\Організувати медіатеку – Собрать файлы (рис.4). Після цього стару папку можна видалити з диску.

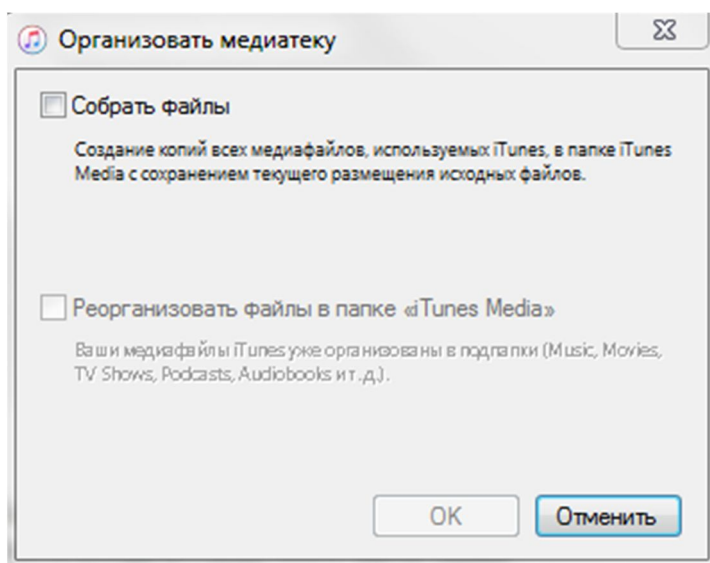


Рисунок 4 – Опція роботи з медіатекою

1.3 Створення нової бібліотеки iTunes.

За необхідністю можна створити нову бібліотеку файлів з розширенням .itl. Для цього, при запуску iTunes треба затиснути клавішу Shift в результаті чого з’явиться відповідне вікно (рис.5).

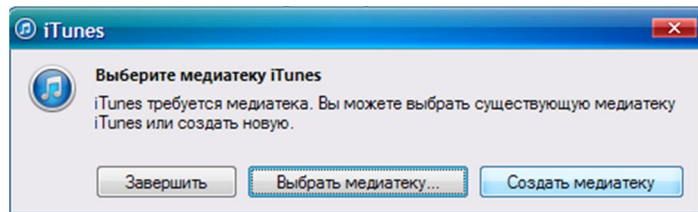


Рисунок 5 – Створення нової медіа теки

1.4 Імпорт файлів.

Для того щоб додати файли у медіа теку треба зайти у головне верхнє меню програми за напрямом: Файл\Добавить файл в медиатеку. Якщо ж включені опції копіювання у папку iTunes зі збереженням інформації про файли, то якщо вони мали теги то їх назви після досилання будуть складені у форматі – номер треку та назви композиції.

1.5 Медіатека.

Медіатека програми iTunes розділена на декілька категорій:

- Музика. Це аудіо файли за виключенням файлів аудіо книг (.m4b) та рінгтонів (.m4r). До цієї категорії також відносять відео файли з лейблом відео кліп.
- Фільми. Сюди відносять усі відео файли за виключенням відео кліпів з першої категорії.
- Телешоу (серіали). Тут лише ті відео файли, які мають відповідну позначку серіалу. Крім цього вони мають вказівник – “не переглянуто” і можуть групуватись за сезонами.
- Подкасти. Програми у форматі радіопередач, яку можна завантажити з мережі. Досить часто подкасти порівнюють зі стрічками новин, блогами у форматі аудіо.
- Аудіокниги. В цій категорії знаходяться аудіо файли з розширенням .m4b або аудіо файли з позначкою аудіокнига. Відмінністю цих файлів є встановлена функція фіксації місця програвання і ці файли не відтворюються якщо встановлена опція програвання у випадковому режимі.

- Програми. Зберігаються програми для iPhone та iPad.
- Радіо. Зберігається список каналів інтернет-радіо.
- Рінгтони. Відображаються мелодії дзвоника для iPhone.

1.6 Зміна тегів.

У випадку додавання файлів в базу даних програми можуть виникнути проблеми з тегами. Це насамперед стосується коли вони відсутні, або коли відрізняється система символічного кодування. Програма може сама провести редагування тегів але зручніше використати сторонні програми для виправлення тегів або їх додавання. Серед них можна використати такі: TuneUp Companion, TriTag, MusicBrainz Picard.

Контрольні запитання.

1. Визначте де безпосередньо знаходиться медіатека програми iTunes.
2. За яким алгоритмом будуть розташовані файли у папці iTunes, якщо в них відсутні усі теги ?
3. Чим відрізняється програма iTunes при переході з Windows на Mac OS ?
4. Що таке подкаст в середовищі iTunes ?
5. Що таке Genius у налаштуваннях верхнього меню програми iTunes ?
6. Що треба зробити у налаштуваннях програми, аби файл при додаванні в iTunes не був про дубльований у відповідну папку iTunes ?

Лабораторна робота №2

Особливості розробки найпростіших мобільних додатків

Мета. Навчитись створювати найпростіші шаблони комп'ютерної гри на основі використання 2D-конструктора з залученням найбільш відомих мультимедійних платформ.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Multimedia Fusion (нова назва - Clickteam Fusion) з підтримкою платформ – Flash, HTML 5, iOS, Android.

Завдання до лабораторної роботи.

1. Створіть новий додаток в середовищі програми Clickteam fusion. Поясніть які коменти можна додати у робочу площину програми до безпосереднього наповнення кадрів проекту.
2. Визначити які об'єкти доступні користувачу при наповненні кадру і в чому полягає їх сутність ?
3. Створити гру з початком і кінцем з певної кількості кадрів (1 група – кількість кадрів = номер бригади+3 (парні числа); 2 група - кількість кадрів=номер бригади+3 (непарні числа)). При цьому, повинні бути два клоновані кадри. Додаткові опції - встановити таймер очікування (бриг. 1), виключити звук (бриг.2), перезапустити гру (бриг.3), прибрати вказівник миші (бриг.4), ввести ім'я користувача (бриг.5)
4. Визначити у створених кадрах принаймні два відмінні результати події. В кадрі повинно бути не менше двох подій (окрім початку та кінця).
5. У створеній грі забезпечити об'єкти динаміки.

Теоретичні відомості

Конструктор зі створення комп'ютерної гри – найпростіший спосіб створення розважального мультимедійного контенту 2D або 3D на основі використання готових шаблонів або налаштувань і без вивчення C# чи Java. В цій лабораторній роботі мова буде йти про створення комп'ютерної гри без знання

особливостей щодо написання коду. Конструктори наразі поширені при створенні прототипів в Game Jams. Вони дозволяють створювати ігри для PC, iOS, Android, HTML 5, Flash тощо. Але, для операційної системи слід додати в програмі додатковий модуль export module ios. У даному випадку модуль конвертує гру в код платформи Xcode задля подальшого тестування програми на пристроях Apple. У властивостях програми (free edition) доступне експортування 3 типів (рисунок 1).

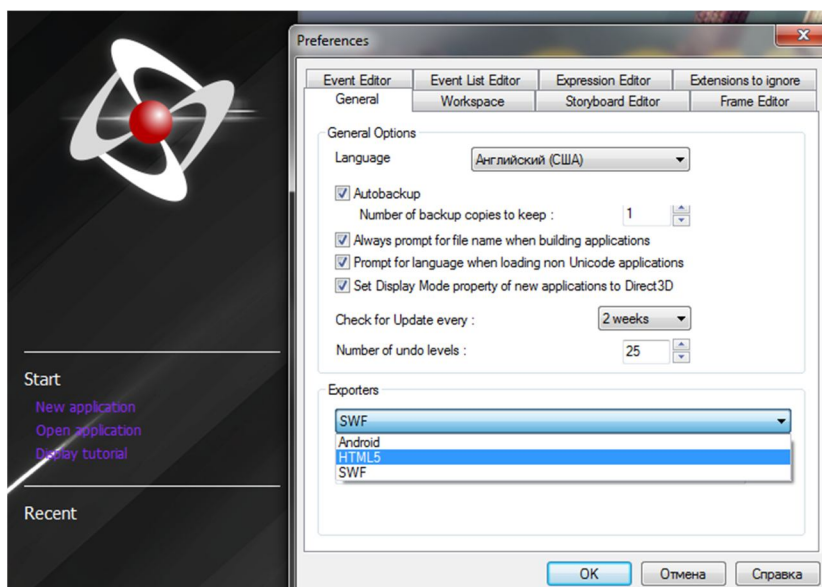


Рисунок 1 – Можливості експортування

Програма Multimedia Fusion є аналогом чи реалізацією концепції створення комп'ютерної гри без написання програмного коду (рис.2). Іншими прикладами таких програм можна відзначити – Game Maker Studio або Construct 2.



Рисунок 2 – Конструктор

Отже, програма Clickteam Fusion має простий інтерфейс, який у вигляді робочого поля (Workspace) складається з декількох робочих областей (рис.3) – Workplace Toolbar, Properties Toolbar, Layers Toolbar та Library Toolbar (не виведено за замовчуванням)

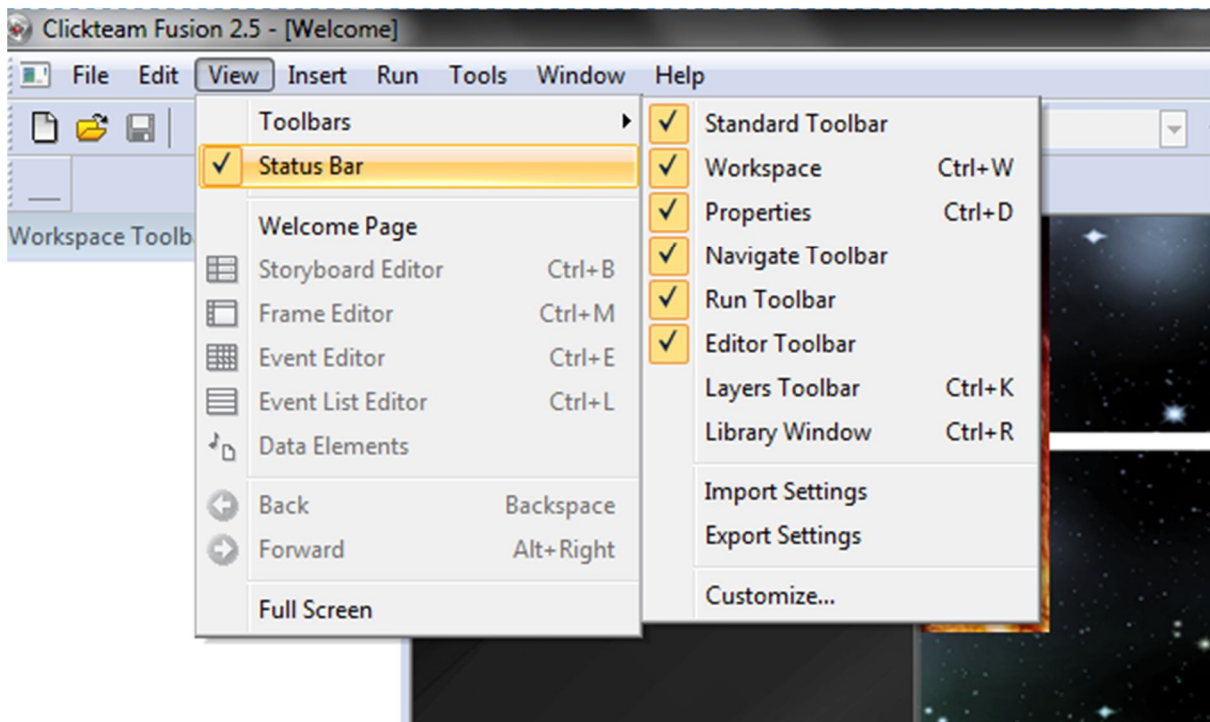


Рисунок 3 – Доступні інструменти користувачу

Комп’ютерна гра, як приклад створеного мобільного контенту в конструкторі зазвичай складається з сцен, які за традицією описують у формі **кадрів** (фреймів). Кожен кадр – об’єкт, який дозволяє вирішити певне завдання. Перед тим як почати створювати елементи гри в середовищі Multimedia Fusion слід створити новий додаток. Для цього у робочій площині програми є меню (рисунок 4), де слід вибрати пункт New application (під словом Start).

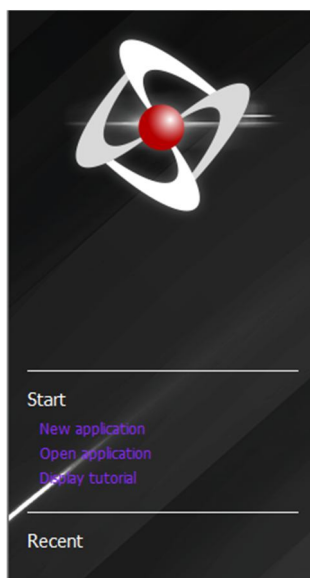


Рисунок 4 – Початок роботи

Результатом цих дій буде модернізація робочого поля програми (рисунок 5)

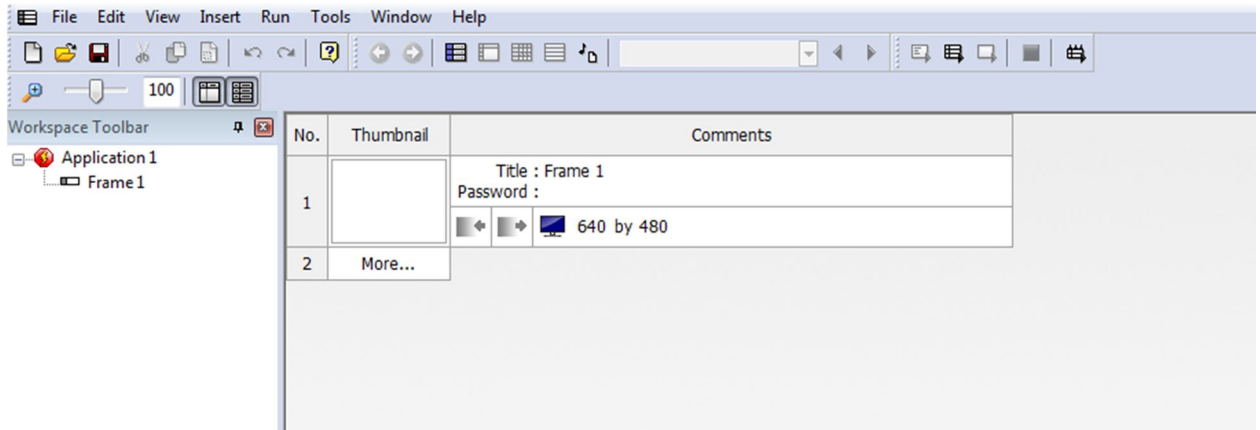


Рисунок 5 – Початкові налаштування інтерфейсу

Для того щоб перейти до наповнення кадру об'єктами слід перейти до редактору кадрів. Для цього зліва у області Workplace Toolbar слід виділити рядок Frame 1 і через праву кнопку миші вибрати пункт "Frame editor". Зверніть увагу на властивості кадру – органи управління, які розміщено у лівій нижній частині програми (рис.6)

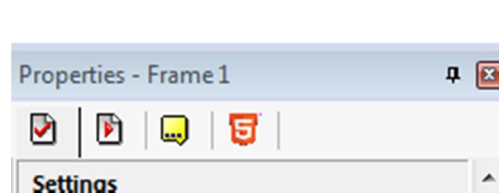


Рисунок 6 – Властивості кадру

Зверніть увагу, що налаштування кожного кадру гри визначається трьома режимами (рисунок 7) – Frame editor, Event Editor, Event List Editor.

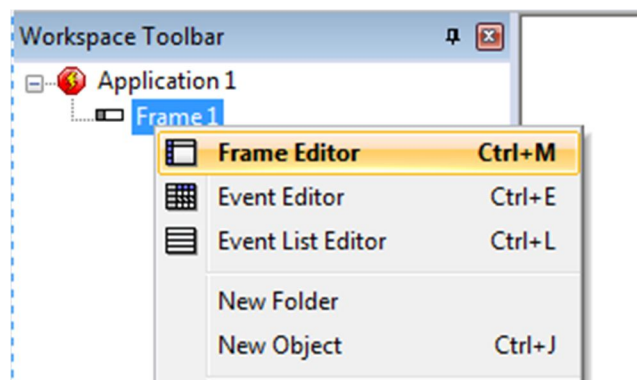


Рисунок 7 – Режими налаштування кадру гри

Для створення першого кадру гри (початок запуску гри) перейдемо для елементу frame 1 в режим редактора кадрів. У цьому режимі через праву кнопку мишу зі списку виберемо пункт “Insert Object” (рисунок 8).

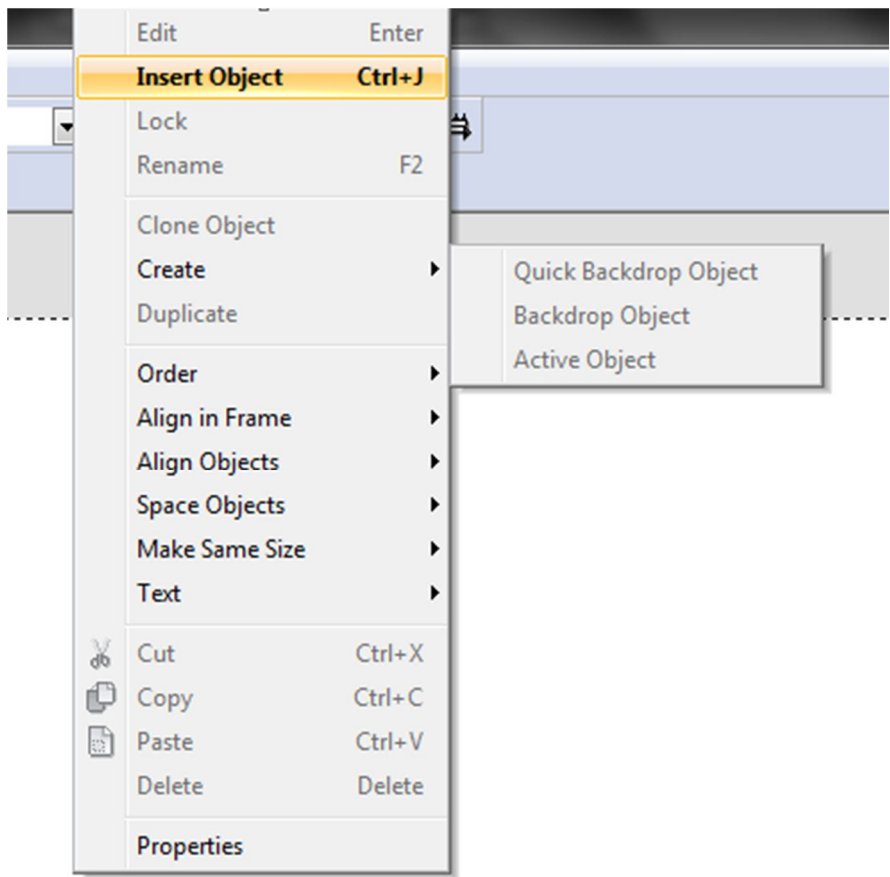


Рисунок 8 – Початок створення гри (наповнення кадру)

В результаті цього буде виведено вікно вибору об’єкту кадру (рисунок 9).

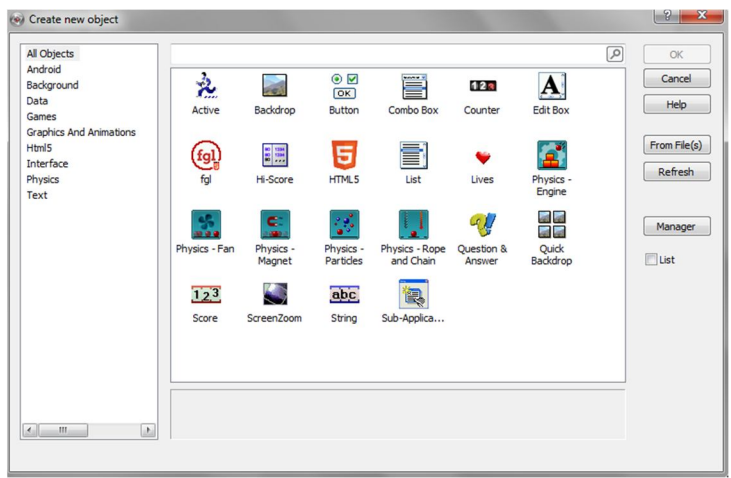


Рисунок 9 – Створення нового об’єкту

Для початку гри виберемо об'єкт "Active". Після цього вікно зникне і треба просто на робочій площині натиснути ліву кнопку миші і отримаємо перше наповнення кадру. Зверніть увагу на властивості цього елемента. Після цього треба додати подію, яка буде асоціюватись з цим об'єктом у грі. Переходимо в режим редактор подій (Event Editor), для цього кадру (рисунок 10).

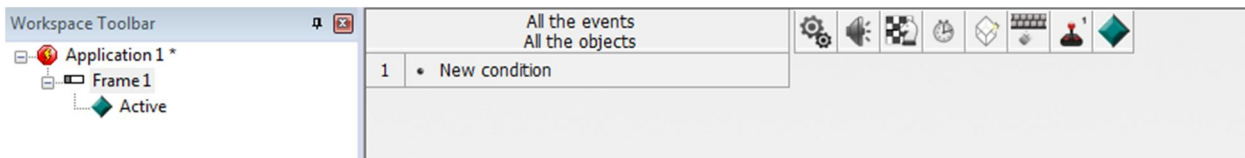


Рисунок 10 – Редактор подій

Тут слід визначити події (умови) при яких відбудуться обрані дії над об'єктом. Обираємо рядок "New Condition" і у відкритому вікні обираємо подію з боку користувача, наприклад для початку гри так, як показано на рисунку 11.

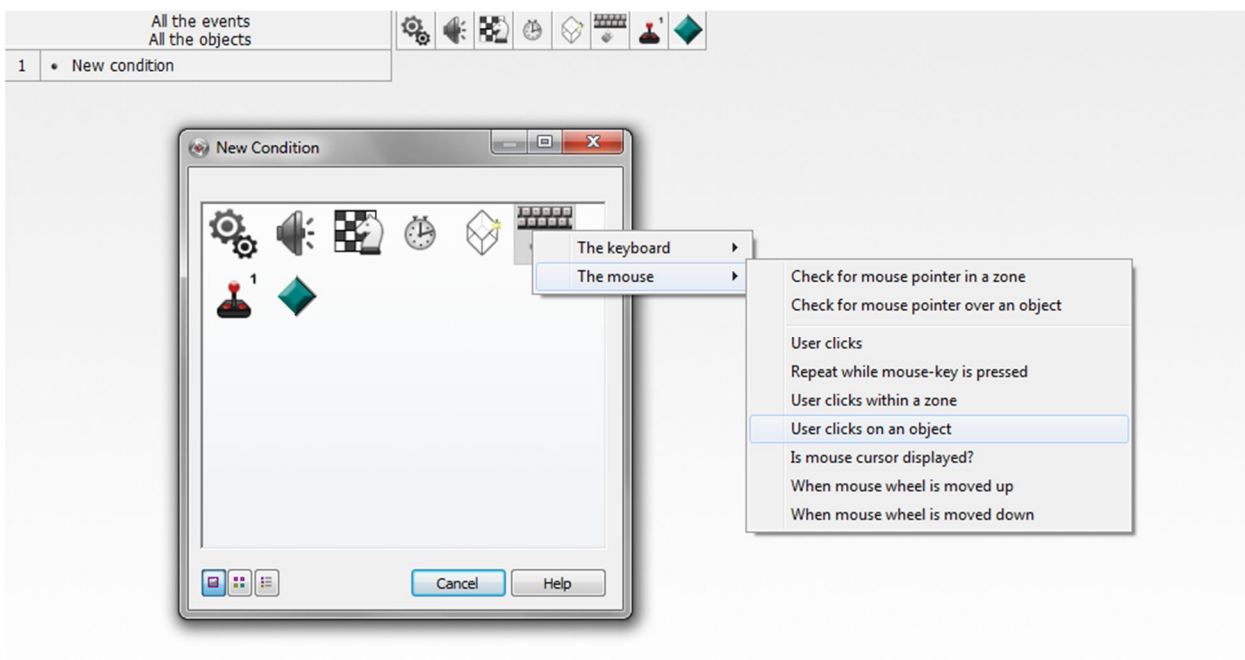


Рисунок 11 – Вибір події

Після цього слід визначити умови дії (рисунок 12), для цього автоматично буде виведено вікно.

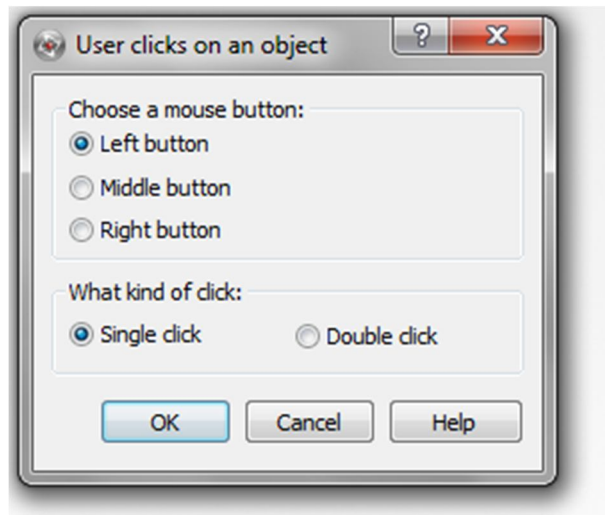


Рисунок 12 – Визначення дії події

Далі тут треба визначити об'єкт для якого призначення ця подія. В даному випадку у нас створено лише один об'єкт active, тому його і обираємо. Тепер залишилось визначити результат натискання кнопки для об'єкту (рисунок 13).

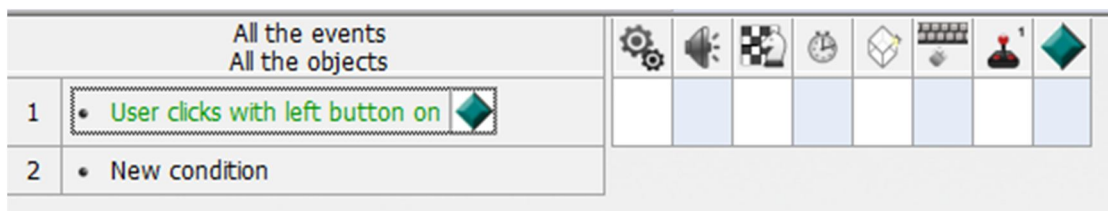


Рисунок 13 – Пропис результату дії

В даному випадку виберемо перехід на наступний кадр гри (рис.14).

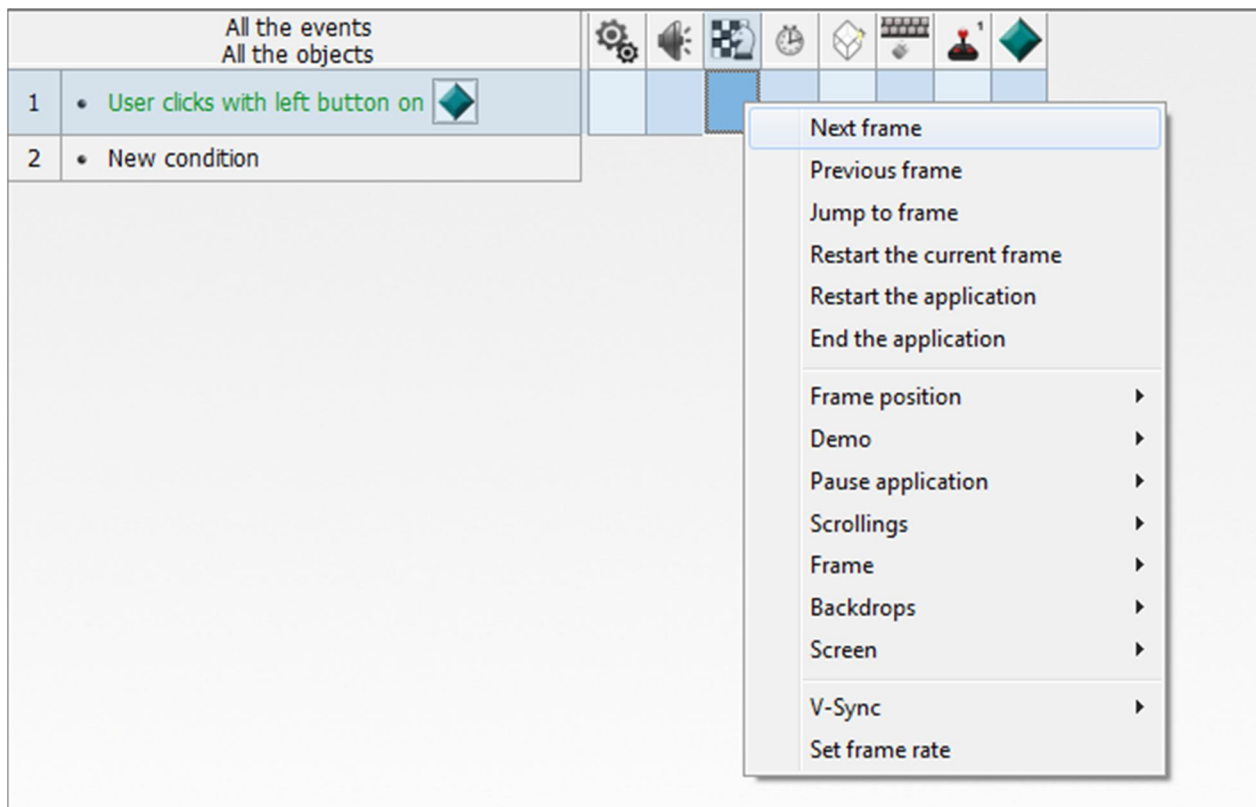


Рисунок 14 – Визначення ходу гри

Аналогічним чином визначаються інші кадри гри (рис.15)

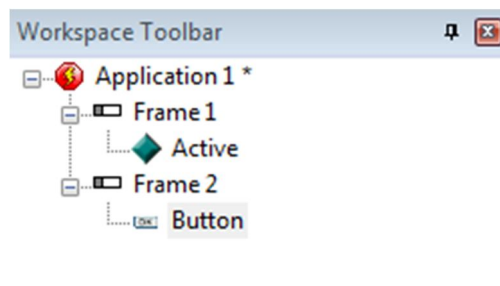


Рисунок 15 – Дерево кадрів

Контрольні запитання.

1. Які органи управління має кадр проекту в програмі Clickteam Fusion ?
Що включає в список Runtime options ?
2. Опишіть органи управління додатком через інструменти основного інтерфейсу програми (верхнє меню програми та кнопки у верхній частині інтерфейсу програми).
3. Визначте які дії в програмі Click team fusion можна проводити з об'єктом кадру “active”.

4. Наведіть які налаштування події доступні користувачу при створенні сцен гри.
5. Визначте та опишіть послідовність створення окремо визначених сцен комп'ютерної гри.
6. Наведіть можливості об'єктів з групи "Background".
7. Наведіть можливості об'єктів з групи "Android".
8. Наведіть можливості об'єктів з групи "Data".
9. Наведіть можливості об'єктів з групи "Games".
10. Наведіть можливості об'єктів з групи "Html 5".
11. Наведіть можливості об'єктів з групи "Interface".
12. Наведіть можливості об'єктів з групи "Physics".
13. Наведіть можливості об'єктів з групи "text".

Лабораторна робота №3

Особливості створення мобільного додатку

Мета. Навчитись створювати найпростіші шаблони комп'ютерної гри на основі використання 2D-конструктора для розміщення у найбільш відомих мультимедійних платформ.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Multimedia Fusion (нова назва - Clickteam Fusion 2.5) з підтримкою експортування у платформи – Flash, HTML 5, iOS, Android.

Завдання до лабораторної роботи.

1. На основі створеного проекту в середовищі програми Clickteam fusion 2.5 необхідно доповнити його кадри інтерактивним змістом. Для цього створити у другому кадрі проекту наступні об'єкти відповідно до номеру бригади. Динамічні об'єкти обираються з бібліотеки шаблонів.

Бригада №1 – 1 основний об'єкт та фон з елементами (що найменше 5 елементів);

Бригада №2 – 2 динамічних об'єкти та фон з елементами (щонайменше 4 елементів);

Бригада №3 – 3 динамічних об'єкти та фон з елементами (щонайменше 3 елементи, з яких 1 – перешкода до руху);

Бригада №4 – 3 динамічних об'єкти (1 - круглий) та фон з елементами (щонайменше 4 елементів);

Бригада №5 – 3 динамічних об'єкти (1 - квадратний) та фон з елементами (щонайменше 3 елементи, з яких одна це прямокутна коробка).

Усі елементи повинні бути різного кольору і один з них повинен бути невидимий (на вибір). Динамічні об'єкти повинні бути керовані і один з них повинен переміщуватись автоматично та в циклічному порядку (кількість рухів в циклі складає номер бригади+1). Частота циклів складає 20 кадрів.

2. Для усіх елементів проекту повинні бути встановлені межі екрану. При налаштуванні слід призначити редакторі подій в умовах оператори умов.

3. Мінімальна кількість умов в кадрі визначається за формулою – “7+номер бригади”.

4. Зібрати створену гру для експортування у всі наявні зовнішні додатки. Для гри у форматі exe у протоколі створити слайди.

Теоретичні відомості (основні налаштування)

Конструктор зі створення комп'ютерної гри – найпростіший спосіб створення розважального мультимедійного контенту 2D або 3D на основі використання готових шаблонів або налаштувань і без вивчення C# чи Java. В цій лабораторній роботі мова буде йти про створення комп'ютерної гри без знання особливостей щодо написання коду. Конструктори наразі поширені при створенні прототипів в Game Jams. Вони дозволяють створювати ігри для PC, iOS, Android, HTML 5, Flash тощо. Але, для операційної системи слід додати в програмі додатковий модуль export module ios.

Кнопки властивостей додатку чи фрейму:



- опції для запуску додатку чи фрейму відповідно ;



- задати символічні змінні;



- перехід у редактор подій проекту ;



- зміна назви проекту чи фрейму ;



- параметри для додатку, для якого розробляється гра ;



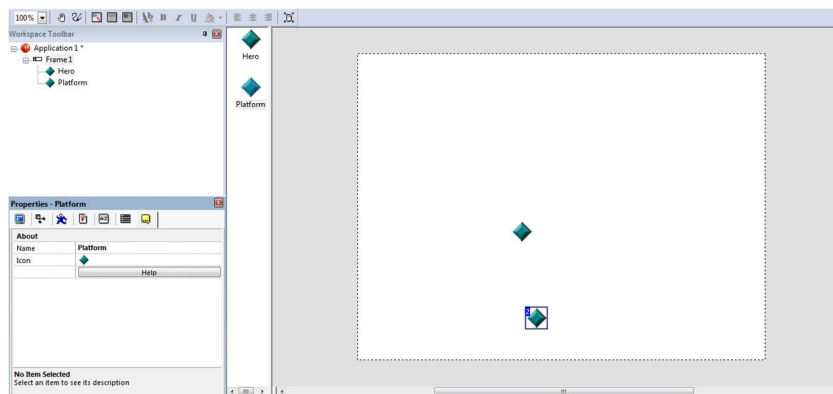
- панель для роботи з шарами в грі.

I Створення герою комп'ютерної гри (алгоритм)

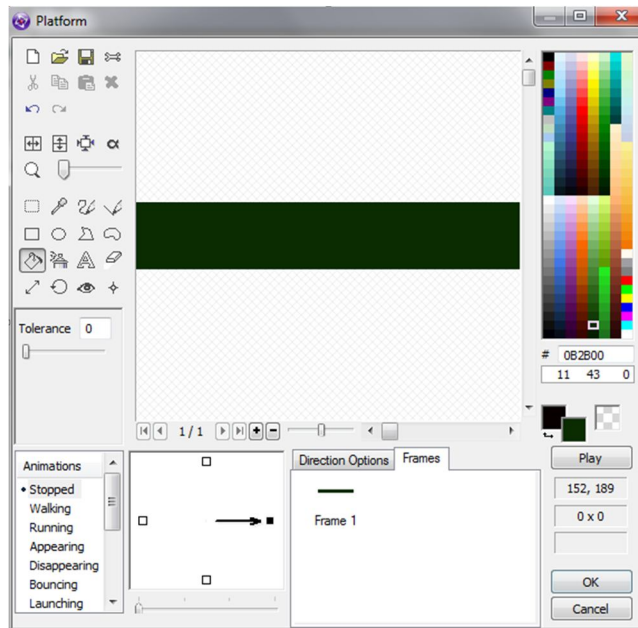
1. Створення 2 активних об'єкти – герой та платформа. Для цього у фреймі створимо або додамо 2 активні об'єкти Active (ctrl+j)



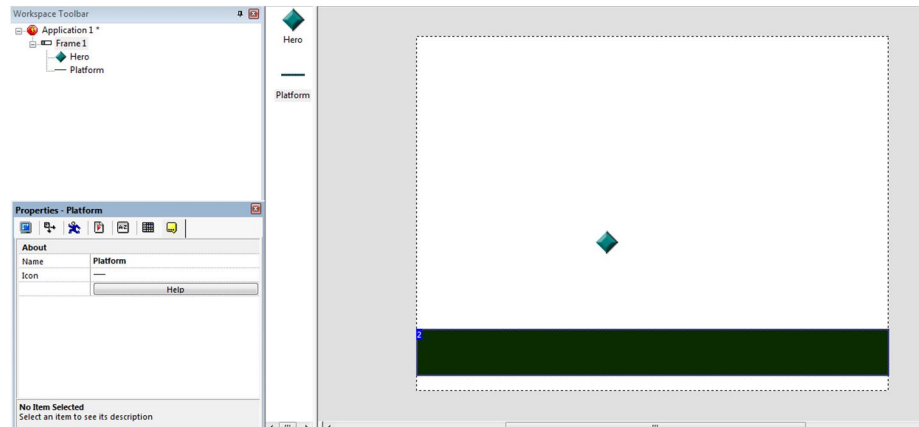
Active – це буде герой ; Active2 – платформа.



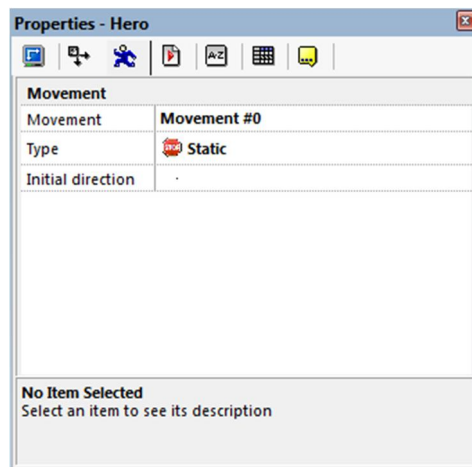
Розтягнемо платформу за усією шириною вікна робочого поля і змалюємо поверхні платформи у зелений колір. Для цього у вікні редактора кольору попередньо очистимо об'єкт (ctrl+N) і елементом Fill Tool виберемо зелений колір



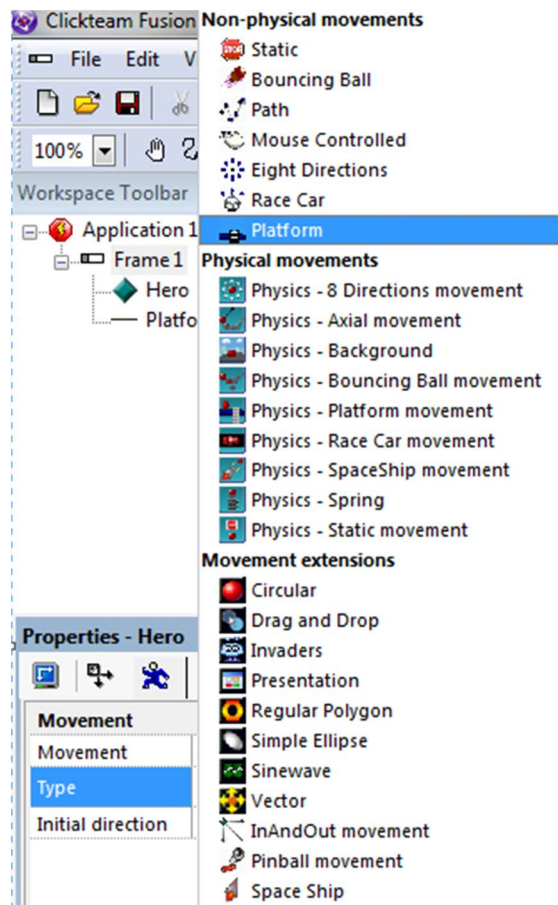
Тоді отримаємо



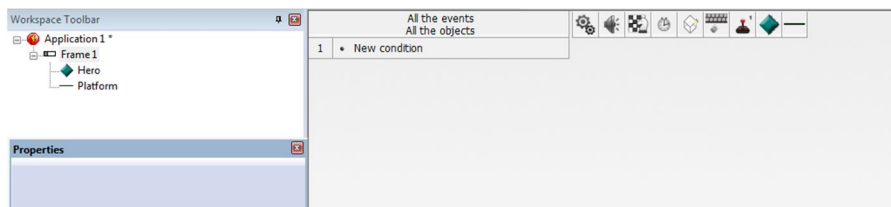
2. Для об'єкту герой (Hero) у вкладці Властивості і через пункт Movement (третій значком за порядком)



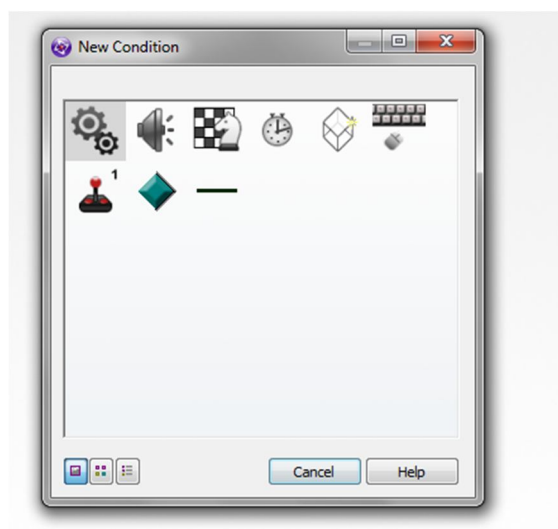
Виберемо тип руху об'єкту. Наприклад "Platform".



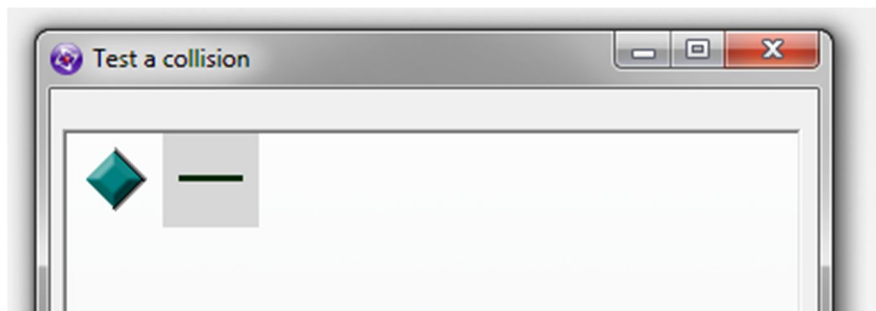
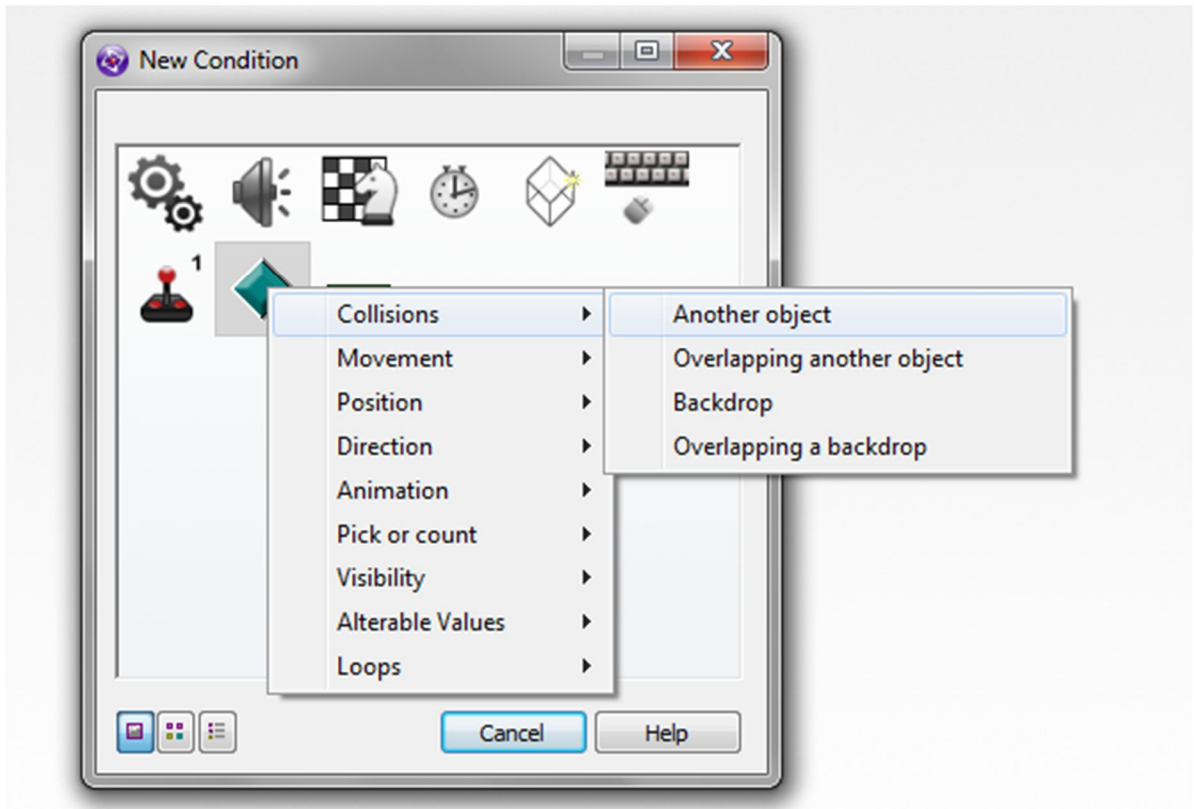
В редакторі подій:



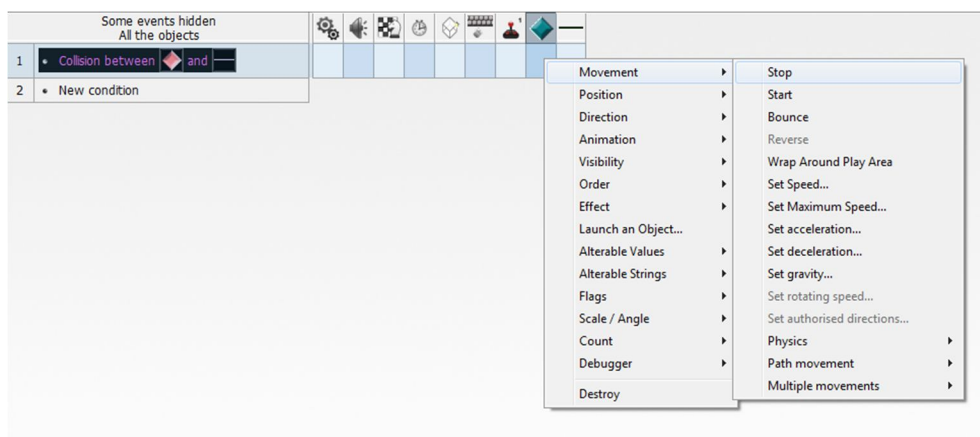
Треба зв'язати об'єкти логічною послідовністю дій. Для цього за допомогою комп'ютерної миші натискаємо на поле "New condition" і у відповідному вікні.



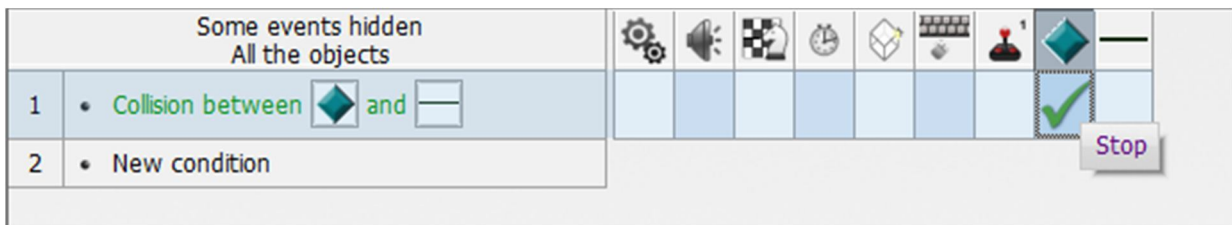
Виділяємо об'єкт Него і далі рядки Collision-Another object (торкання до іншого об'єкту).



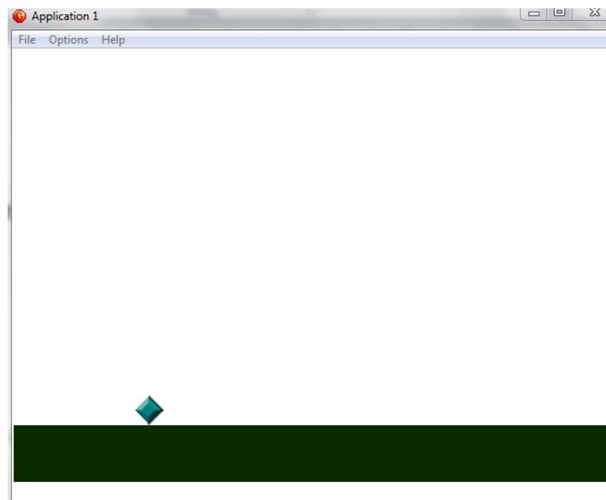
То його потрібно зупинити



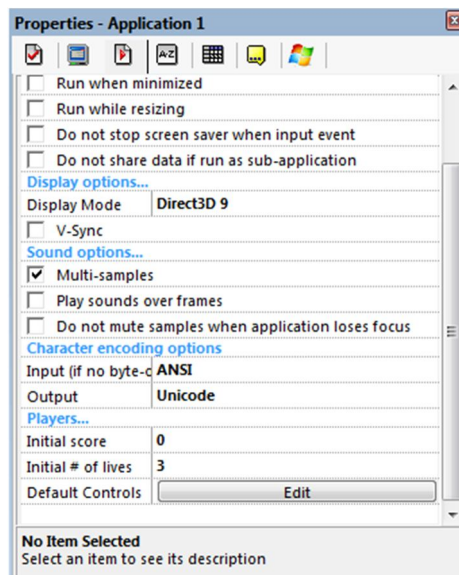
тобто



3. Запускаємо фрейм (F7). І клавішами з клавіатури (стрілки вліво, вправо, стрибок (shift)) можна керувати об'єктом.

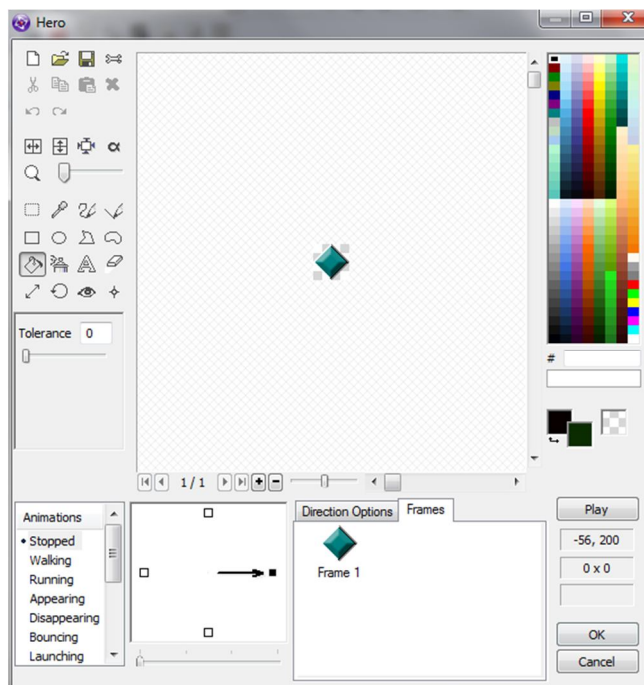


4. Для зміни клавіш з клавіатури у вікні властивостей Application через кнопку Runtime Options знаходимо пункт "Default Controls"

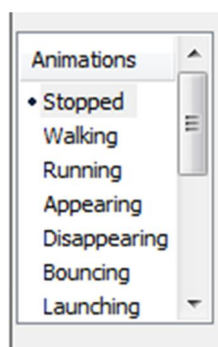


II Створення головного героя

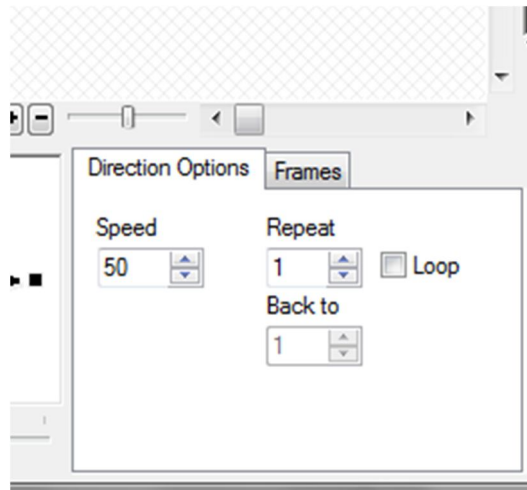
Після вибору і додавання об'єкту Active відкриємо його властивості натиснувши на зображення об'єкту в списку Frame 1 пункту Application в області вікна Workspace Toolbar



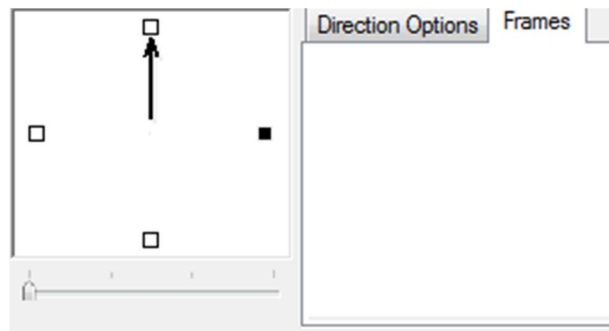
І у нижньому лівому куті вікна є область “Animation”, яка відповідає положенням або станам героя у грі. Тут треба або в цьому вікні або у редакторі Paint намалювати 2 елементи (у щонайменше положення об'єкту) і далі імпортувати ці зображення



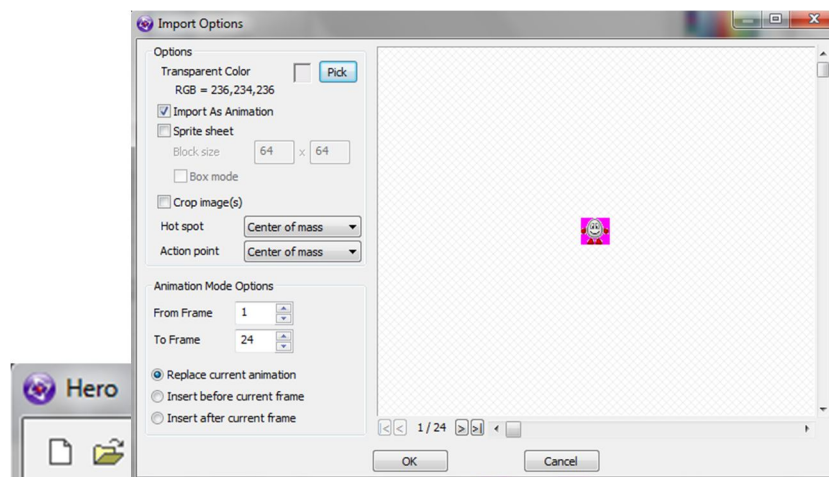
Теж саме треба зробити і для інших елементів анімації героя. Вікно на рисунку дозволяє задати циклічність частоту (швидкість анімації) і повтори (позначка Loop) при натисканні у цьому вікні кнопки програвання Play (вкладка Direct Options).



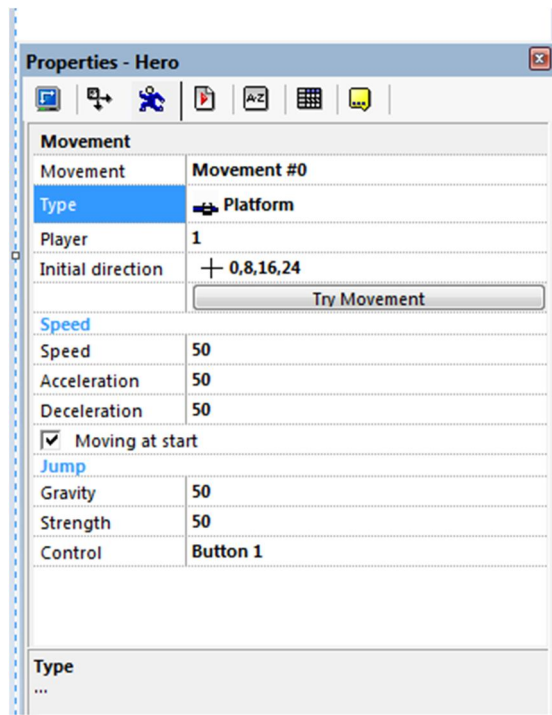
Для анімації руху (пункт Walking) треба задати положення елементів об'єкту в різних напрямках (біг та стрибки). Для цього мишею виділяємо напрямок у вікні властивостей об'єкту для цього режиму



І далі імпортуємо положення героя з готових шаблонів.



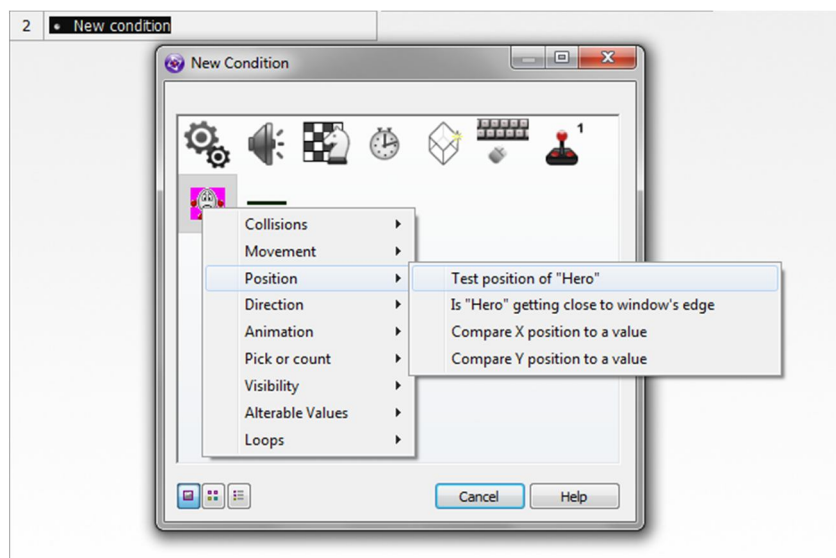
Далі для героя треба задати тип переміщення. Для цього у властивостях об'єкта відкриваємо Movement і задаємо тип



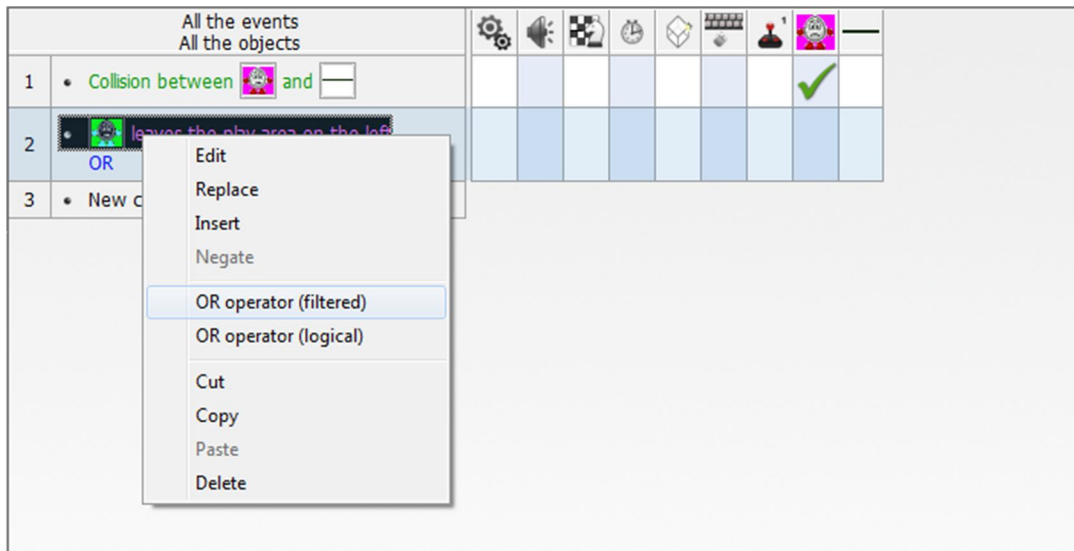
І тут визначаємо параметри сили, гравітації та швидкості, прискорення та сповільнення.

Після цього можна додати ще декілька об'єктів Active, які будуть елементами фону гри (сонце, земля, небо, тощо).

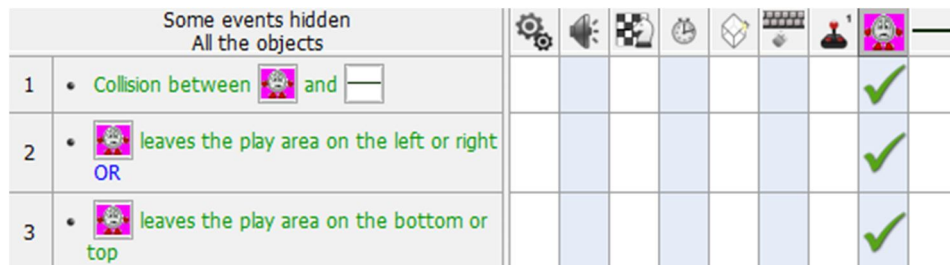
Для того щоб об'єкт не виходив за межі вікна програми, треба для нього задати ще одну умову, відповідно до дій на вікні:



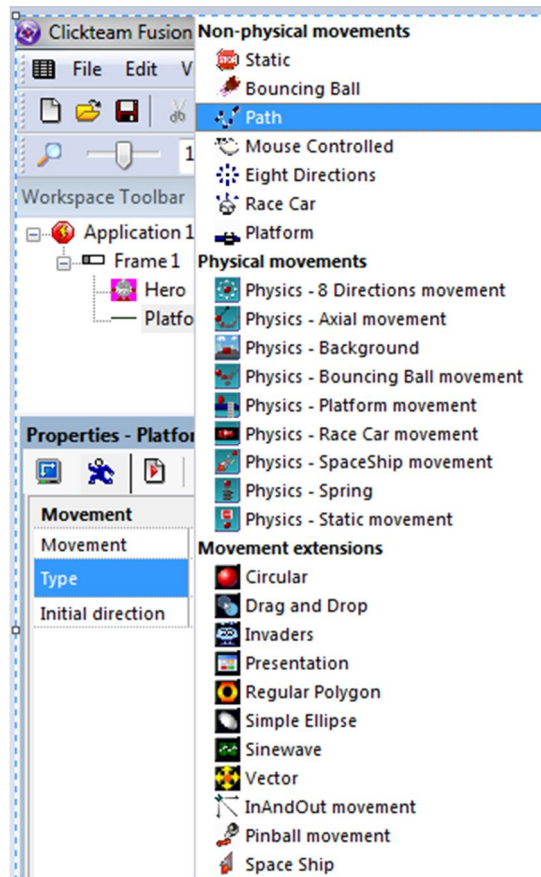
І тут для двох дій екрана, його меж можна використовувати оператор OR



Для того, щоб цю умову розповсюдити на наш об'єкт достатньо виділити позначку і лівою кнопкою миші перенести на нижні відповідні рядки :

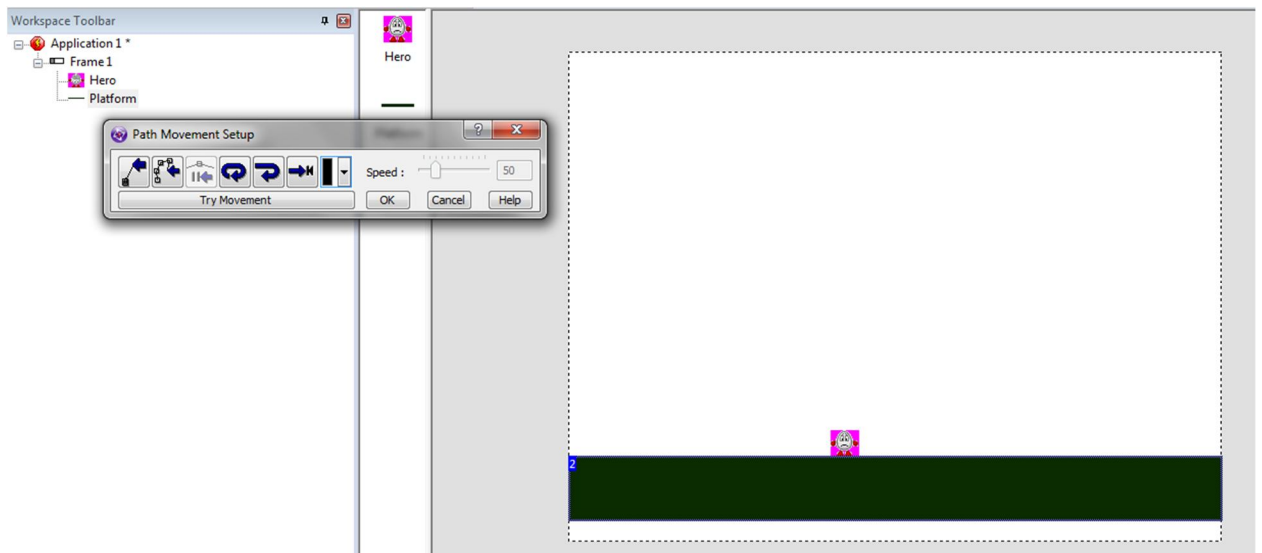


2. Для того щоб задати траєкторію руху об'єкту слід у його властивостях у вкладці Movement обрати рядок Path:

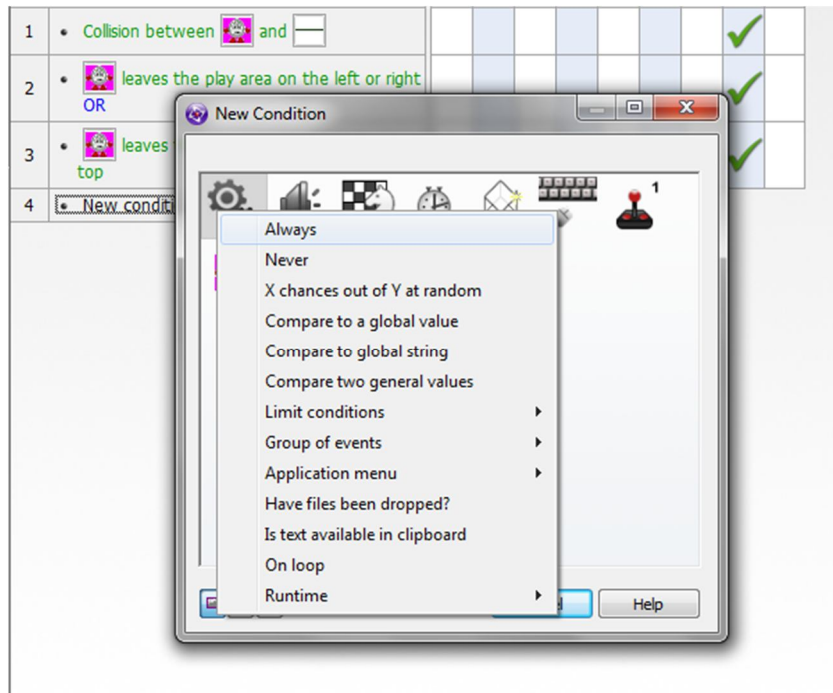


І у властивостях опції натиснути кнопку Edit в області Edit movement з

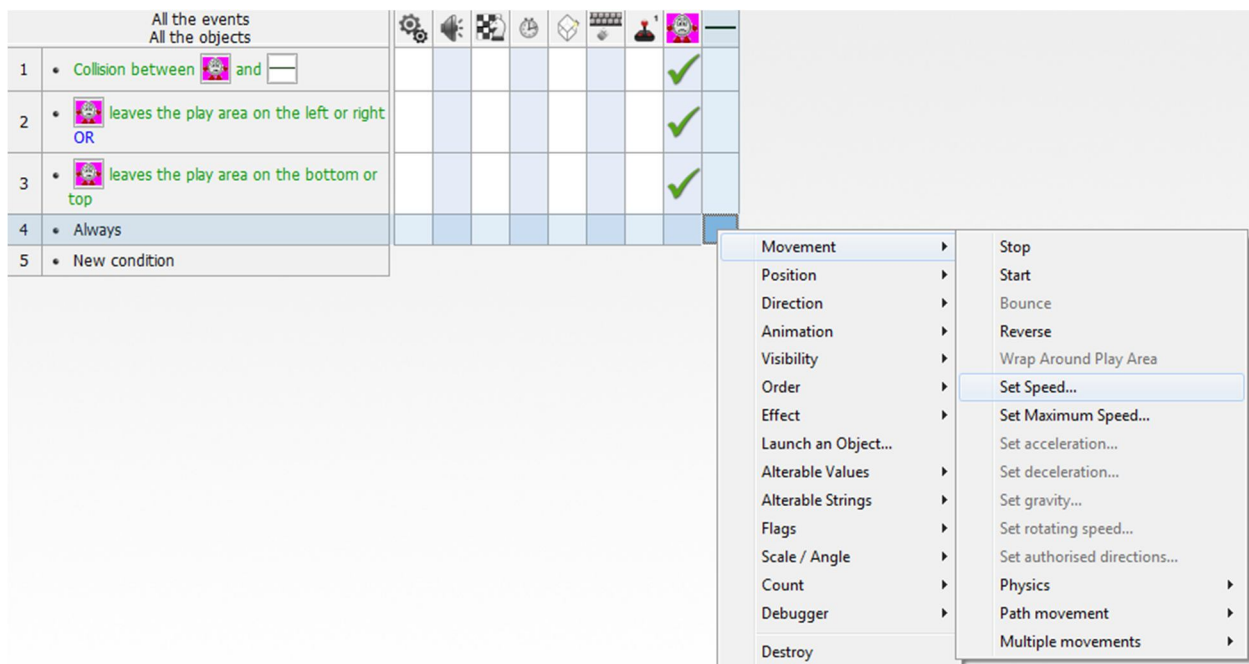
забезпеченням циклічності повторення руху (кнопка )



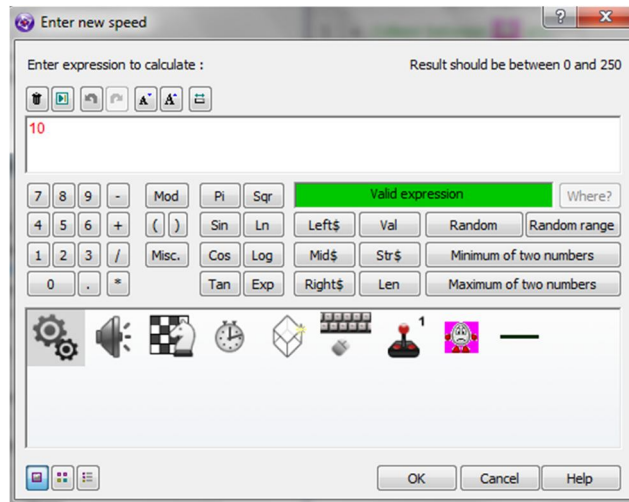
За допомогою меню Path Movement Setup встановлюємо траєкторію руху об'єкти і для нього треба встановити відповідну умову. Але для нього можна визначити умову програвання без зупинки і для цього вибираємо умову Always :







і для відповідного об'єкту в редакторі подій встановлюємо швидкість повторення цього руху.



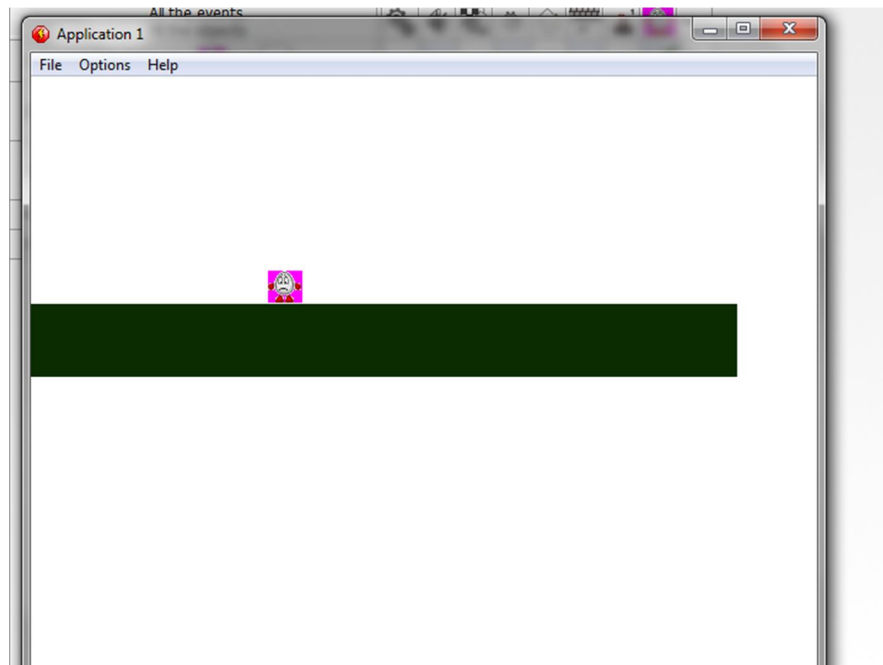
І звідси задаємо швидкість



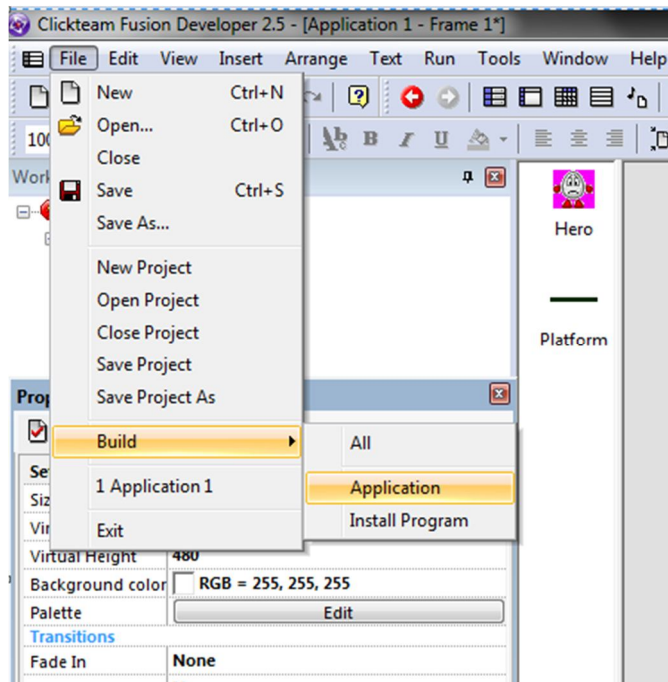
в результаті отримаємо

All the events		All the objects						
1	• Collision between  and 							✓
2	•  leaves the play area on the left or right							✓
3	•  leaves the play area on the bottom or top							✓
4	• Always							✓
5	• New condition							

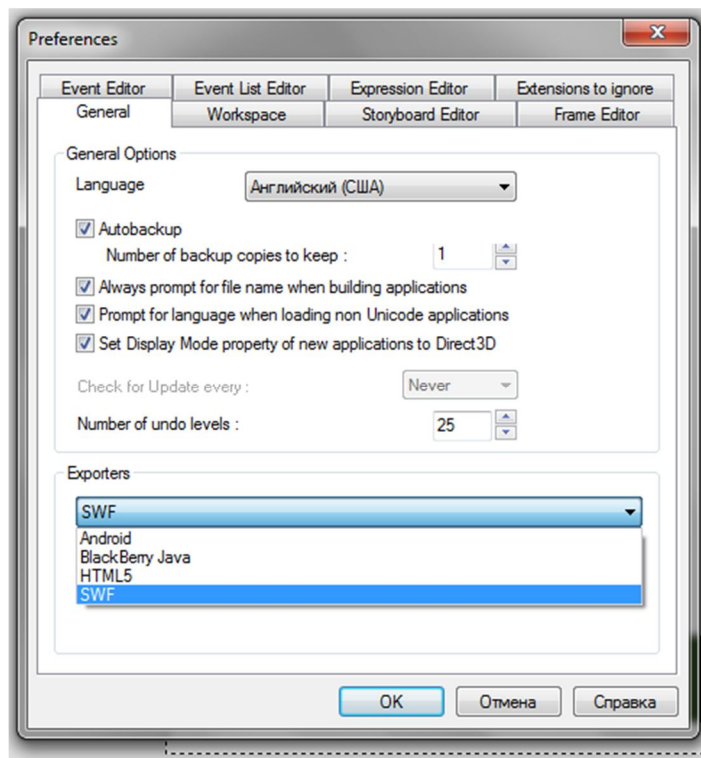
що платформа рухається разом з героєм.



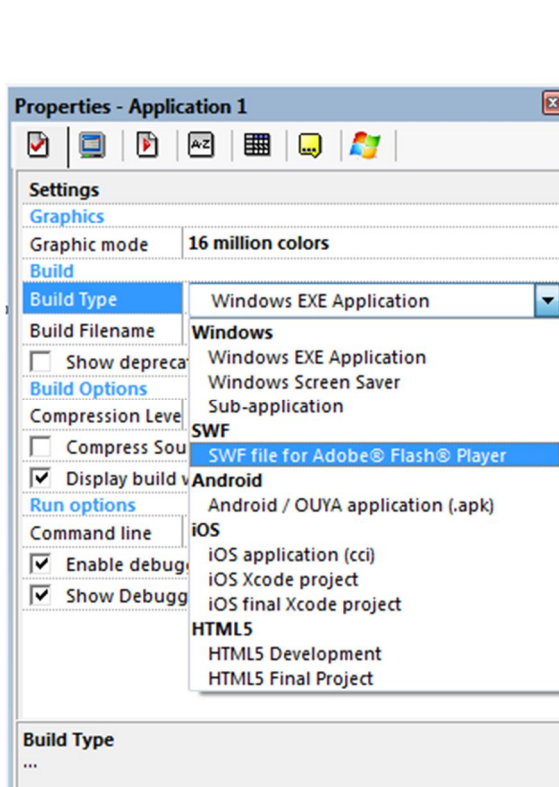
3. Для того, щоб створити гру, треба через верхнє меню обрати пункт з зібрання гри :



І до цього у вікні властивостей програми слід у вкладці General встановити модуль експорту :

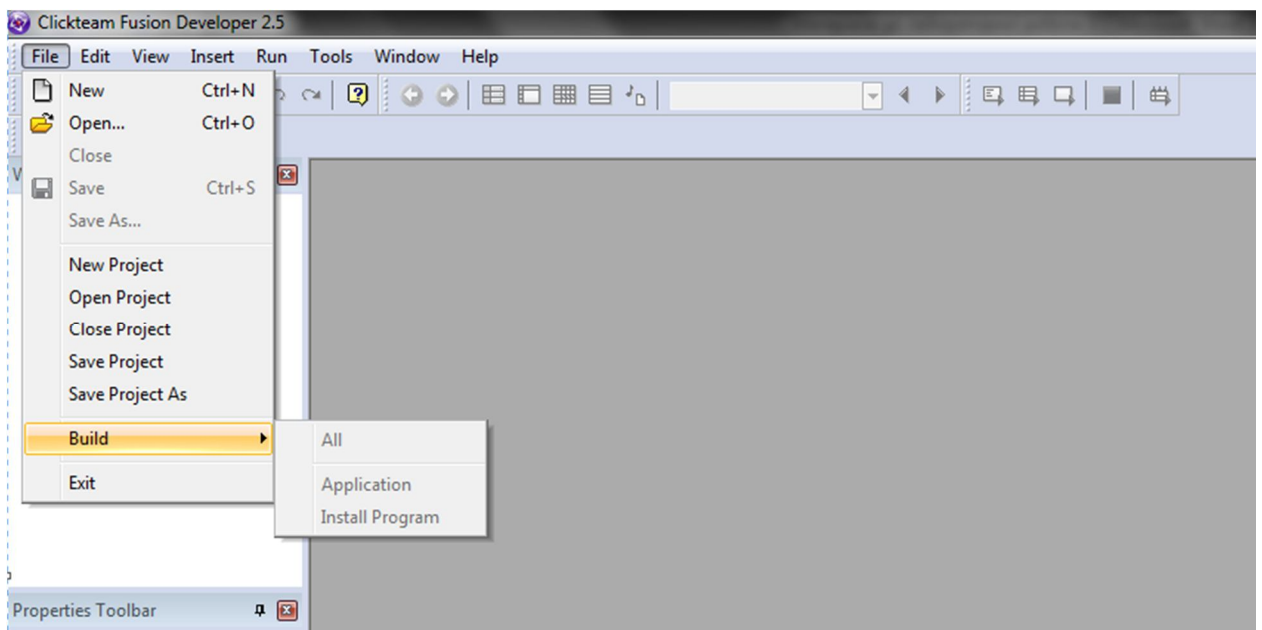


А також у властивостях самого додатку встановлюємо формат гри для зібрання та збереження:

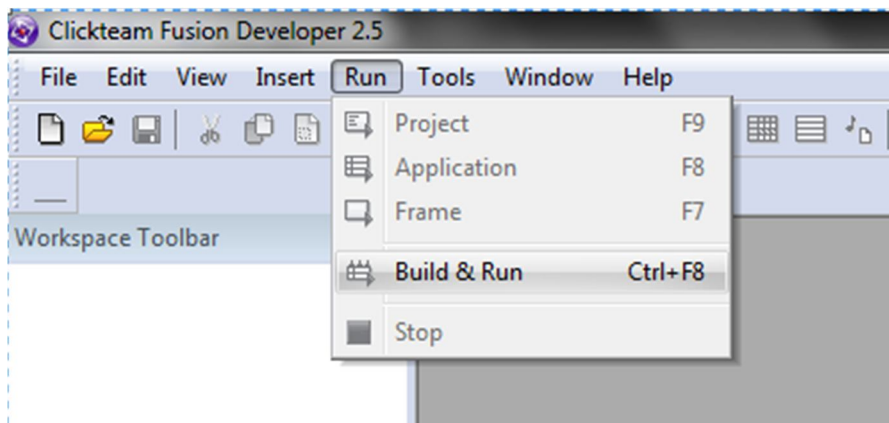


III Прорахунок створеного додатку

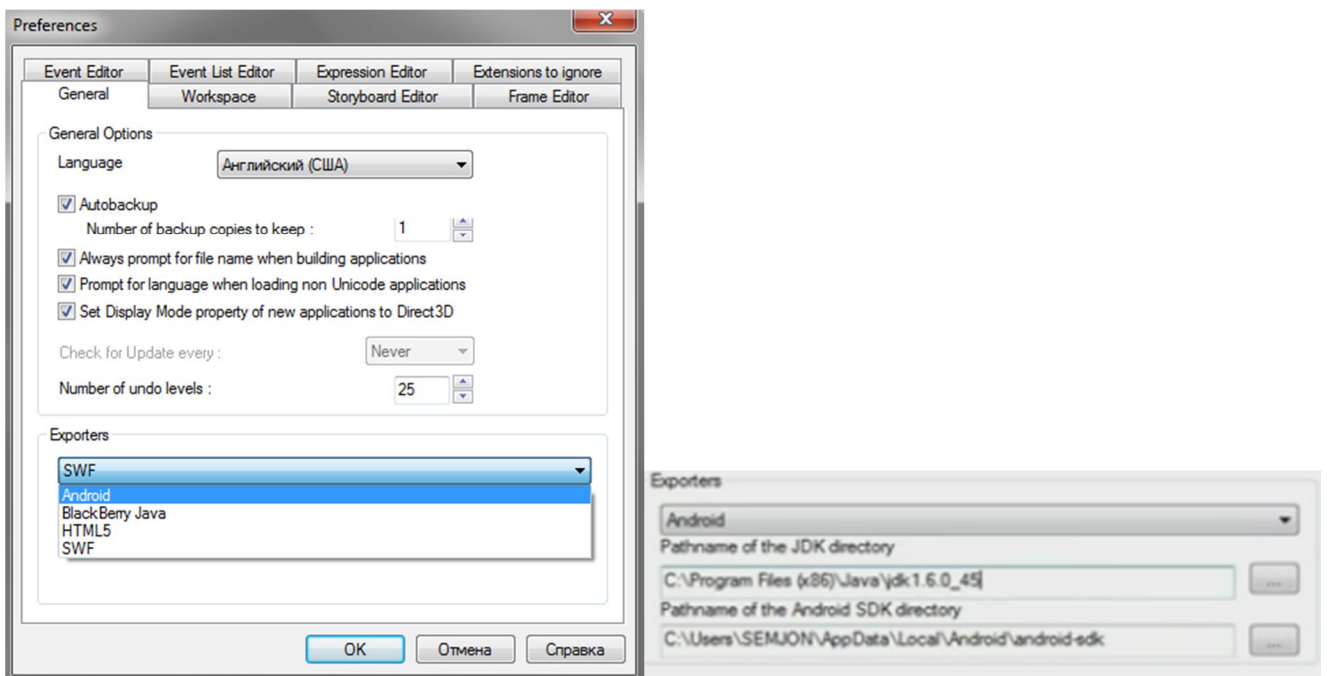
Зібрати гру можна через команду, яка розташована у головному верхньому меню (опція Build)



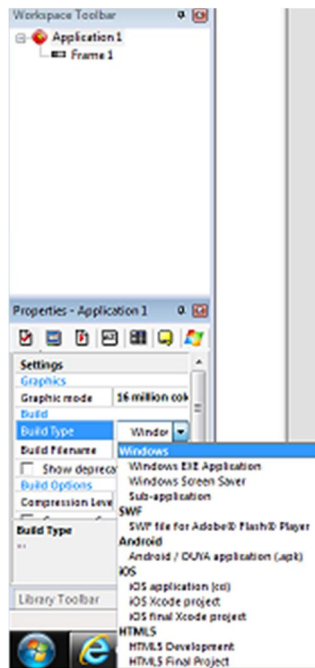
Для того, щоб зібрати створену гру та запустити використовують комбінацію гарячих клавіш ctrl+F8, або використовуючи верхнє меню (підменю Run)



Для вбудованих додатків експорту програми є визначені властивості. Для їх перегляду треба через верхнє головне меню знайти підменю Tools там через рядок Preferences відкрити відповідне вікно, де у вкладці General є відповідне поле:



У вікні властивостей (Properties-Application) після створення проекту можна вибрати в яку платформу буде експортовано створена гра:



Контрольні запитання.

1. Яким чином встановлюється анімація для створеного об'єкту в рамках проекту ?
2. Як можна перевірити анімацію для певного об'єкту при налаштуваннях ?
3. Наведіть стисло алгоритм встановлення зв'язку між окремими об'єктами в проекті.
4. Яким чином встановлюється та контролюється переміщення об'єкту в рамках додатку проекту ?
5. Яке призначення має вкладка Direct options при налаштуванні об'єкту ?
6. Як можна задати циклічну траєкторію руху об'єкту в проекті без органів управління з боку користувача ?
7. Яким чином проводиться рендерінг та перевірка правильності створеного додатку ?

Лабораторна робота №4

Візуальні ефекти при створенні мультимедійного контенту

Мета. Навчитись проводити процедури налаштування 3D-трекінгу для відзнятого відеопотоку, як складової одиниці комп'ютерної графіки при створенні мультимедійного контенту.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Adobe After Effects CC 2017.

Завдання до лабораторної роботи.

1. Створити проект, композицію та імпортувати файл, де були б плоскі частини зображення та контрастні точки (тривалість кліпу до 3 хв).
2. Застосувати ефект трекінгу та провести відсіювання точок.
3. Створити в рамках проекту два додаткових шари – один для контуру об'єкту, інший для забезпечення тіні від цього об'єкту. Додати об'єкт (картинка) в межах створеного контуру площини.
4. Перевірити чи при перемотуванні не випадає контур об'єкту за межі екрану. Якщо така ситуація виникає, то змінити положення об'єкту (опція Position).
5. Самостійно провести налаштування додатково створеного шару та шару Light. Для Track Solid (Material Options) – Cast shadow (бригада 1); Light Transmission (бригада 2); Ambient (бригада 3); Diffuse (бригада 4); Metal (бригада 5). Для елемента Light (Light Options) – Intensity (max) (бригада 1); Color (бригада 2); Shadow darkness (бригада 3); Shadow diffusion (бригада 4); Light Type (бригада 5).
6. Провести візуалізацію кліп зберігши його у форматі .avi.

Теоретичні відомості (основні налаштування)

Алгоритм слідкування (від англ. tracking) або процедура визначення руху (від англ. match moving) при налаштуванні візуальних ефектів на етапі створення аудіовізуального контенту є однією з основних, яка поєднує наявні кінокадри та віртуальну доповнену реальність і дозволяє розробнику додати додаткові елементи

комп'ютерної графіки. З іншого боку, 3D-tracking – це відслідковування траєкторії камери або об'єктів на етапі аналізу відзнятого матеріалу. Результатом такого процесу є отримання набору точок (так званої “хмари точок”), який відповідає статичним об'єктам шота (англ. shot). Тобто, з'являється можливість отримати траєкторію рухомих точок відносно статичної чи динамічної камери (так звані “tracking геометрії”). Ця інформація дозволяє у спеціалізованому програмному забезпеченні виконати заміну певних елементів зображення відеоряду.

Для проведення процедури 3D-трекінгу треба відкрити програму Adobe After Effects (демо-версія), де створити спочатку новий проект і в ньому через вкладку головного верхнього меню створити нову композицію (composition, ctrl+N). Тут слід звернути увагу на налаштування нової композиції в межах котрої буде імпортовано у програму відео кліп, і для якого буде проведено трекінг (рис.1).

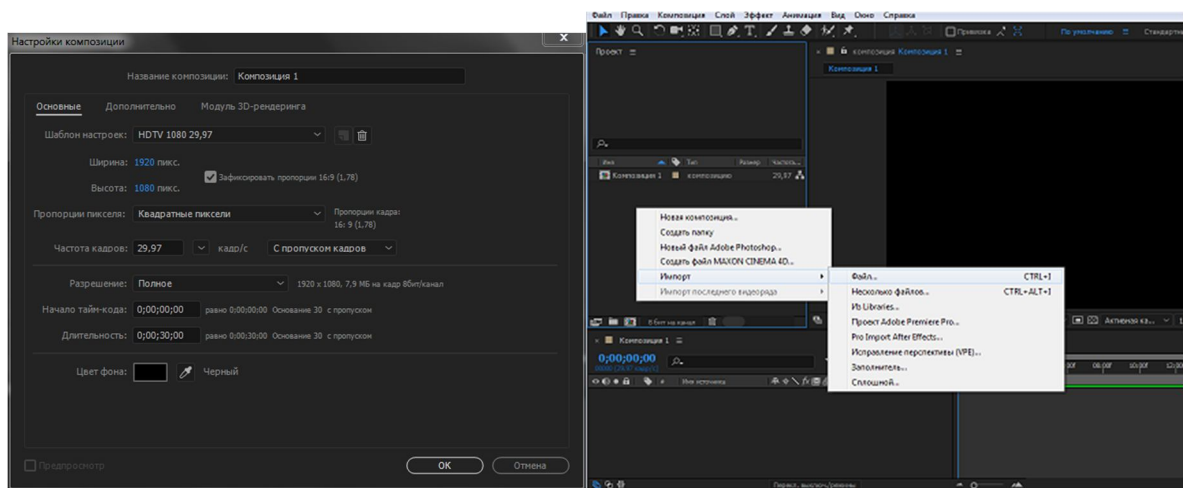


Рисунок 1 – Налаштування композиції та імпортування відеокліпу

Після цього слід імпортований кліп перемістити в область часового коду (timeline). В результаті отримуємо аудіо та відеодорожки (рисунок 2).

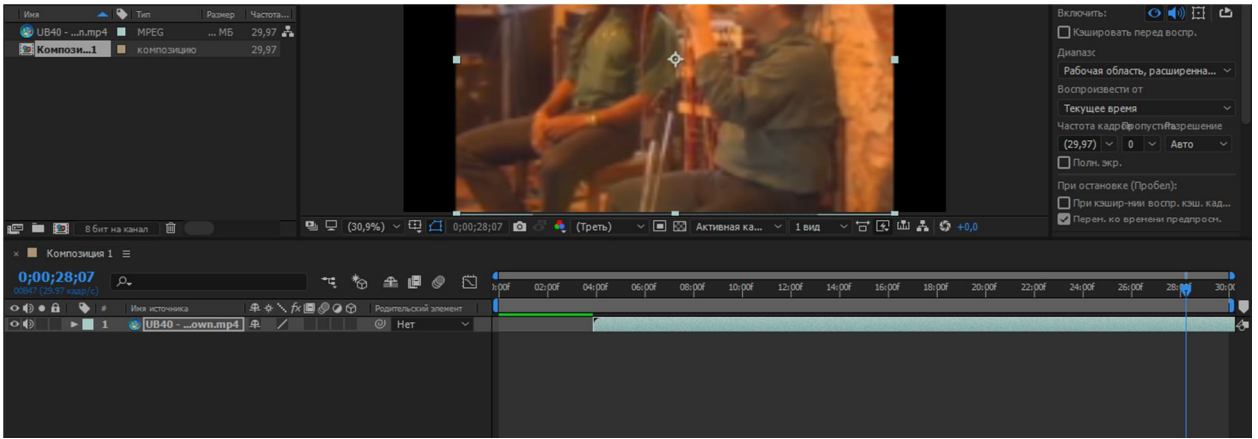


Рисунок 2 – Наповнення композиції контентом

Для реалізації операції слідкування треба з набору вбудованих ефектів програми (вкладка верхнього меню Effects) знайти групу Perspective, де обрати інструмент 3D Camera Tracker (рисунок 3).

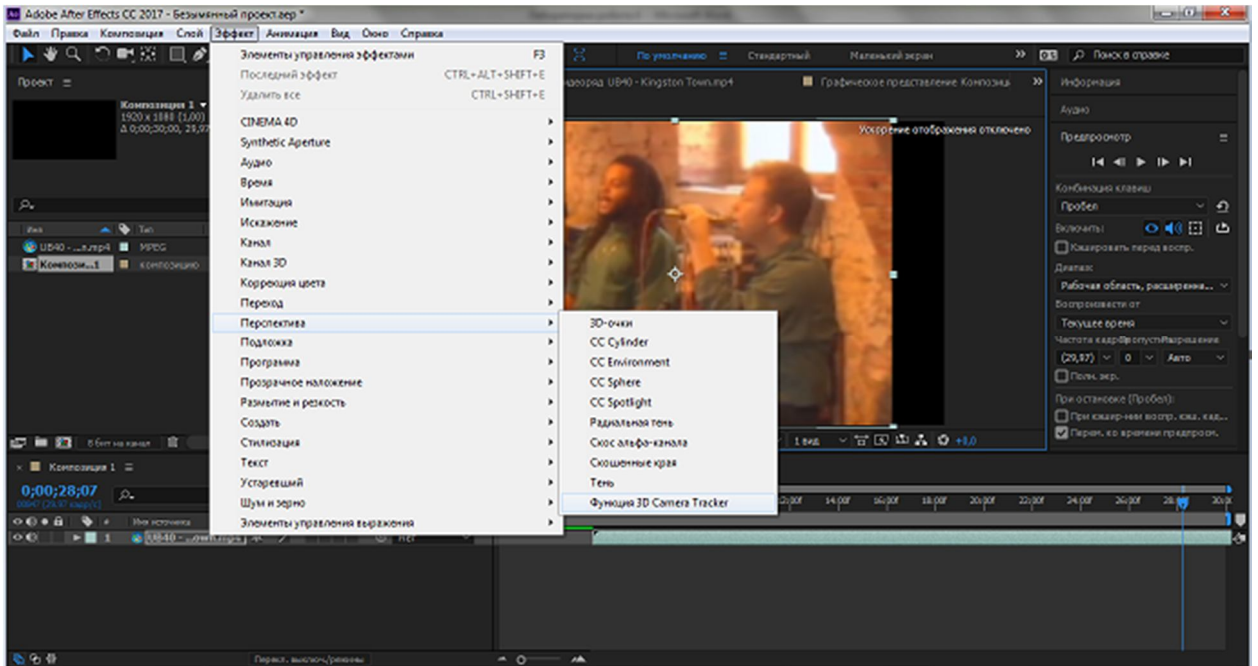


Рисунок 3 – Вибір інструменту

Після цього, з урахуванням застосування ефекту, програма буде шляхом аналізу кадрів створювати так звану “хмару точок”. Ця процедура проводиться 2-ма етапами (рис.4).

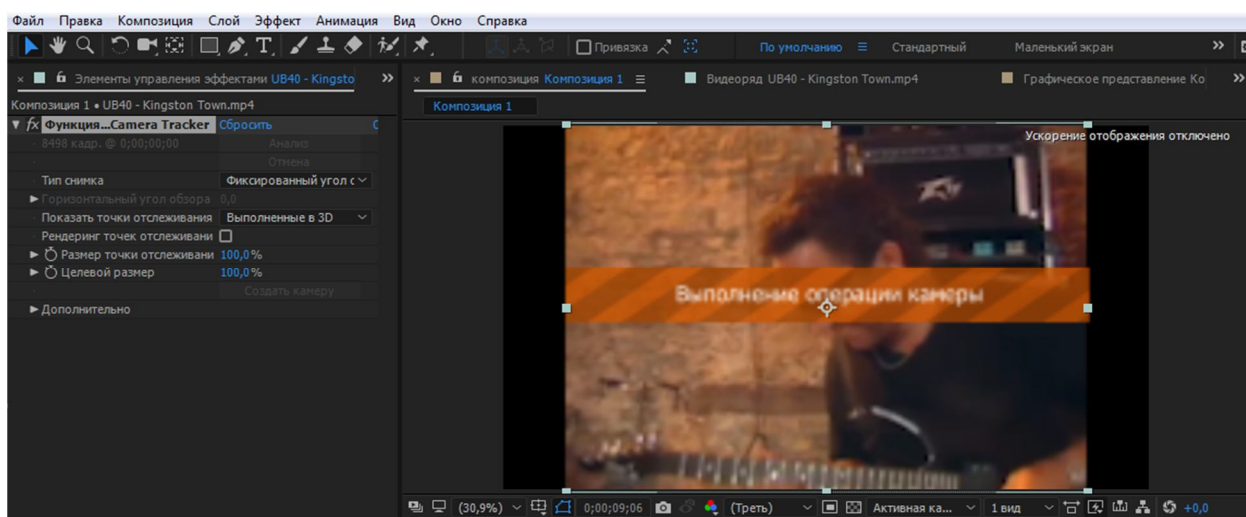
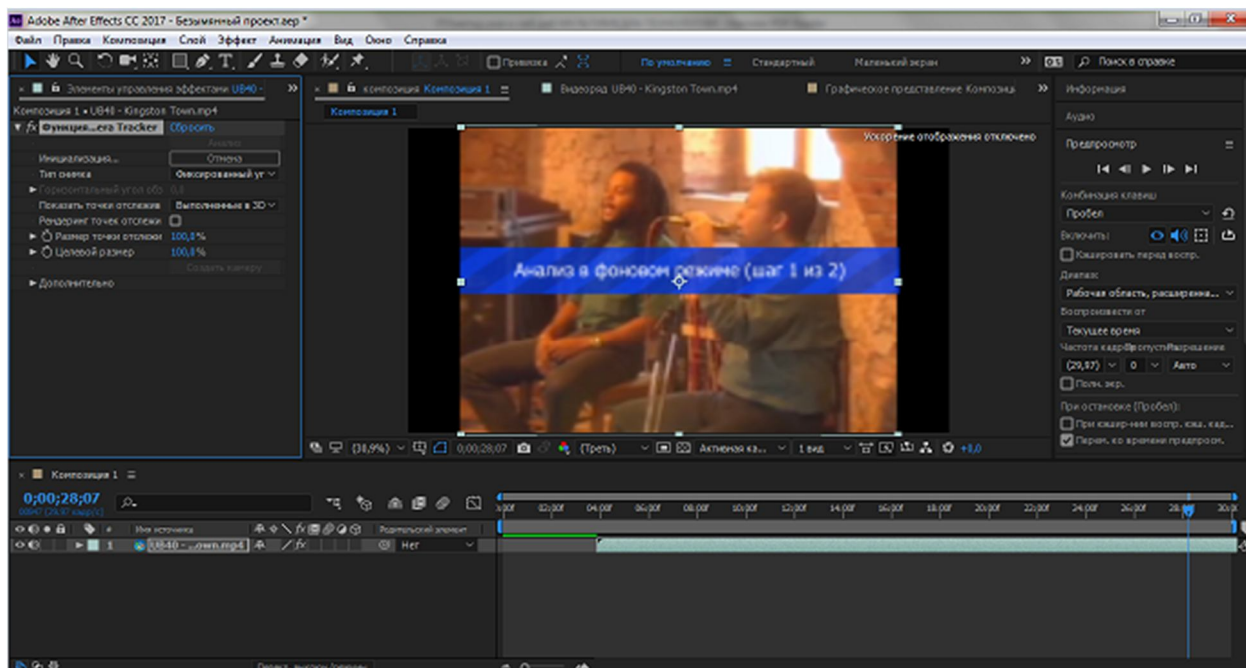


Рисунок 4 – Аналіз кадрів

Врешті-решт можна побачити набір точок, які при перемотуванні будуть зникати а нові будуть з'являтися. В роботі треба обрати (виділити) ті точки, які за “часом” спостерігаються найбільш тривалий час. Вибір цих точок необхідний для того, щоб в рамках утвореної площини вставити в кадр об'єкт. Додатково при виборі точок можна виходити з контрастних точок на кадрі.

Після цього треба відкоригувати положення площини і створити через праву кнопку миші або через верхнє меню ще один шар – тїнь від доданого майбутнього об'єкту (рисунок 5).

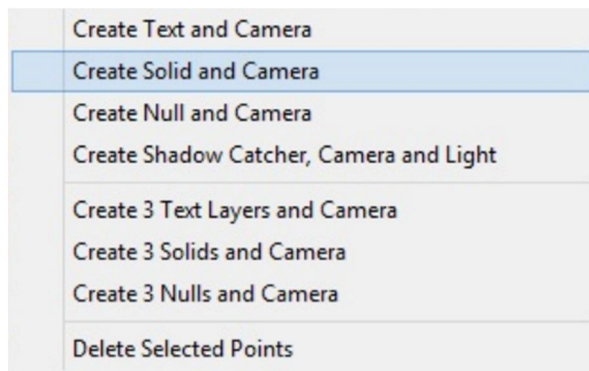


Рисунок 5 – Створення нового шару

Для цього контуру також треба провести корегування площини (рис.6)

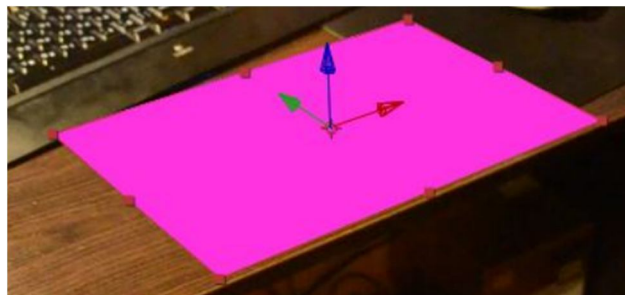


Рисунок 6 – Корегування площини

Далі для контурів площини слід додати рисунок об'єкту в рамках створеного проекту (рисунок 7).

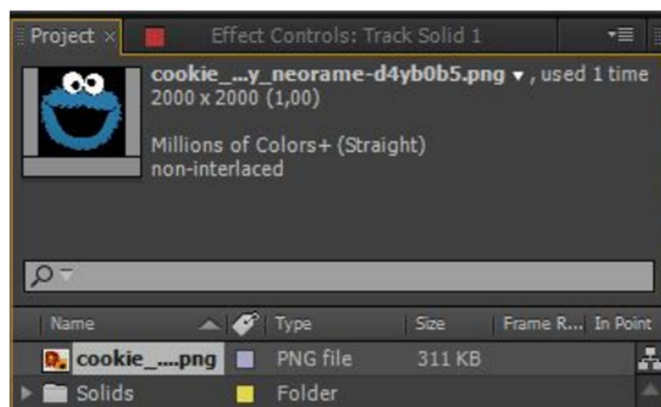


Рисунок 7 – Додавання елемента

Додатковий шар Light необхідно долучити, щоб позначити контур картинки (рисунок 8).

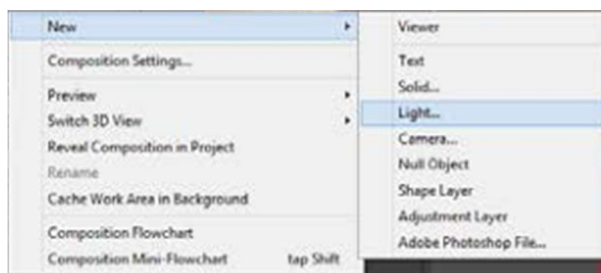


Рисунок 8 – Додавання

В результаті на основі ефекту ми отримали можливість накласти 3D-елементи на двохмірний відео ряд.

Контрольні запитання.

1. Для чого призначений ефект 3D Camera tracker ?
2. Як проводиться матчмувінг в програмі Adobe After Effects ?
3. Для чого призначені шари в рамках проекту і як їх створюють в програмі ?
4. За яким критерієм створюється хмара точок при трекінгу ?
5. Чим відрізняється розмір точки відслідковування від цільового розміру ?
6. Які Ви знаєте режими відображення точок трекінгу ?
7. Як проводять візуалізацію кадрів кліпу в рамках проекту ?

Особливості застосування масок при створенні мультимедійного контенту

Мета. Навчитись створювати маски, як елемент управління об'єктів зйомки для відеоряду, шляхом зміни наповнення альфа-каналу. Вивчення можливостей налаштування елементів маскування в середовищі програми After Effects.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Adobe After Effects CS4 та віртуальні засоби роботи, які подібні до інструментів програми Adobe Photoshop.

Завдання до лабораторної роботи.

1. Створити проект, композицію та імпортувати в область Timeline 2 файли – основний та допоміжний, який буде у динамічному режимі перекривати частину основного відеоряду.
2. Створити маску необхідної форми – бригада №1 - прямокутна з вигнутими вершинами (Convert Vertex Tool); бригада №2 – квадратна; бригада №3 – крива 5 вершин; бригада №4 – еліпс; бригада 5 – багатокутник.
3. Перевірити режим накладання масок для одного шару відеоряду. Визначити яке призначення для маски має опція RotoBezier. Провести суміщення за точністю контуру маски.
4. Додайте до маски ефект відбиття (додатковий шар). Провести його налаштування, зокрема прозорість та суміщення та накладання. Накладання шарів для бригади, його вид, відповідає порядку переліку в програмі.
5. Провести корекцію кольору шару відбиття.
6. Провести візуалізацію (рендерінг) кліпу, зберігши його у форматі .avi.

Теоретичні відомості (основні налаштування)

Маска – контур, який використовується для зміни ефектів та властивостей шару, як окремої складової роботи з зображеннями в програмі. Найпоширенішим застосуванням масок є зміна альфа-каналу шару імпортованого контенту. Зазвичай

маска складається з сегментів та вузлових точок. Сегменти – лінії, які з’єднуються через вузли. Маска може бути відкритим або замкненим контуром. Пряма лінія – приклад відкритого контуру. Натомість, замкнені контури можуть створювати прозорі області для шару. Маска являє собою частину окремого шару. Кожен шар може містити величезну кількість масок. Масками можуть бути стандартні геометричні форми.

Для проведення процедури маскування спочатку треба відкрити програму Adobe After Effects (демо-версія, 30 днів), де створити спочатку новий проект і в ньому через вкладку головного верхнього меню створити нову композицію (composition, ctrl+N). Тут слід звернути увагу на налаштування нової композиції в межах котрої буде імпортовано у програму відео кліпи (рис.1).

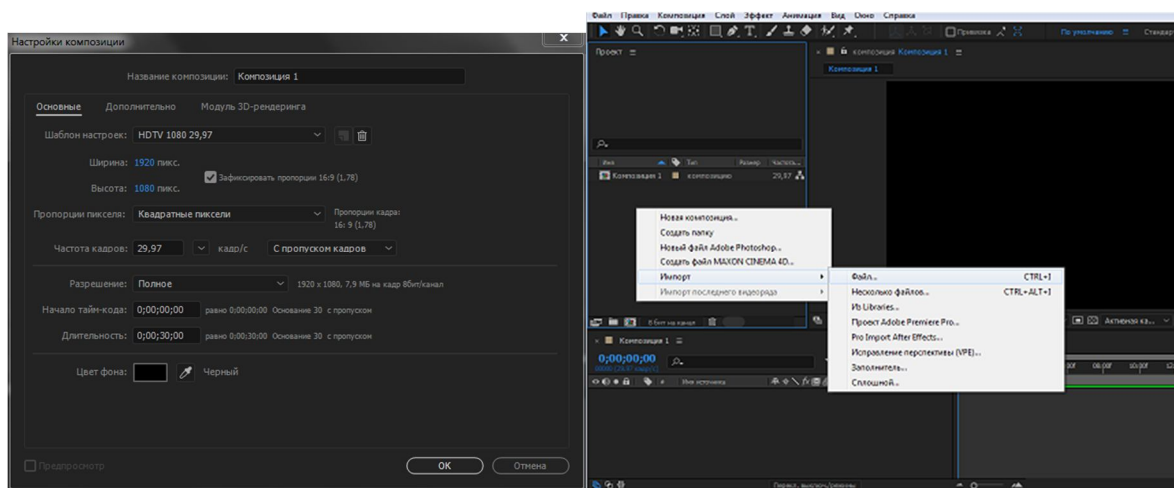


Рисунок 1 – Налаштування композиції та імпортування відеокліпів

Після цього слід імпортовані кліпи (спочатку основний) перемістити в область часового коду (timeline). В результаті отримаємо аудіо та відеодорожки (рисунок 2).

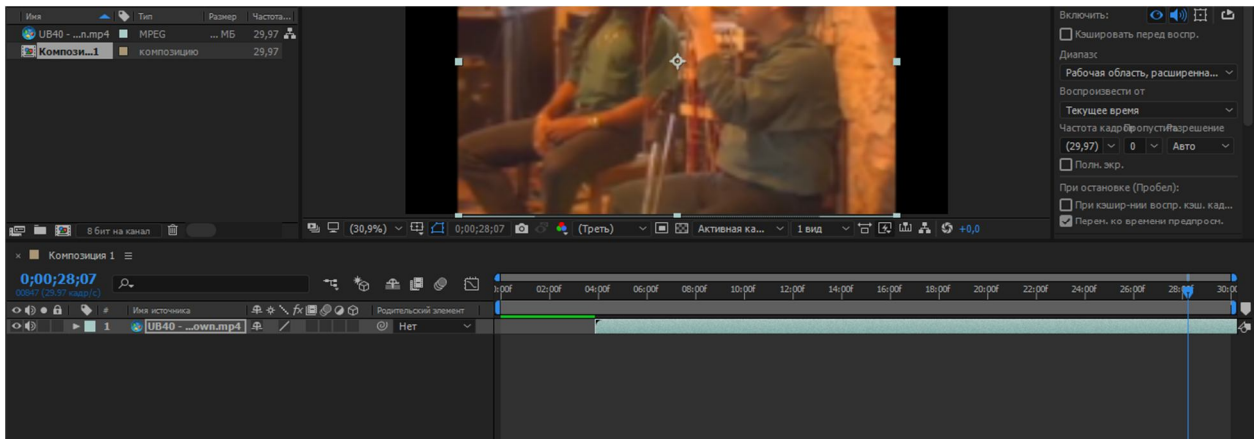


Рисунок 2 – Наповнення композиції контентом

На етапі процедури імпортування відео може з'явитись вікно Interpret Footage, яке говорить про те, що у відео є альфа-канал. Для лабораторної роботи у цьому вікні встановити позначку Ignore.

Після того, як імпортували в область Timeline основний файл треба створити контур маски. Але до цього слід розтягнути зображення до розмірів екрану перегляду – це можна зробити у ручному режимі мишею змістивши межі екрану та зображення (рисунок 3)

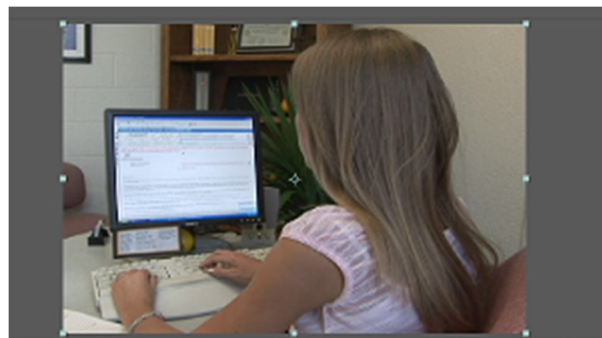


Рисунок 3 – Корегування розмірів

Для створення маски, як елемента геометричної фігури використаємо інструменти програми, а саме елемент пір'я, який знаходиться зверху програми під елементами головного меню. Після створення замкненого контуру, можна помітити, що програма видалила все, що знаходиться за межами цього контуру (чорна область, рисунок 4)

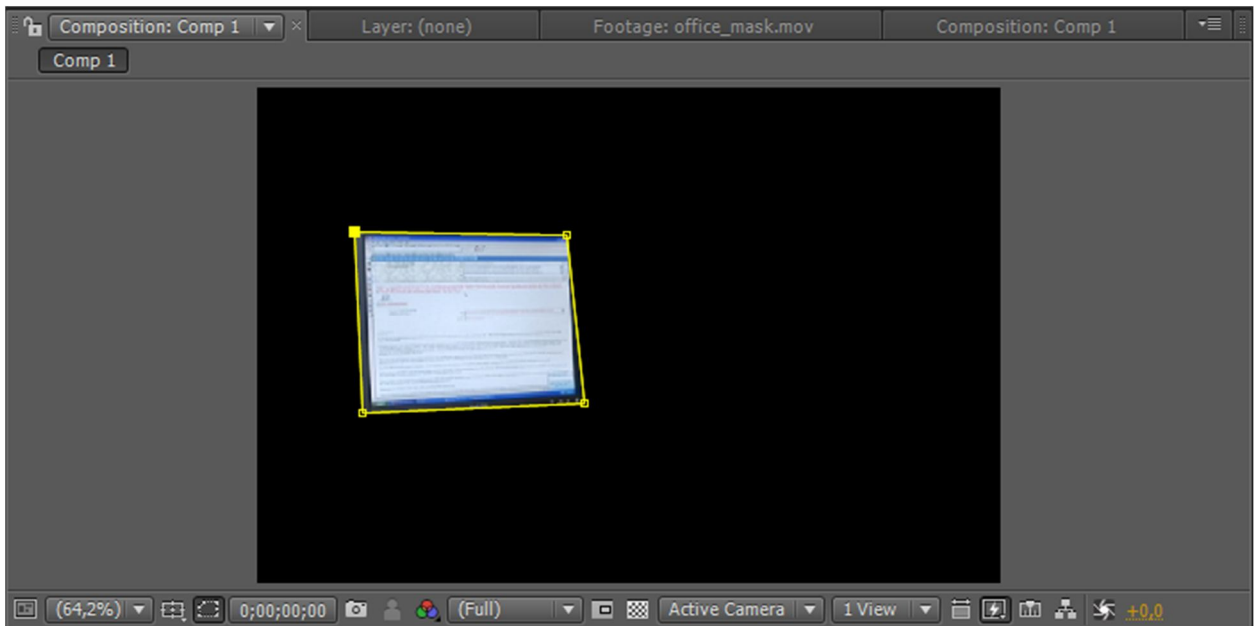


Рисунок 4 – Маска

Для того, щоб виправити таку ситуацію треба інвертувати маску. Для цього треба виділити ряд основного файлу в панелі Timeline і натиснути клавішу M, щоб в композиції відобразити елементи управління маски (зокрема контур маски Mask Path).

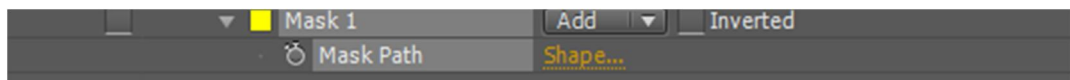


Рисунок 5 – Налаштування маски

І для інвертування маски достатньо просто поставити відмітку напроти пункту Inverted. Варіанти контурів маски в програмі показано на рисунку 6.

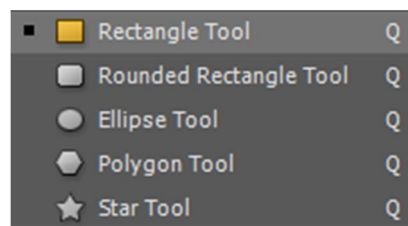


Рисунок 6 – Інструменти креслення

Клавіша F для маски дозволяє встановити межі закруглень вершин маски (Mask Feather).

Далі необхідно замінити вміст маски. Для цього, другий файл – фон маски треба розмістити на окремій дорожці в області Timeline, після чого треба провести корегування розмірів (рисунок 7).



Рисунок 7 – Наповнення маски

Для шару який накладено треба встановити у налаштуваннях композиції значок 3D Layer (рисунок 8)

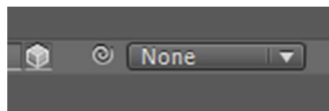


Рисунок 8 – Активація 3D-шару

Наступним кроком буде встановлення ефекту відбиття від зображення маски. Для цього необхідно створити новий шар (рисунок 9), де у властивостях в полі кольору слід виставити білий колір (рисунок 10).

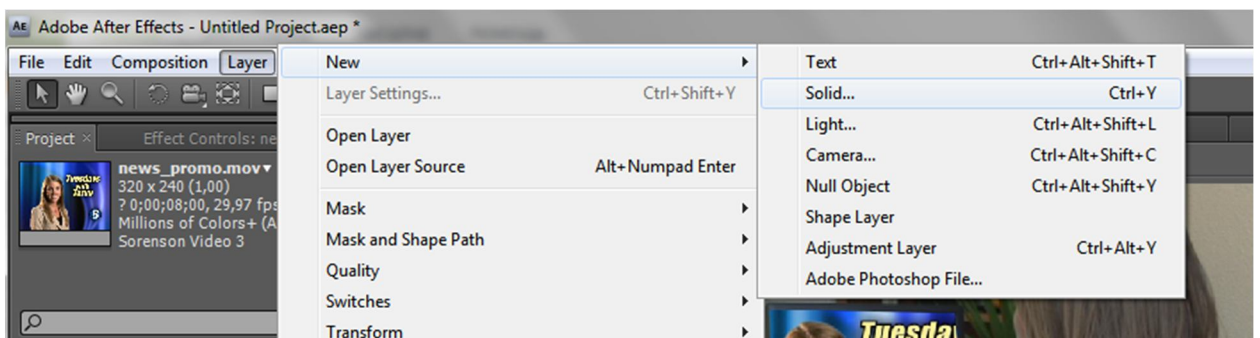


Рисунок 9 – Створення шару відбиття

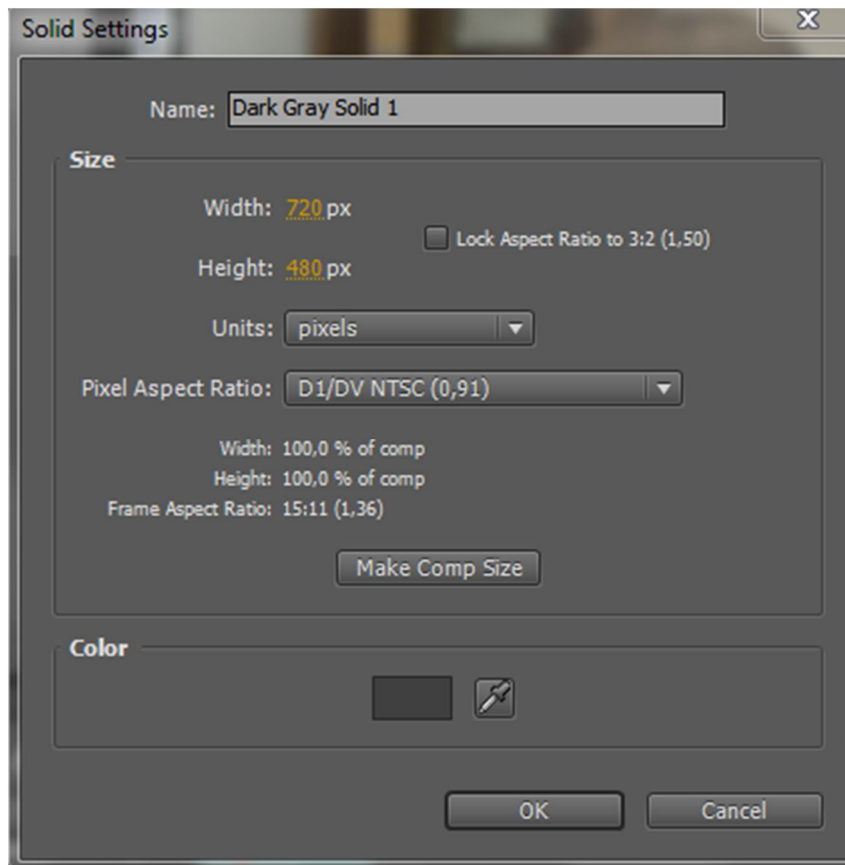


Рисунок 10 – Налаштування шару

Для нового шару простіше за все просто скопіювати розміри маски до розмірів цієї області (у налаштуваннях композиції при виділенні шару слід використати гарячі клавіші копіювання та вставки). І для цього шару відбиття слід зняти позначку *Inverted*. Клавіша *T* для цього шару дозволяє відкрити налаштування щодо прозорості. Тут можна вказати значення у 25%.

Наступним кроком слід провести етап накладання двох шарів з однаковими розмірами і для цього слід у налаштуваннях *Timeline* (права верхня кнопка області) активувати пункт *Modes* (рисунок 11)

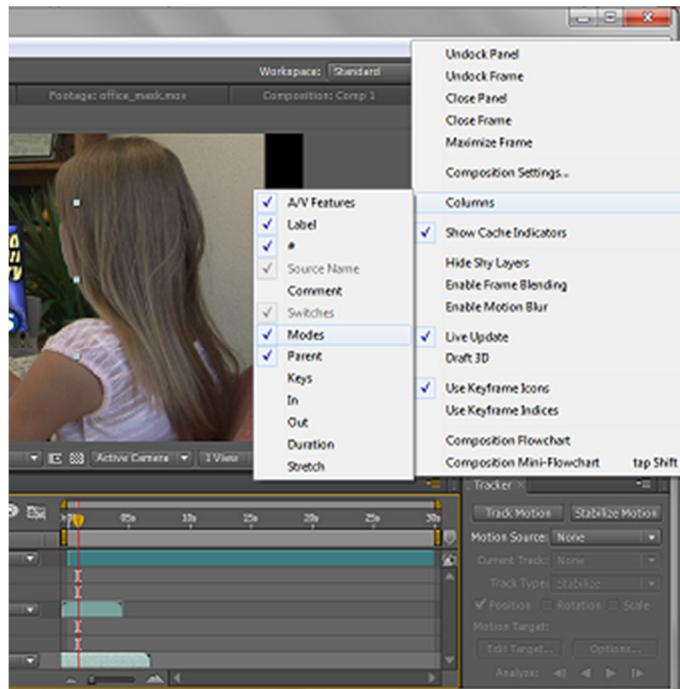


Рисунок 11 – Налаштування суміщення шарів

Результатом цього у панелі інструментів шару з'явиться список режимів суміщення

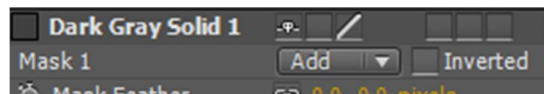


Рисунок 12 – Налаштування

Останнім кроком слід провести візуалізацію створеного кліпу, і для цього у вкладці композиції (верхнє меню) слід вибрати елементи рендерінгу (рисунок 13).

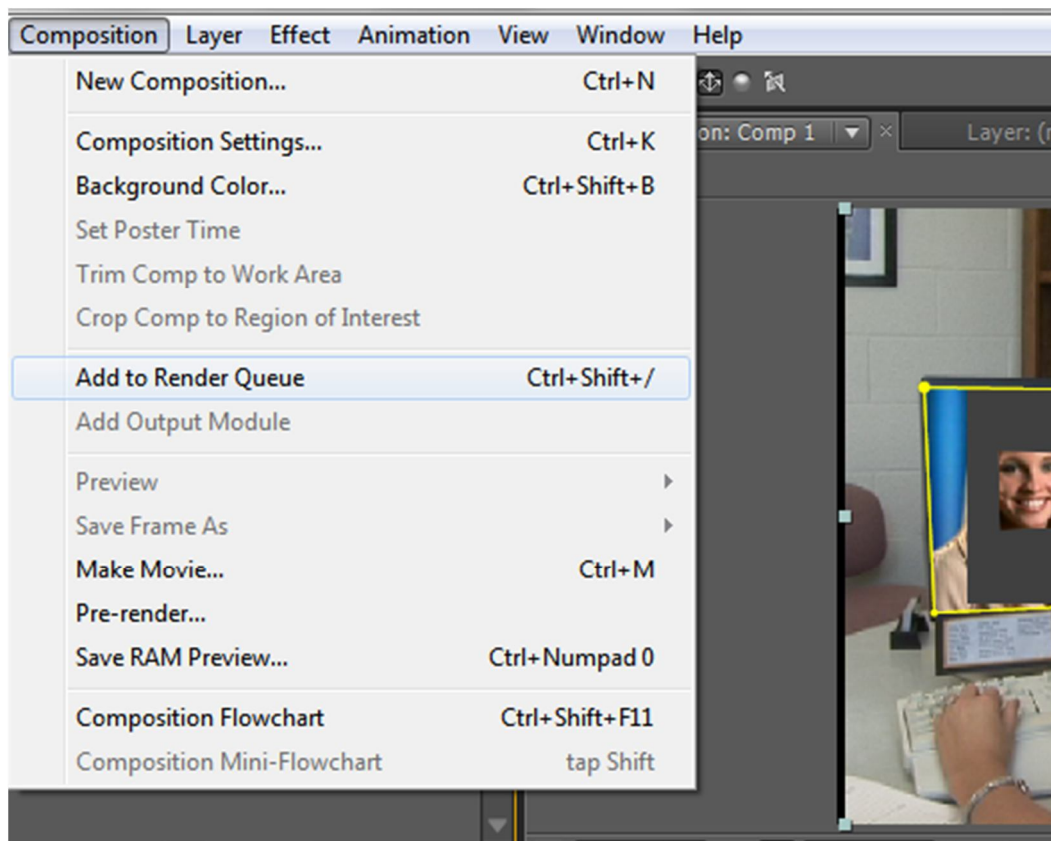


Рисунок 13 – Візуалізація кліпу

Контрольні запитання.

1. З якою метою у відео файлах використовується альфа-канал ?
2. Яке призначення масок при обробці відео файлів ?
3. Які Ви знаєте параметри налаштування шару ?
4. У чому різниця між відкритою та замкненою формами маски ?
5. Як можна провести процедуру “пом’якшення” маски ?
6. Які недоліки має режим суміщення шарів в програмі After Effects ?
7. Як створюється динамізм картинки при налаштування маски ?

Лабораторна робота №6

Засоби ротоскопінгу та планарного трекування при обробці мультимедійного контенту

Мета. Навчитись проводити планарне трекування при роботі з контентом, вміти аналізувати планарне відслідковування та знати за яким алгоритмом проводиться програмне відслідковування окремих частин відеоряду.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення MochaPro 4.0.0 від Imagineer Systems або відповідний плагін в інструментах програми Adobe After Effects (Animation-Tracking mocha AE). Програма Sony Vegas Pro для створення секвенції кадрів відеоряду.

Завдання до лабораторної роботи.

1. Створити з обраного відео кліпу послідовність кадрів на основі залучення інструментів програми Sony Vegas Pro або іншої подібної.

2. Створити проект та імпортувати в програму ротоскопінгу MochaPro. Перевірити та змінити розміри частоти кадрів, пікселю в імпортованому кліпі.

Тривалість кліпу – номер бригади +10 сек (для другої групи +15 сек). Кількість планарних площин відповідає номеру бригади.

3. Провести алгоритм трекінгу, виконати алгоритм вирізання шарів (де є перетин). Додати в обрану планарну площину новий об'єкт і провести його налаштування.

4. Виконати рендерінг та досилання результатів в програму Adobe Premiere.

Теоретичні відомості (основні налаштування)

1. Створення секвенції кадрів на прикладі програми Sony Vegas Pro. Після завантаження кліпа в програмі з'явиться уточнююче вікно (рис.1)

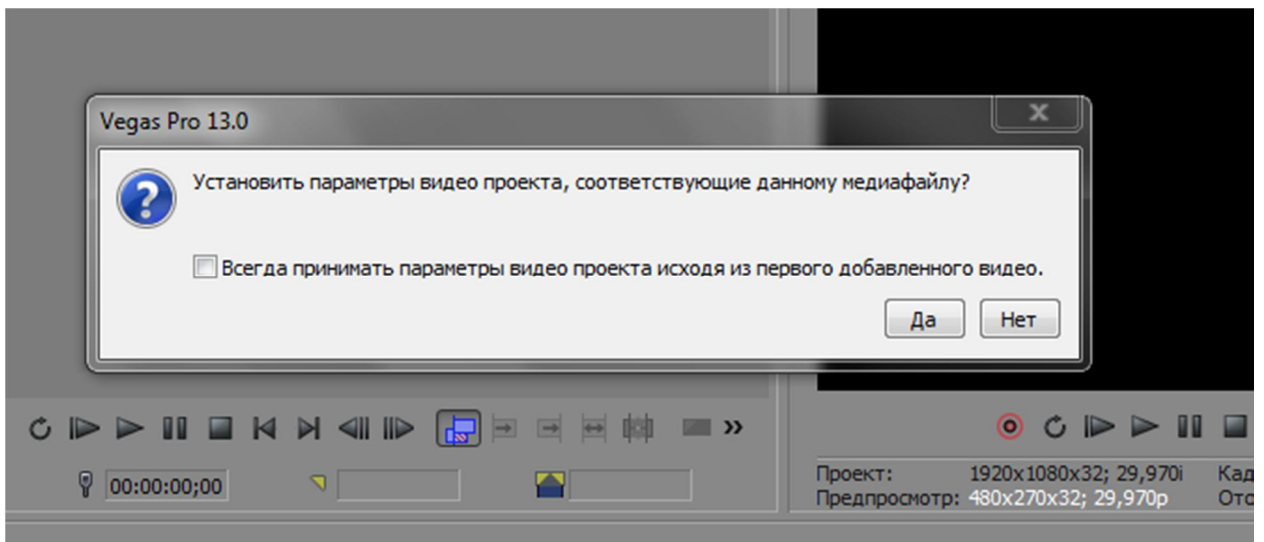


Рисунок 1 – Налаштування проекту

Після згоди на параметри відео проекту отримаємо інтерфейс програми за часовими дорожками (рис.2) :

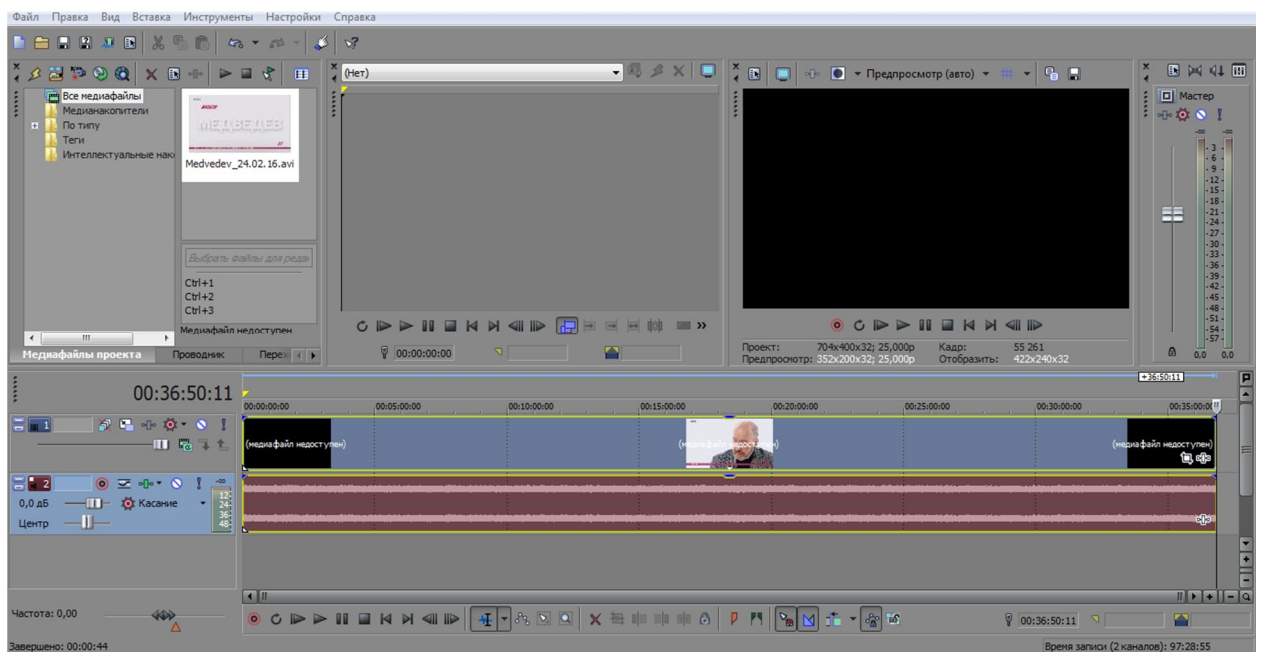


Рисунок 2 – Робоча область програми Sony Vegas Pro

Далі слід для створення секвенції кадрів – обрати та перевести курсор для здійснення операції підрізання файлу для вибору заданої тривалості (початок та кінець підрізки - $Alt+[$ та $Alt+]$)

Після цього, слід провести рендерінг у вигляді створення набору послідовності. Верхнє меню – File/Render As (рис.3):

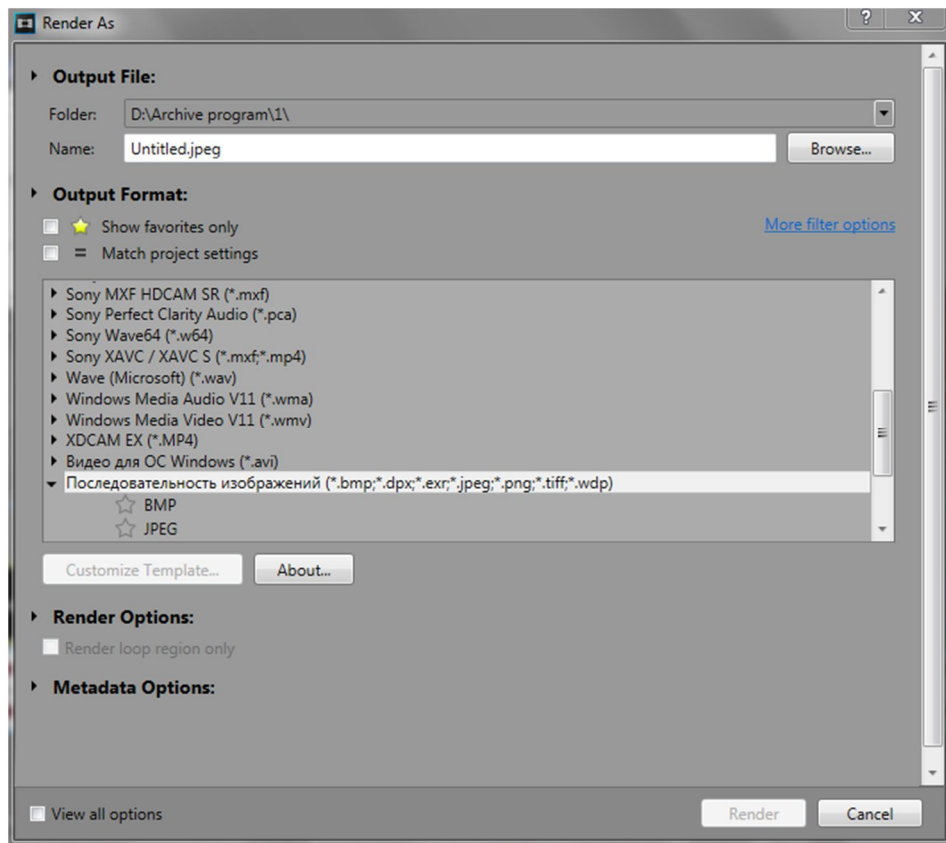


Рисунок 3 – Рендерінг

Обравши місце розташування вихідної мапи натискаємо кнопку “Render”.

2. Відкриваємо програму ротоскопінгу Mocha Pro 4.0 (рис.4).

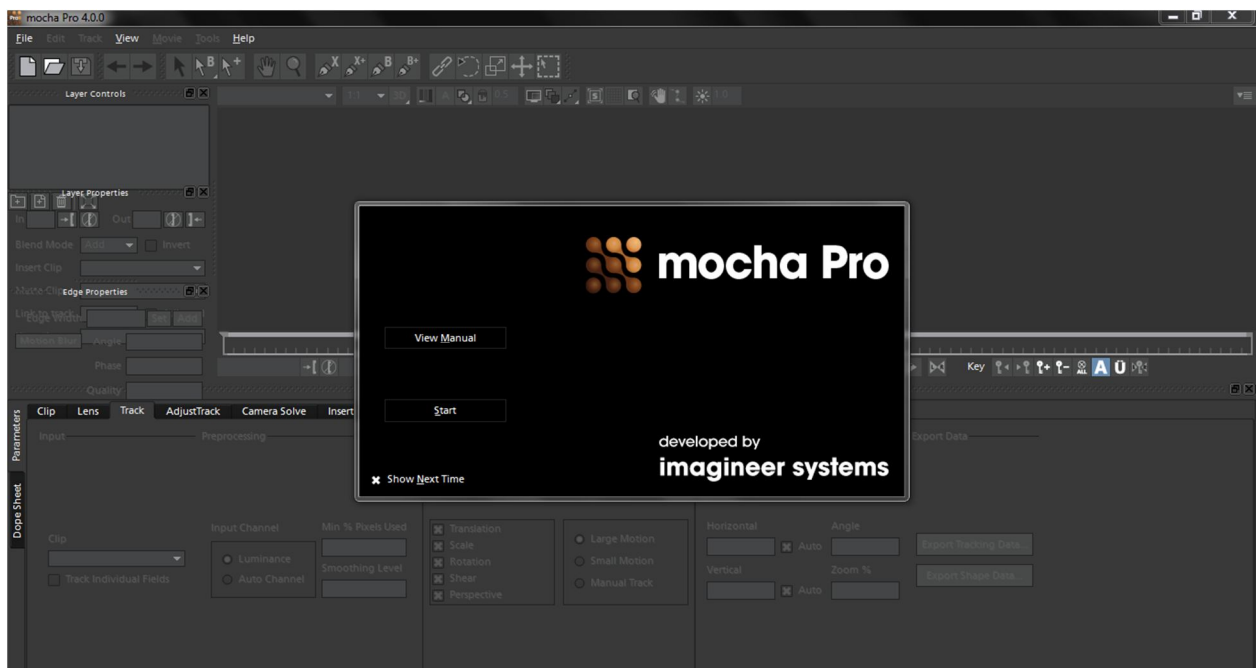


Рисунок 4 – Робоча площа програми ротоскопінгу

3. Через верхнє меню у вкладці File вибираємо рядок New Project і отримуємо вікно, де через кнопку Import Clip вказуємо мапу, де знаходиться секвенція кадрів кліпу.

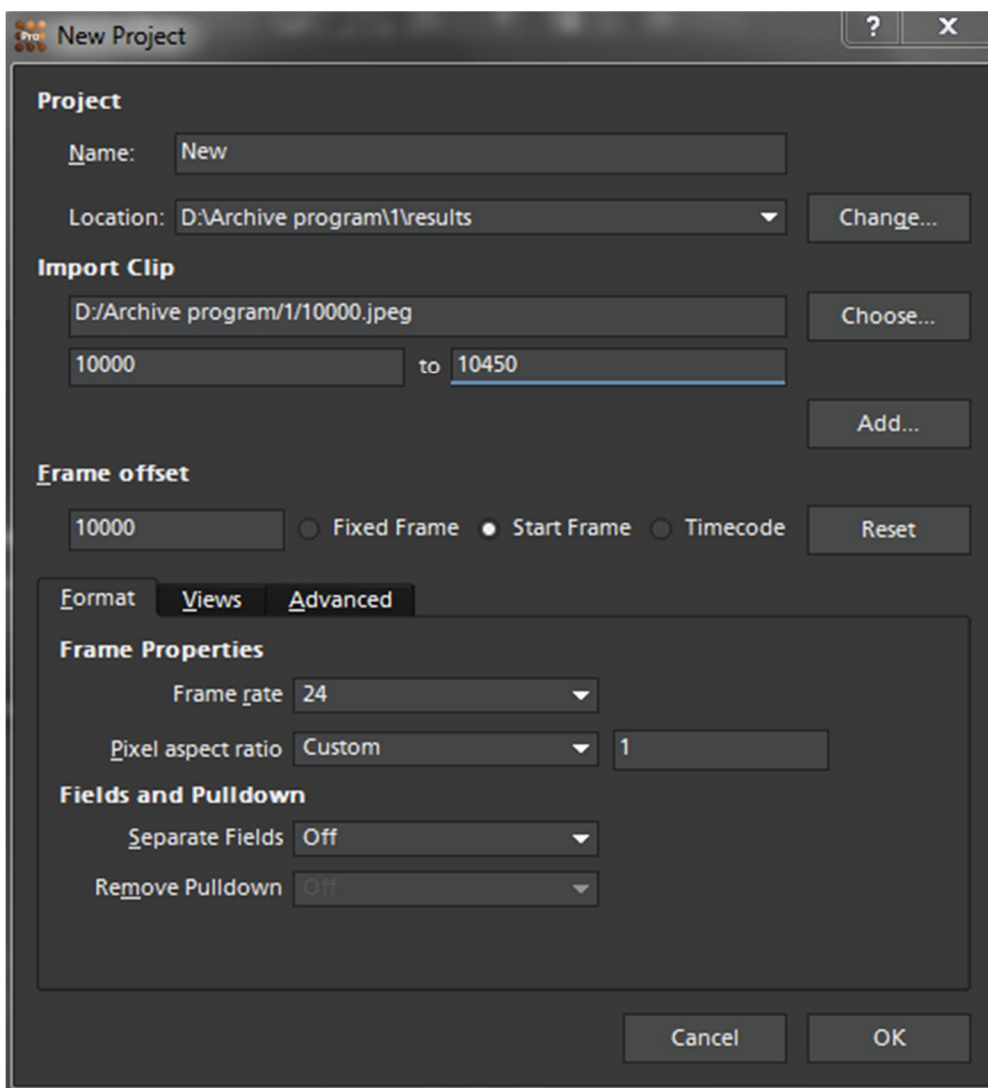
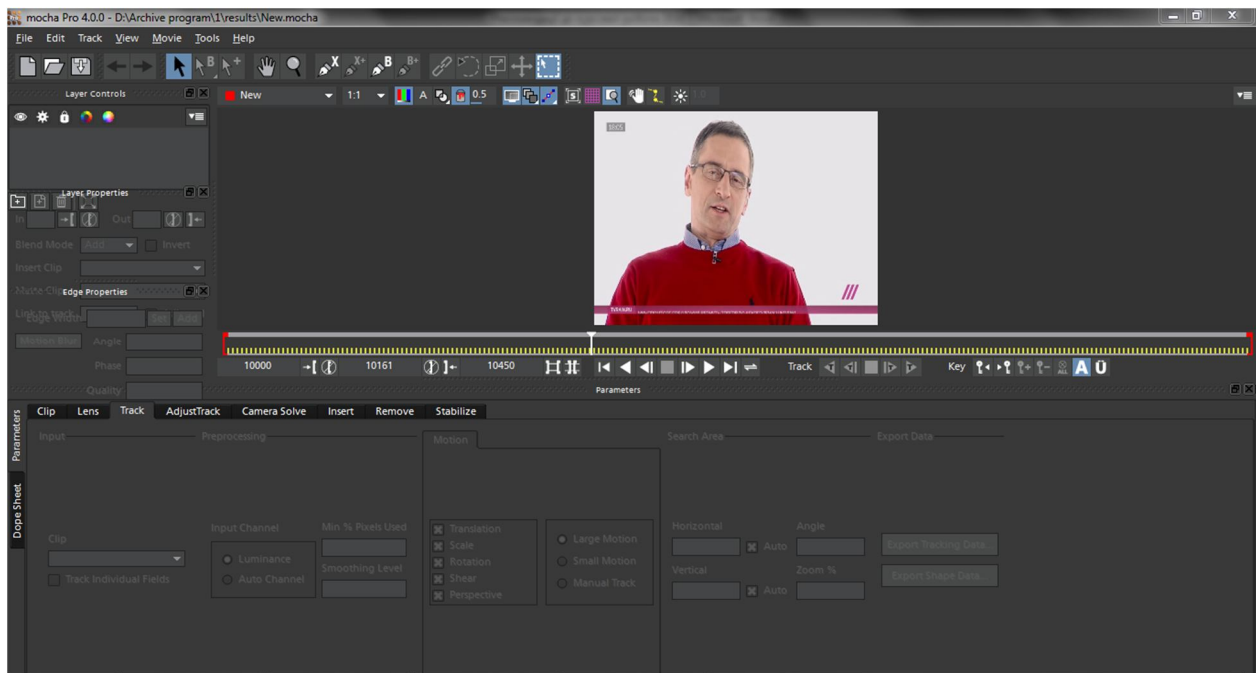


Рисунок 5 – Параметри проекту

Це є фактично вступний етап ротоскопінгу. Основне, що треба звернути в отриманому вікні, це параметр Frame rate – відео формат- 25 FPS (кіно матеріал – 24 FPS). У програмі, куди будемо досилати отримані результати, це значення повинно бути таким самим. Далі розмір пікселю 1.09. І після цього натискаємо кнопку ОК. В результаті отримуємо (рисунок 6).



І маємо ситуацію з перекриттям

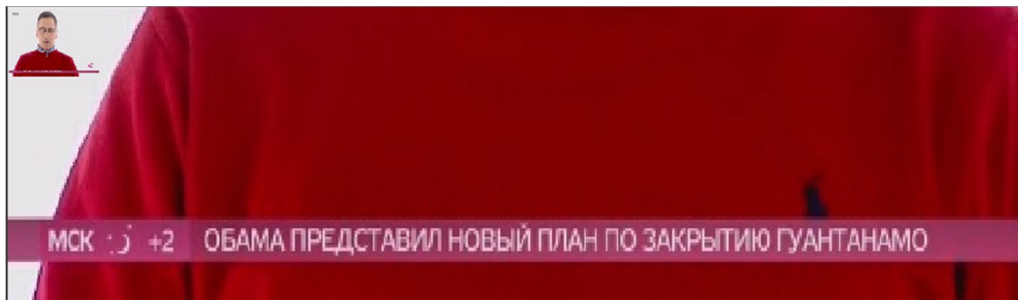


Рисунок 6 – Перекриття частин ображення

4. Далі, слід нарисувати Shape перетин для трекінгу програми. Тобто візьмемо проведемо через кнопку X-Spline Layer Tool у верхньому меню програми виділити область (ротоскоп області – шар, який відповідає цій області). Можна обрати і інші інструменти. Знак X+ - додати ще один шар до створеного (приєднання).



Для цього можна через інструмент масштабу – слід збільшити область кадру (при обраному інструменті слід мишею переміщати по лінії при натиснутій лівій кнопці миші).



Таким чином, отримаємо область трекінгу (рисунок 7)

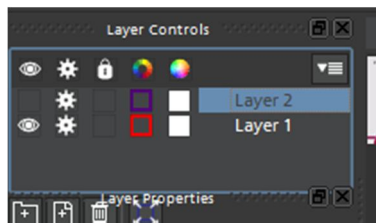
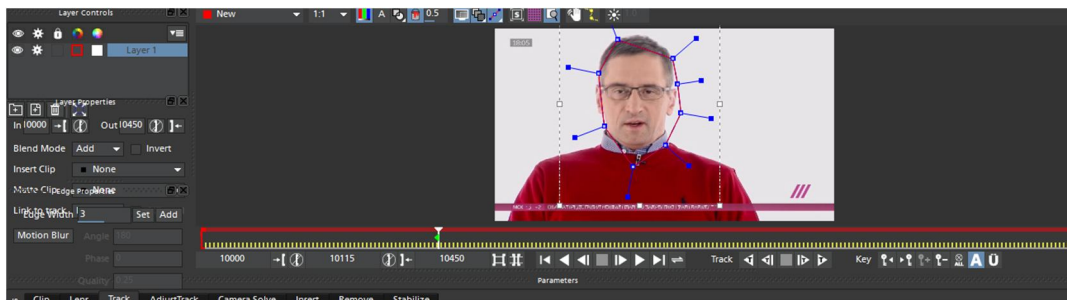
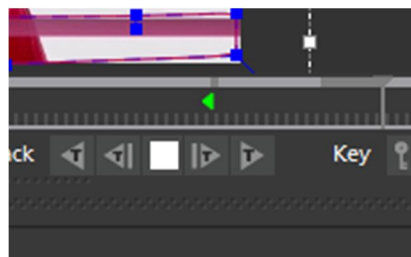


Рисунок 7 – Управління шарами

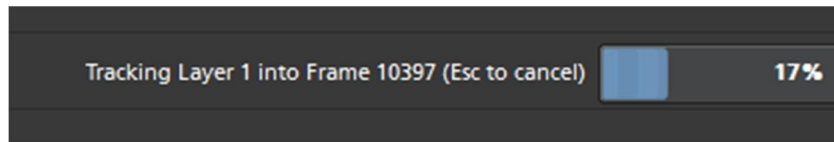
Виріжемо рядок:



Операцію трекінгу виконують через кнопки до першого, останнього кадру:



При цьому трекінг буде проходити через область у крайньому правому верхньому меню програми



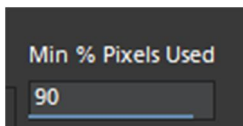
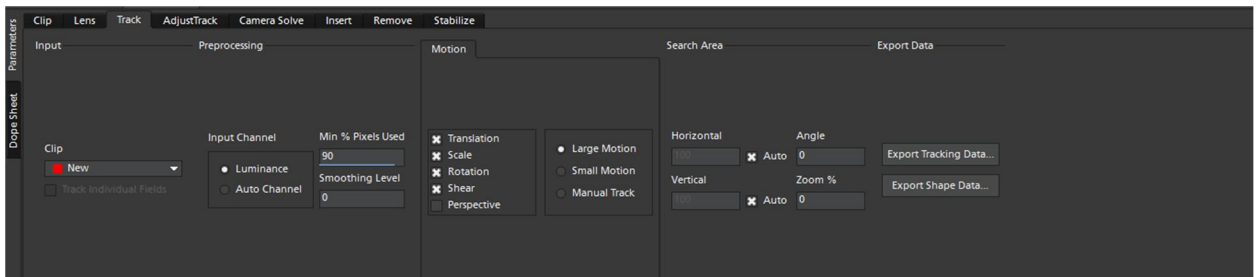
Результатом цих дій буде синя смужка



Перевірити вирізання слів через меню у верхній частині інтерфейса програми

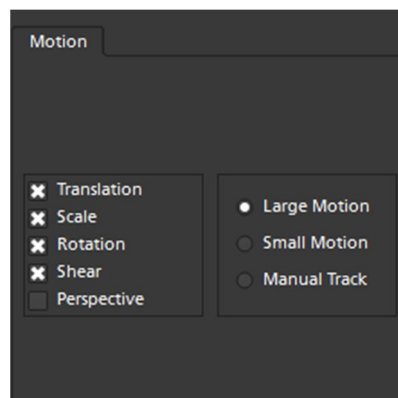


5. Розглянемо налаштування панелі трекінгу програми

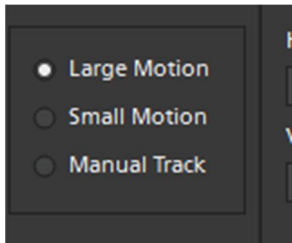


– мінімальна кількість пікселів, які використовуються для трекінгу. Цей параметр залежить від області трекінгу і визначається програмою автоматично. Це значення впливає на якість трекінгу.

Далі область



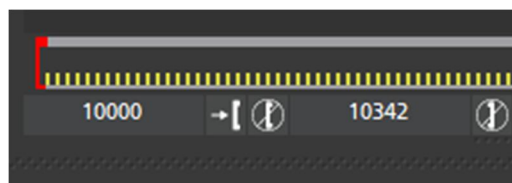
Translation- об'єкт переміщується зліва або вправо; Scale – при переміщенні цей об'єкт змінює свої розміри; Rotation – ще й обертається; Shear – випадок, коли один з кутів об'єкту зміщується; Perspective – область трекінгу (планарна площина) змінюється у перспективі при русі (наближається, наприклад). Далі



- не ціла кількість пікселів, які переміщуються. Large Motion – звичайне зміщення.

Результат трекінгу – зайвий шар не впливає на основний шар об'єкту.

6. Експорт результатів у зовнішню програму – Nuke, Adobe After Effects. Спочатку слід перевірити номери кадрів кліпів, які використовуються при трекінгу.



Для цього відкриваємо вкладку Clip (рис.8)

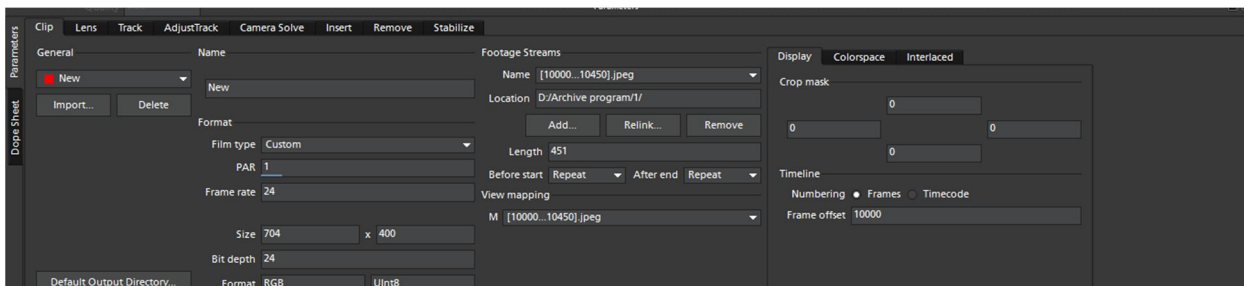
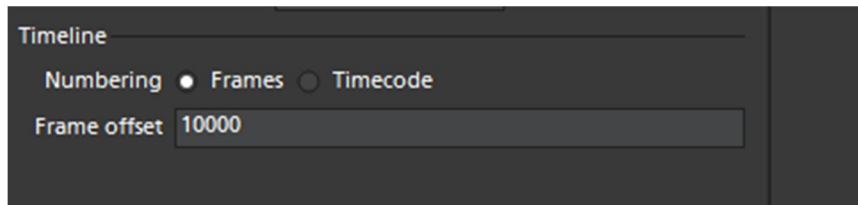
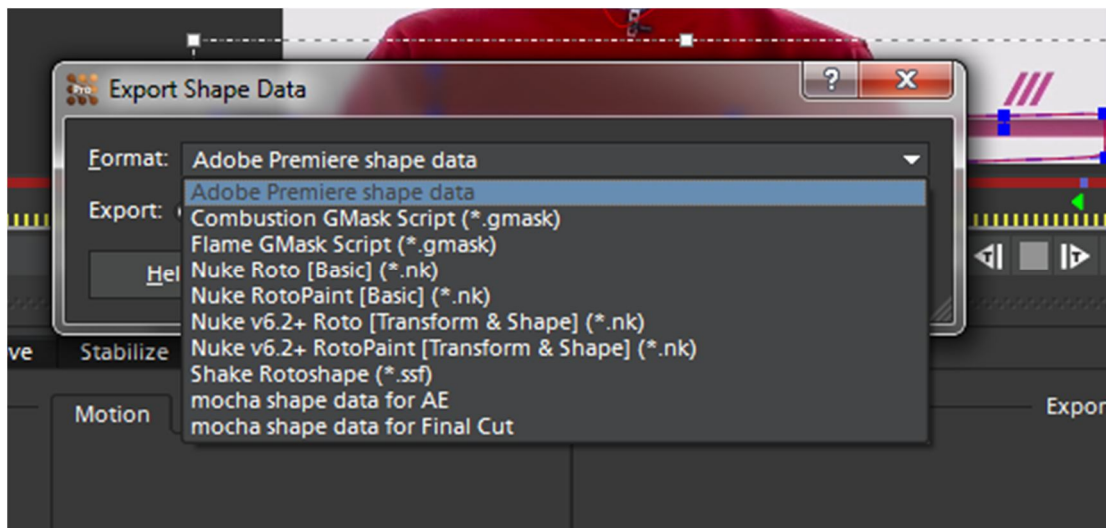


Рисунок 8 – Параметри кліпу

Тут встановлюємо номер кадру (нульовий кадр трекінгу)




Далі вибираємо шар і у вкладці Track натискаємо кнопку “Export Shape Data...”. Отримаємо вікно:



І зберігаємо у скрипті. На цьому трекінг проведено.

Планарні площини – області кадру які відрізняються від загальному плану кадру (точки шарів не знаходяться в одній площині).



Інструмент  - робота операції з групою точок для трекінгу.

Контрольні запитання.

1. Сформулюйте основне призначення трекінгу.
2. Що таке планарність в ротоскопінгу ?
3. Яким чином можна перевірити параметри проекту та кліпу в програмі Mocha Pro ?
4. Які недоліки планарного трекінгу в програмі MochaPro ?
5. Які Ви знаєте інструменти роботи з точками трекінгу ?
6. У яких випадках автоматичний трекінг може замінити ручний режим для зображення високої чіткості ?
7. Як впливає кількість планарних площин на якість трекінгу в програмі MochaPro ?

Лабораторна робота №7

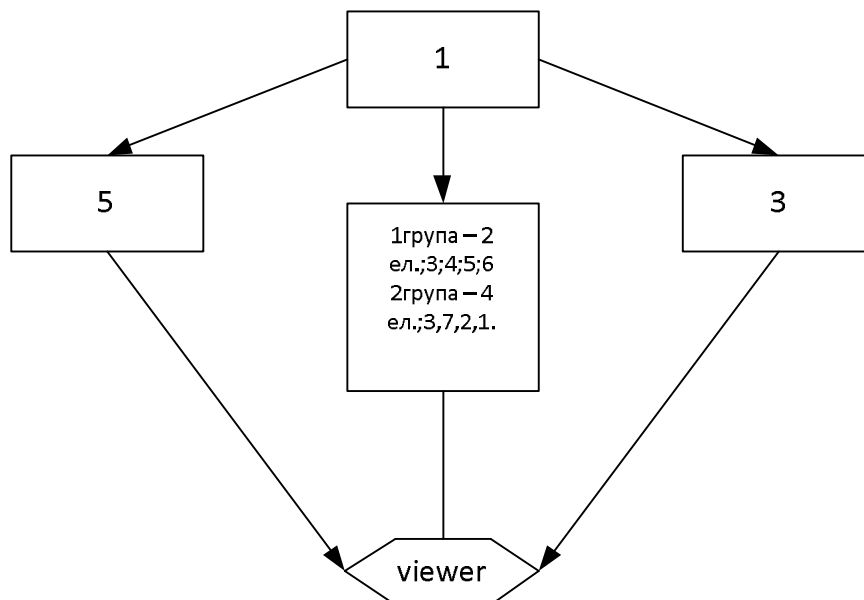
Основи роботи в програмі Nuke для обробки різнопланового мультимедійного контенту

Мета. На основі правил та процедур програми Nuke вивчити основні інструменти програми, які дозволяють користувачу виконувати основні технічні процедури з різноплановим контентом. Навчитись проводити організацію нодової структури проекту та вміти проводити композітинг (compositing) елементів аудіовізуального контенту на основі використання альфа-каналу.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Nuke.

Завдання до лабораторної роботи.

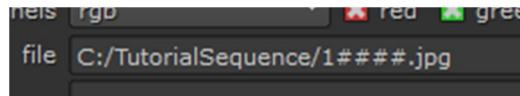
1. Створити проект, де завантажити: файл-зображення; розкадровану послідовність відеокадрів.
2. До файла-зображення долучити наступні ефекти: розмиття, хвиля, кольорова корекція
3. Створити наступну блок-схему архітектури проекту роботи з мультимедійним контентом:



4. У створеній нодовій схемі забезпечити: дублювання (1 група); клонування (2 група).
5. Забезпечити у створеній структурі “1 коліно”.

6. Один з нодів відключити в структурі, а інший вилучити зі схеми не руйнуючи зв'язки.

7. За допомогою ноди Write та інструменту рендерінгу ноди провести візуалізацію створеного матеріалу за усіма кадрами за основною лінією схеми зі збереженим лічильником кадрів (xx_####.jpg).



8. Перевірити та зафіксувати у протоколі чи мають створені ноди альфа-канал (канал прозорості).

Теоретичні відомості (основні налаштування)

Програма Nuke побудована таким чином, що усі елементи програми співпрацюють з шарами і визначаються вузловою, або нодовою структурою. Програма була створена відомої світі комп'ютерної графіки компанією Digital Domain і наразі розробкою версій цього продукту займається компанія The Foundry. Програма підтримує створення власних плагінів на основі мови C++ і її компоненти можна змінювати за допомогою мови Python (див. лабораторну роботу 8). Розробка компанії The Foundry Nuke X відрізняється від базової модифікації тим, що у версії Nuke X є можливість налаштування планарного трекінгу (3D-трекінгу).

Основне меню програми складається з 6 елементів (рис.1):

- File – стандартне меню для роботи з проектами (композиціями). Відмінною рисою меню є додаткові можливості для завантаження скриптів;
- Edit – меню для проведення операцій з вставки, копіювання, вирізання, пошуку виділення елементів. Особливістю цього меню є насамперед наявність рядків виклику властивостей програми та налаштувань проекту (Project Settings);
- Workspace – різні режими роботи програми з контентом (стара назва Layout визначає призначення меню – різні режими роботи з шарами);
- Viewer – вікно відображення програми (екран або монітор програми);

- Render – інструменти візуалізації створеного матеріалу проекту;
- Cache - інформація про залучення кешу робочої станції;
- Help – виклик файлів-довідки та реєстрація програми.

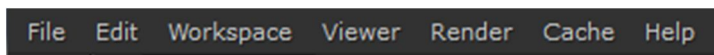


Рисунок 1 – Панель головного меню програми

Задля забезпечення зручності роботи з програмою у структурі виведено окрему панель інструментів, де розміщено усі основні елементи програми Nuke і які поділено на групи. Ця вертикальна панель знаходиться з лівого боку робочого поля програми (рис.2)



Рисунок 2 – Панель інструментів програми

Слід відмітити, що основна частина програми Nuke поділені на 3 області (рис.3): область перегляду матеріалу viewer; область з вкладками Dope Sheet, Node Graph (за замовчуванням, і тут розміщується дерево ноду), Curve Graph; область Properties. Кожну панель можна закрити.

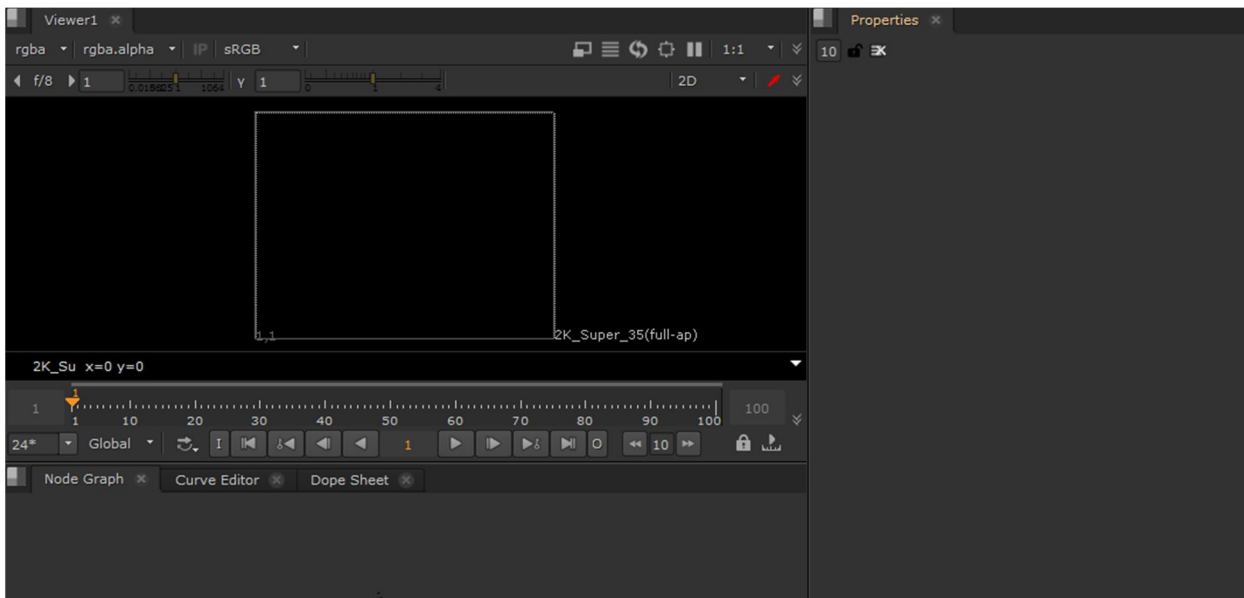



Рисунок 3 – Області програми

Основні елементи управління з основними панелями програми викликаються через кнопку, яка розташована у вигляді піктограми у крайньому верхньому лівому куті програми (рис.4) .

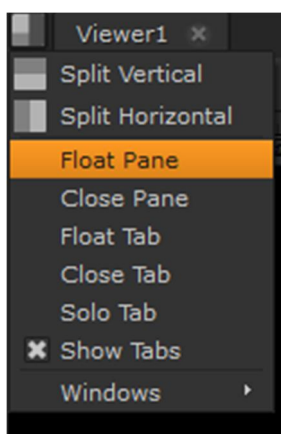


Рисунок 4 – Варіанти розташування панелей

У цьому наборі варто відмітити пункт Windows – вибір з усіх доступних користувачу панелей або інструментів в даній області програми. Слід зазначити, що панелі або вікна відображаються в програмі у вигляді закладок.

1. Особливості панелі Node Graph.

Панель Node Graph – панель, до розміщуються усі елементи створеного проекту. При цьому ці елементи є графічними і складаються компоненти мультимедійного контенту. Прикладом таких елементів може бути зображення, відео файл або навіть відео ефект.

Спочатку додамо до створеного проекту зображення. Для цього в області Node Graph через праву кнопку миші викликаємо список, де знаходимо рядок Image і там дістаємо рядок Read (рис.5)

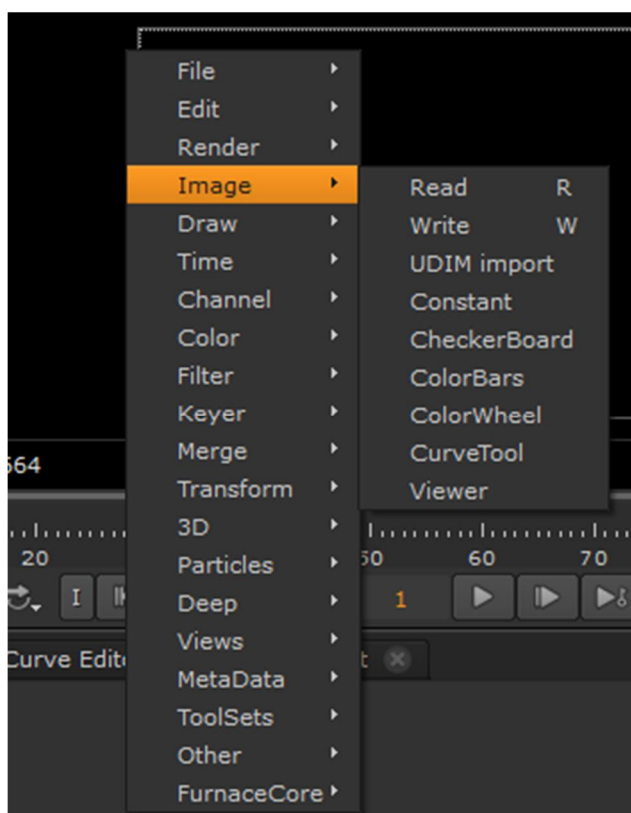


Рисунок 5 – Виклик на панель файла-зображення

Слід відмітити, що зв'язки між елементами показано через лінії зі стрілками. Як правило, в проекті кількість нодів не менше 100. Для переміщення виділених нодів в панелі Node Graph можна використовувати комбінацію клавіш: Alt+ліва кнопка миші. Крім картинок в проект можна завантажувати і секвенції (папка з набором подібних файлів). Але для завантаження секвенції в цілому у вікні Read file(s) є опція – позначка sequences (рис.6) і тоді файли можна завантажити, але якщо їх назва буде складатись з 2 частин – назва файлів і номер кадру.

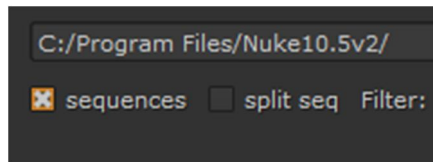


Рисунок 6 – Встановлення опції завантаження секвенції

Далі для підключення зображення до екрану (рис.7) її потрібно підключити до ноди viewer. Для встановлення зв'язку слід перетягнути кінцівку стрілки одного ноду до іншого. Для розірвання зв'язку треба мишею виділити цю стрілку і відтягнути її по далі від ноду.

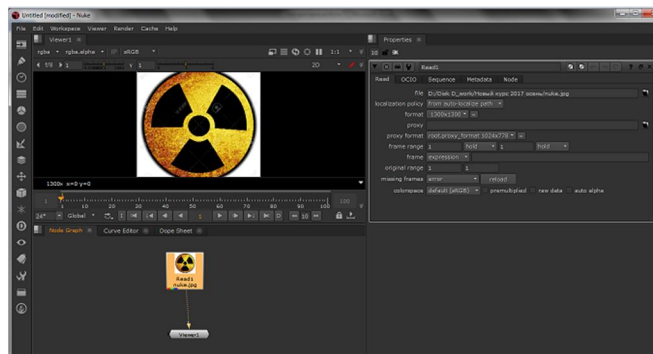
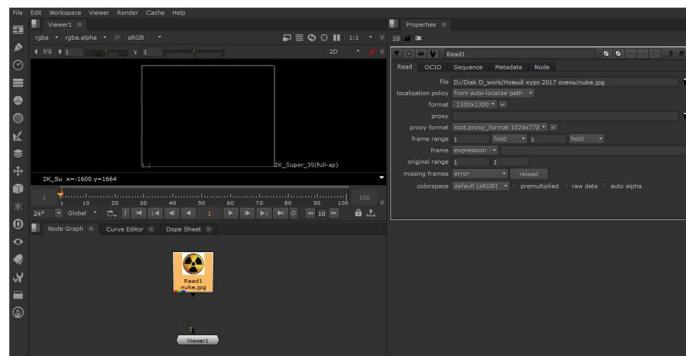


Рисунок 7 – Елементи проекту

Для додавання нодів у проект, які безпосередньо використовуються для роботи з зображеннями можна використати групу Filter, яка міститься у списку головної панелі інструментів програми. Зазначимо, що деякі ноди не можуть існувати окремо від зображення.

Результат дії однієї з обраних нодів можна спостерігати на рисунку 8.

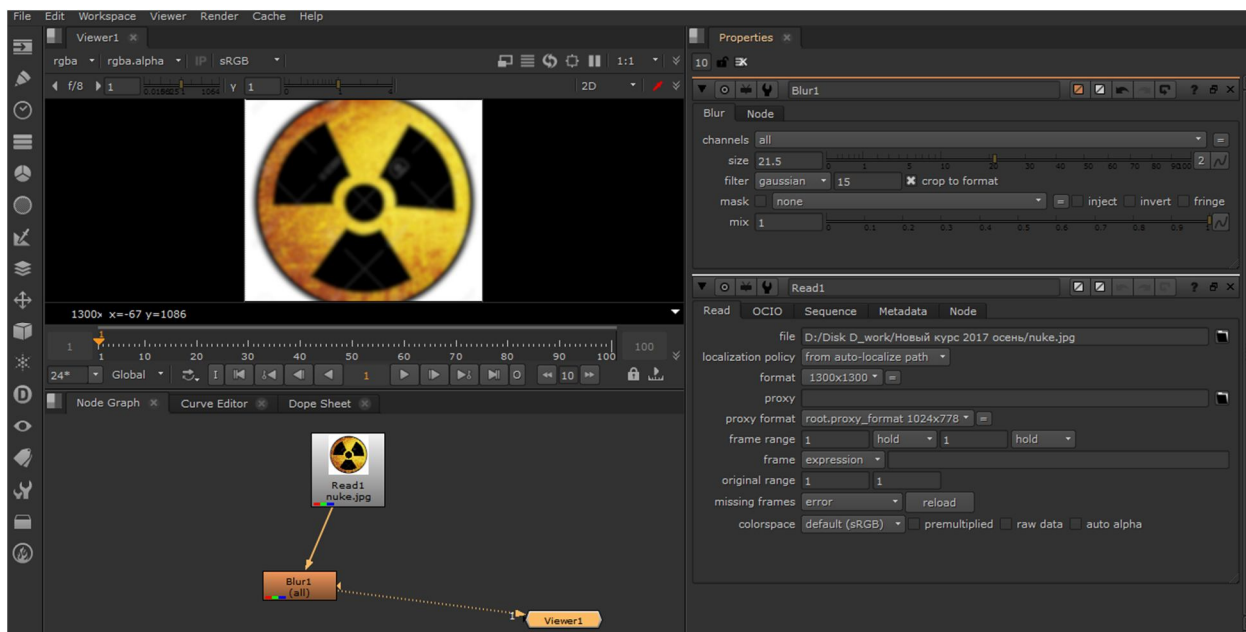


Рисунок 8 – Обробка зображення

Зазначимо, що для додавання ноди до виділеної ноди але з підключенням через окремий канал виходу, треба при виділеному зображенні натиснути клавішу Shift.

2. Операції з нодами.

Для проведення різних операцій з нодами слід використовувати елементи верхнього меню Edit програми Nuke. При цьому варто розуміти різницю між клонуванням та дублюванням ноди. Дублювання – операція отримання ноди, зміни котрої не впливають на вихідну ноду, з якої зроблено дубль.

При клонуванні ноди у верхньому кутку елемента з'явиться літера "с" (рисунок 9).

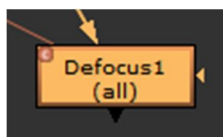


Рисунок 9 – Клонований елемент

Слід відмітити, що деякі ноди мають лише вхідні з'єднання (наприклад viewer), а деякі лише вихідні з'єднання, при чому їх кількість може бути довільною (наприклад для ноди read). Окремі ноди можуть мати окремі входи для маски.

Для створення нового з'єднання ноди слід натиснути клавішу Shift і мишею тягнути основну стрілку зв'язку до потрібного місця підключення.

Для встановлення ноди у існуюче з'єднання слід просто перетягнути цю ноду на стрілку з'єднання.

В області Node Graph клавіша Ctrl на полі проекту дозволяє вивести посередині ліній зв'язку позначки у вигляді жовтих ромбиків (рис.10). Натискання на ці елементи дозволяє створити “коліно”- злам для упорядкування підключень.

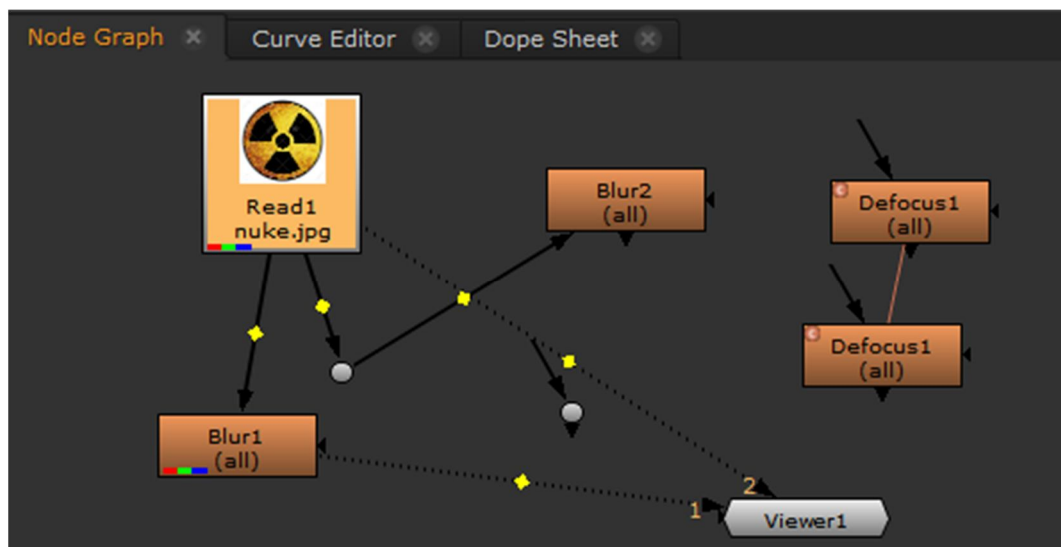


Рисунок 10 – Додаткова інформація про з'єднання

Для розірвання зв'язків ноди слід її виділити і натиснути комбінацію клавіш: Ctrl+D. Для витягування ноди зі з'єднання не розриваючи останнє слід використати комбінацію Shift+Ctrl+X.

Для відключення ноди в проекті слід при її виділенні натиснути клавішу D.

З правої сторони від назви ноди в області Preferences є дві позначки у вигляді квадратів (рис.11). Перший квадрат title color – зміна кольору самої ноди в області проекту, gl_color – колір елементів управління, які з'являються на самому зображенні у екрані viewer.

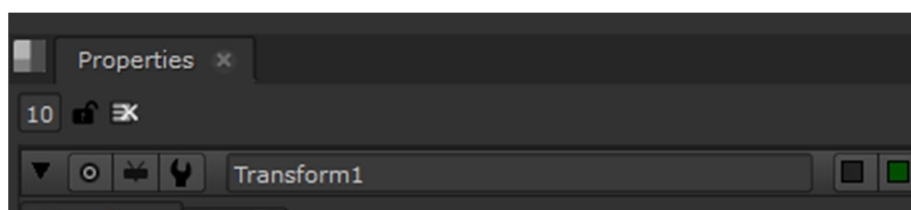


Рисунок 11 – Додаткові налаштування

3. Панель viewer (монітор перегляду).

Монітор перегляду є основним в програмі Nuke і дозволяє в режимі реального часу переглядати зміни, які відбуваються на основі створеної схеми дерева нодів. Монітор перегляду має великий набір інструментів (рис.12), які структурно можна виділити у частини – верхня (2 рядки) та нижня – панель часової дорожки.

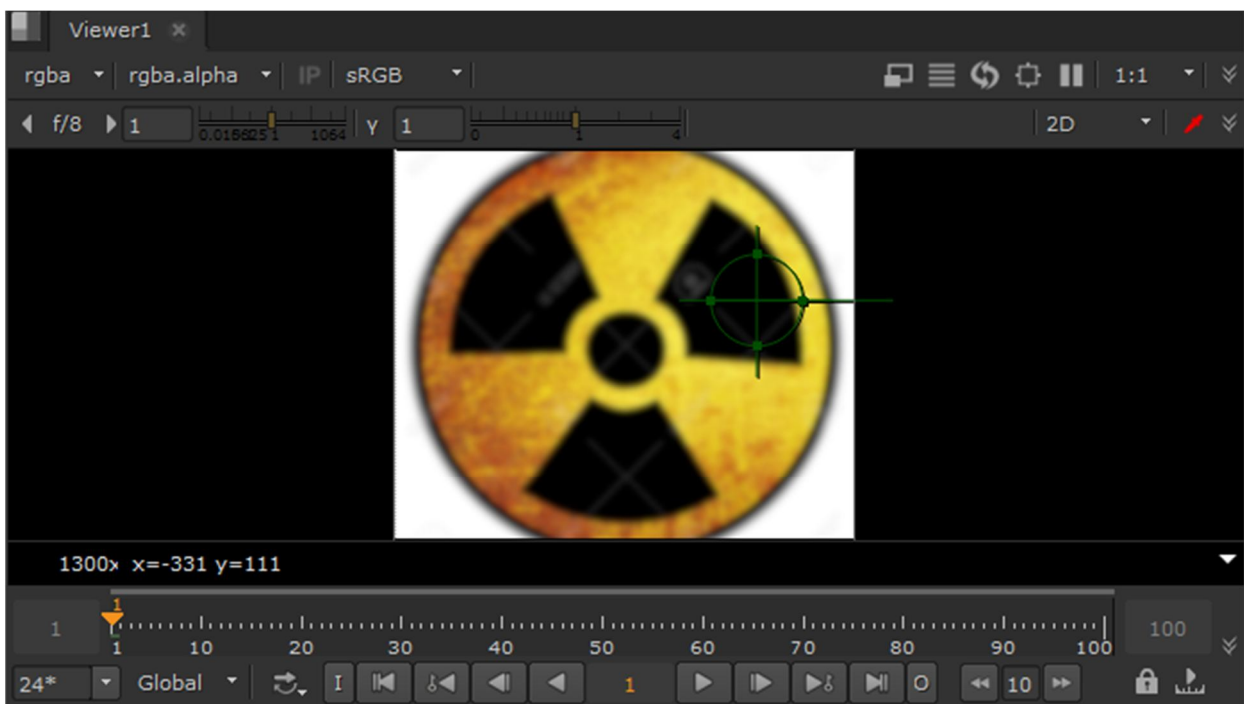


Рисунок 12 – Модуль viewer

Окрім стандартних інструментів програми, такі як зміна масштабу зображення є панель каналів зображення (1 верхній ряд кнопок) (рис.13)



Рисунок 13 – Канали зображення

Налаштування rgba – канал записаного зображення (кольоровий шар та шар альфа каналу); RGB – відображення каналів у viewer (канал ноду), тобто показується відразу 3 канали обраного шару.

Кнопка панелі 2D вибір навігації простору (рис.14)

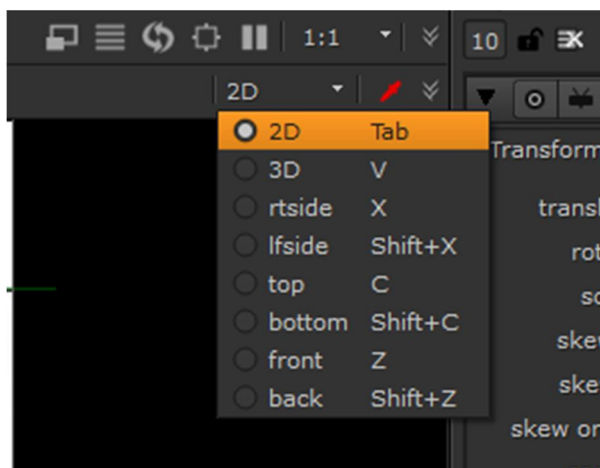
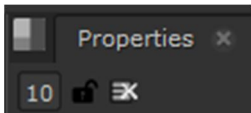


Рисунок 14 – Навігація

Для перевірки наявності альфа-каналу слід знайти над панеллю часової дорожки біла крапка (координати кольоровості - цифра білого кольору поряд з цифрою синього кольору).

Контрольні запитання

1. Як в програмі Nuke зробити панель окремим вікном ?
2. Як в області дерева нодів можна викликати тестову настрою вальну таблицю, кольорове коло, інструмент малювання кривих ?
3. Яким чином можна записати створений файл контенту ?
4. Що означає позначка '1' у виділеному ноді viewer ?
5. Чому при додаванні ноди Blur і підключення її до ноди viewer спостерігаємо чорний екран в моніторі програми ?
6. Поясніть алгоритм обробки зображень з групи нодів Filter у програмі Nuke.
7. Для чого в проекті іноді необхідно блокувати ноду з залишення існуючої архітектури схеми ?
8. Яким чином можна змінити назву ноди ?
9. Що означає позначка "10" на рисунку  ?
10. Що необхідно виконати щоб провести перегляд 3D об'єктів сцени ?

11. Як можна переглянути канал прозорості зображення в моніторі перегляду ?

Засоби використання скриптів на мові програмування Python в програмі Nuke FX для обробки різнопланового мультимедійного контенту

Мета. На основі правил об'єктно-орієнтованого програмування Python навчитись створювати скрипти для створення, зміни та обробки аудіовізуального контенту (зображення та відеопослідовності) в спеціалізованому вузловому середовищі Nuke FX. **Вміти** Навчитись проводити обробку зображень лише на основі створених блок-схем проекту програми, які побудовано через написання програмних кодів.

Інструменти. Встановлене на персональному комп'ютері програмне забезпечення Nuke FX та наявні функції, процедури, алгоритми мови програмування IDLE Python 2.7.

Завдання до лабораторної роботи.

1. Створити композицію в проекті програми Nuke і перейти в режим скриптіngu. **Усі дії в лабораторній роботі виконуються лише шляхом написання коду.**

2. Створити блок-схему обробки відеоряду. Кількість кадрів ТС обмежено 100, при цьому анімація параметрів для бригади визначається за правилом: бригада 1 -20; бригада 2 – 30, бригада 3 – 40; бригада 4 – 50; бригада 5 - 50.

Кількість обов'язкових елементів композиції – не менше 9 (не враховується ноди viewer, read, write).

Обов'язкові елементи в конструкції:

Бригада 1- **CheckerBoard; Color (3 різні будь-які на вибір);**

Бригада 2- **Flare; Text; Filter (4 різні будь-які на вибір);**

Бригада 3- **Grid; Keyer; Particle; Deep;**

Бригада 4- **Dither; Merge (3 різні будь-які на вибір);**

Бригада 5- **ColorBars; Transform (2 різні будь-які на вибір); 3D.**

3. Підключити та упорядкувати створені ноди за горизонтальною та вертикальною структурою (один під одним або в лінію).

4. У доданих нодах на вибір змінити 3 параметри, причому один з них зробити анімованим. Для однієї з ноди (не анімована) створити бігунок, слайдер, позначка.

5. Створити цикл виведення усіх назв нодів композиції проекту (цикл for).

6. Кількість рендерованих кадрів відповідає кількістю кадрів анімованого ноду. Вихідний контент для обробки і для запису, його кількість - номер бригади+2.

Звіт за кожним пунктом повинен обов'язково мати скрипт та результат його виконання (скріншот).

Теоретичні відомості (основні налаштування)

Скриптинг або програмування дозволяє користувачу або розробнику зручніше та швидше використовувати інструментарій програми Nuke користуючись в даному випадку основами об'єктно-орієнтованого програмування Python. Завдяки скриптингу також з'являється можливість створювати нові елементи та змінювати за своїм уподобанням існуючі елементи, які відображають алгоритм роботи з аудіовізуальним матеріалом. В лабораторній роботі буде визначено можливості мови Python в середовищі Nuke а також проведено прийоми персоналізації меню програми.

Для входу в режим скриптингу можна використати відповідній елемент верхнього меню (Workspace-Scripting) або використати комбінацію Shift+F4 (рис. 1)

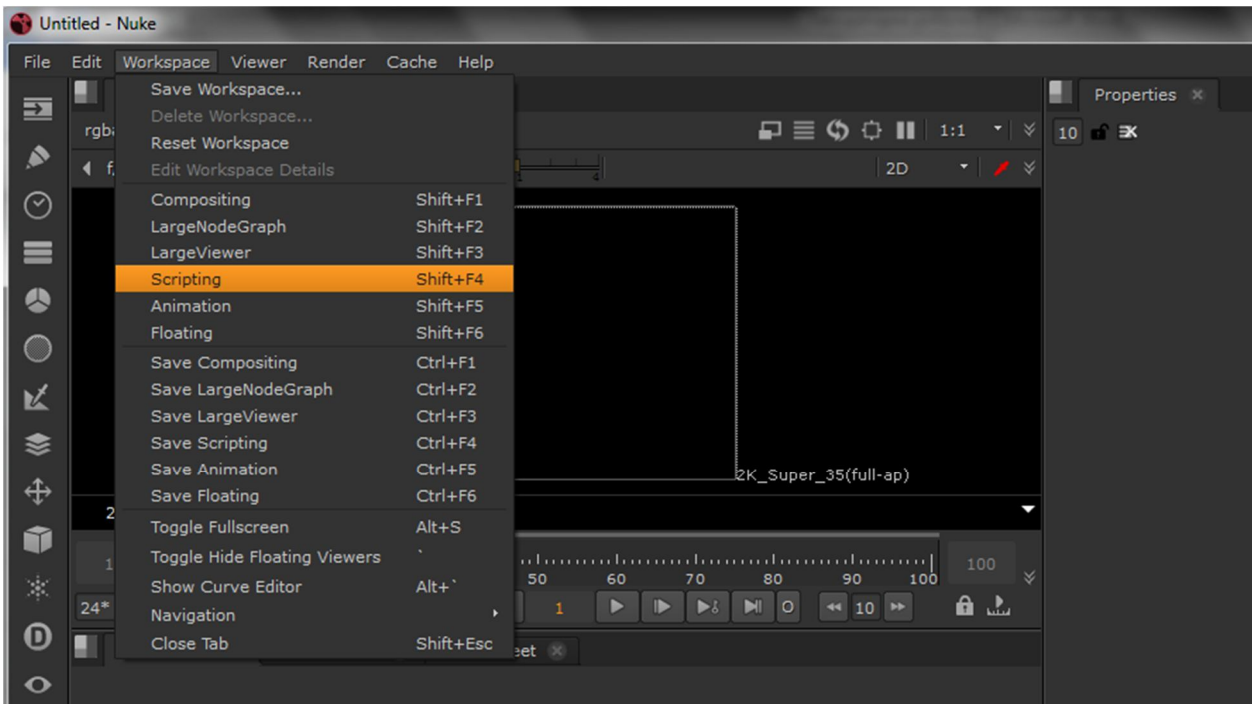


Рисунок 1 – Вхід в режим програмування

В результаті отримаємо режим скріптіngu (робоча область умовно поділена на 4 частини), де верхнє ліве вікно буде показувати результати програмування, а нижня частина режиму – це вікно де буде вводиться код (рядок “з жовтою одиницею”).

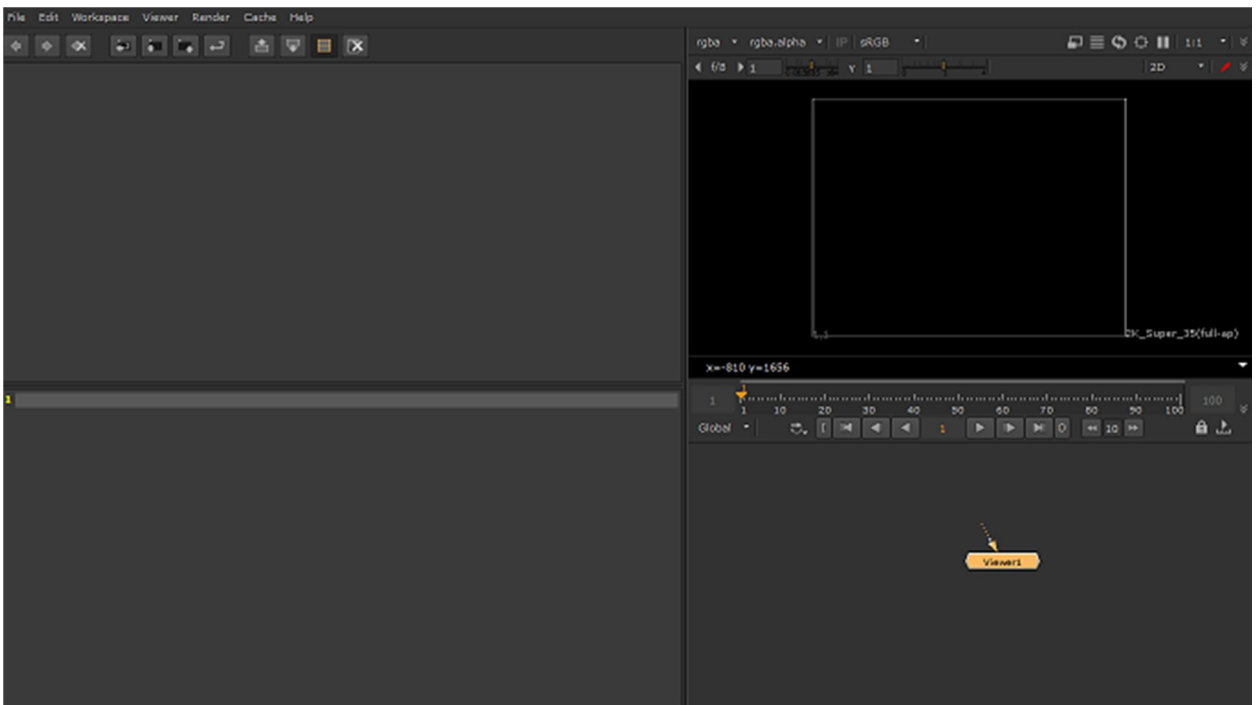



Рисунок 2 – Режим скріптіngu

Протестуємо режим і для цього після збереження файлу композиції, у нижній частині вікна введемо функцію:

```
print 'hello'
```

і для запуску цієї функції (найпростішого скрипту) використаємо кнопку у верхньому меню  або через гарчу клавішу ctrl+Enter і в результаті отримаємо результат (рисунок 3)

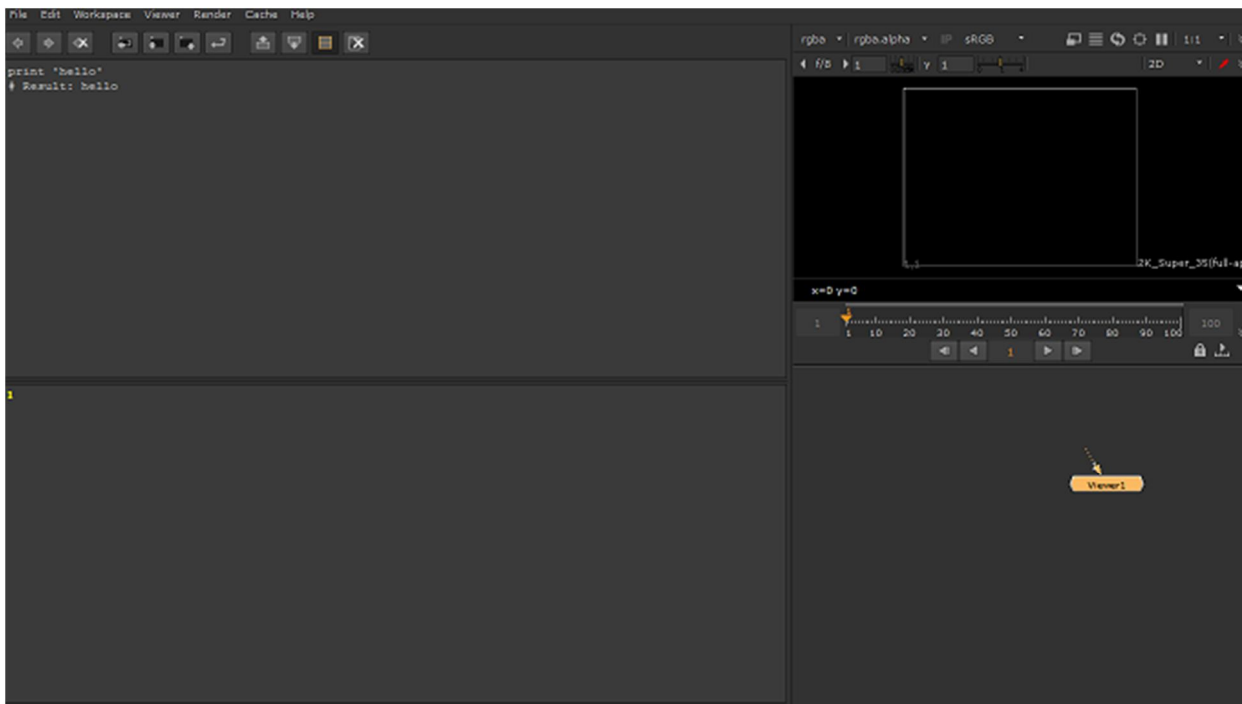


Рисунок 3 – Тестування режиму

Для виведення різної інформації в окремому вікні можна використати команду:

```
nuke.message('hello')
```

і тут отримаємо окреме вікно як результат роботи скрипту (рисунок 4).

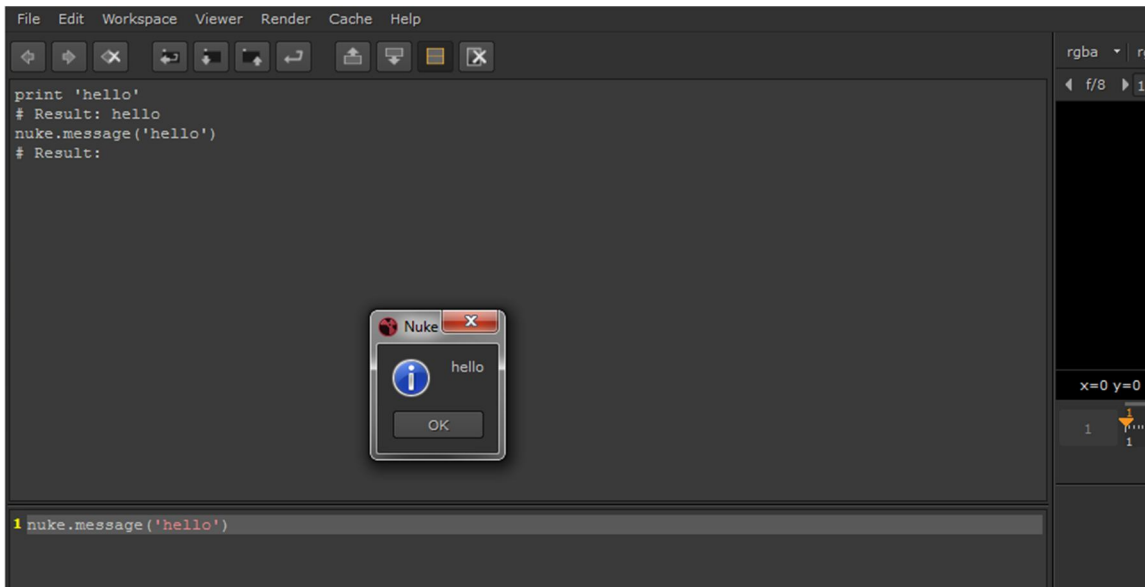


Рисунок 4 – Виведення повідомлення в окремому вікні

Для відображення вікна введення інформації (**крапка і дужки обов’язкові елементи**)

`nuke.getInput()`

– де у дужках буде знаходитись текст, який буде визначено за замовчуванням, наприклад “default” (текст можна у вікні написати будь-який).

Результат цієї команди можна побачити на рисунку 5.

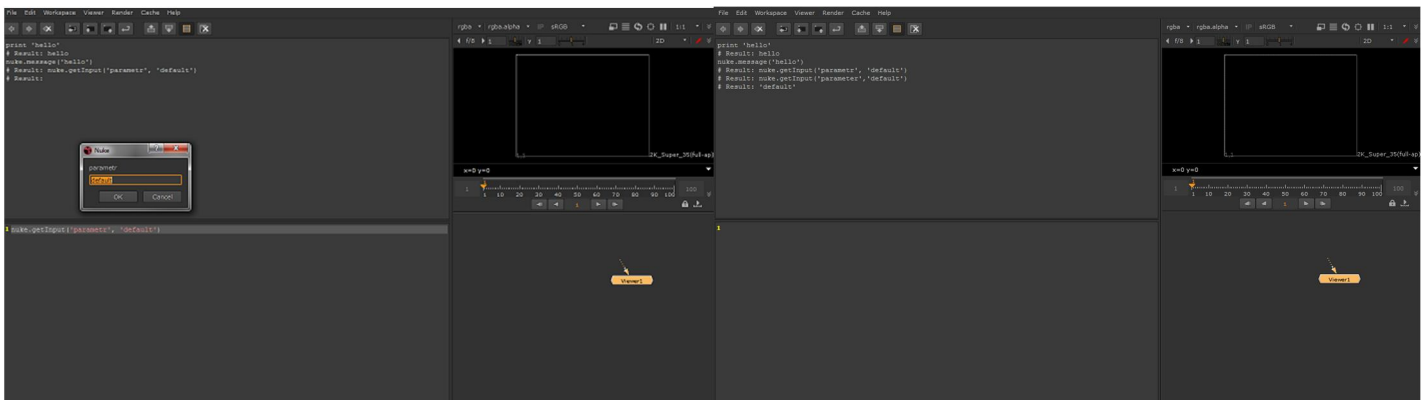


Рисунок 5 – Виведення вікна з інформацією

Команду виклику панелі з кольором (рисунок 6):

`nuke.getColor()`

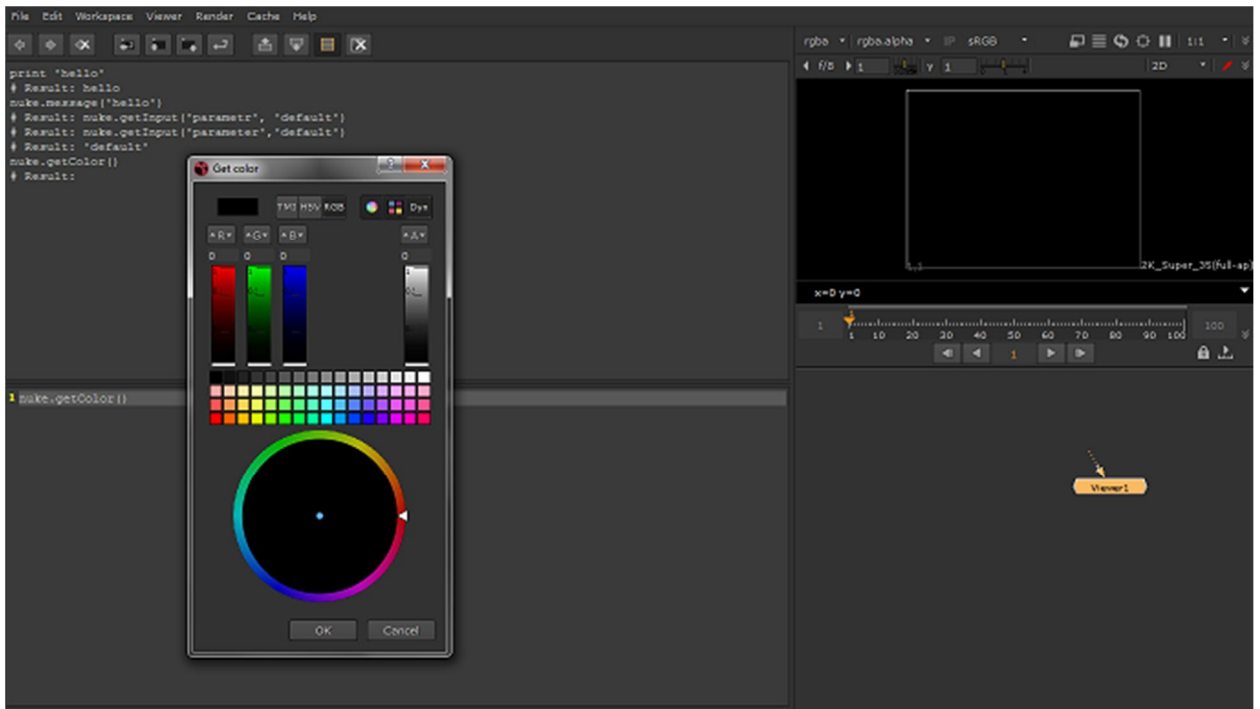


Рисунок 6 – Виведення панелі з кольором

Вікно виклику файлу можна викликати через команду
`nuke.getFilename('test')`

і тут треба обов'язково вказати назву вікна (рис.7), інакше буде виведена помилка.

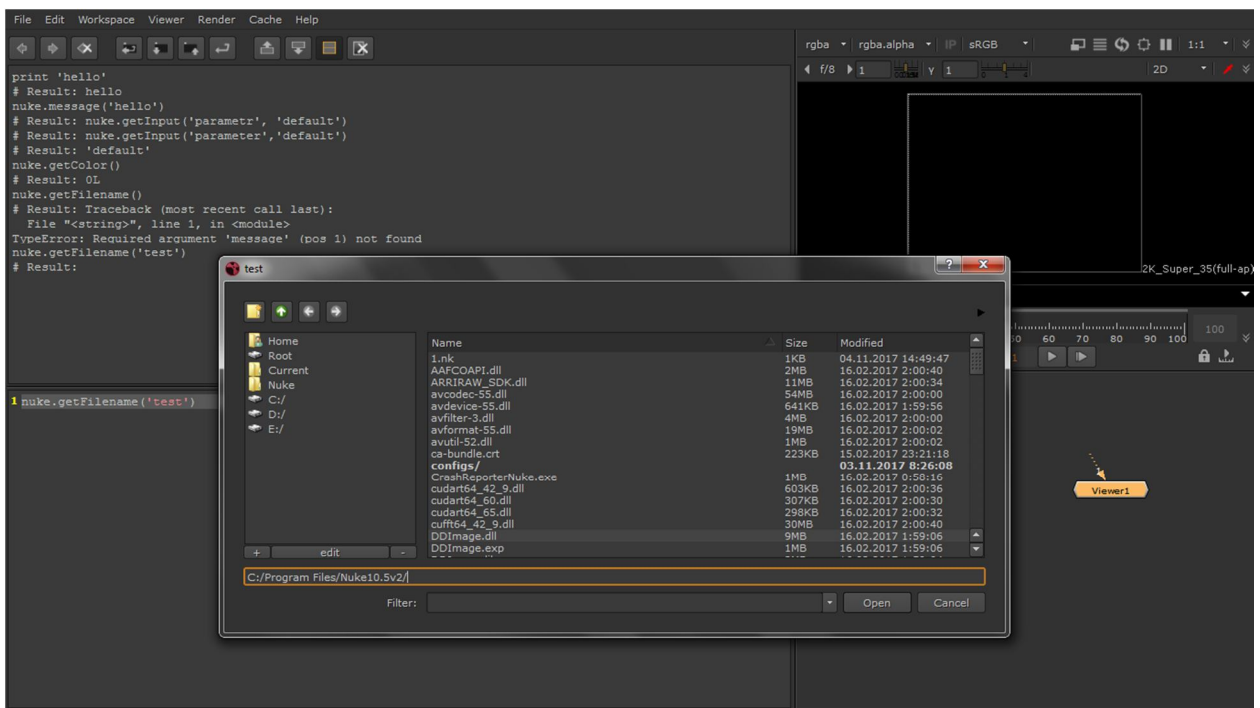


Рисунок 7 – Виведення шляху до файлу

1. Додавання вузла (ноди) в панелі NodeGraph (праве нижнє вікно, там же елемент екрану Viewer1).

Багато команд в програмі починаються зі слова `nuke.` та крапки після слова (слова і команди розділяються або точками або великими літерами). Для додавання ноду використаємо команду

```
nuke.nodes.Blur()
```

тобто з усіх доступних в програмі нодів обираємо ту, яка записана після другої крапки в команді (наприклад нода `Blur`) (рис.8):

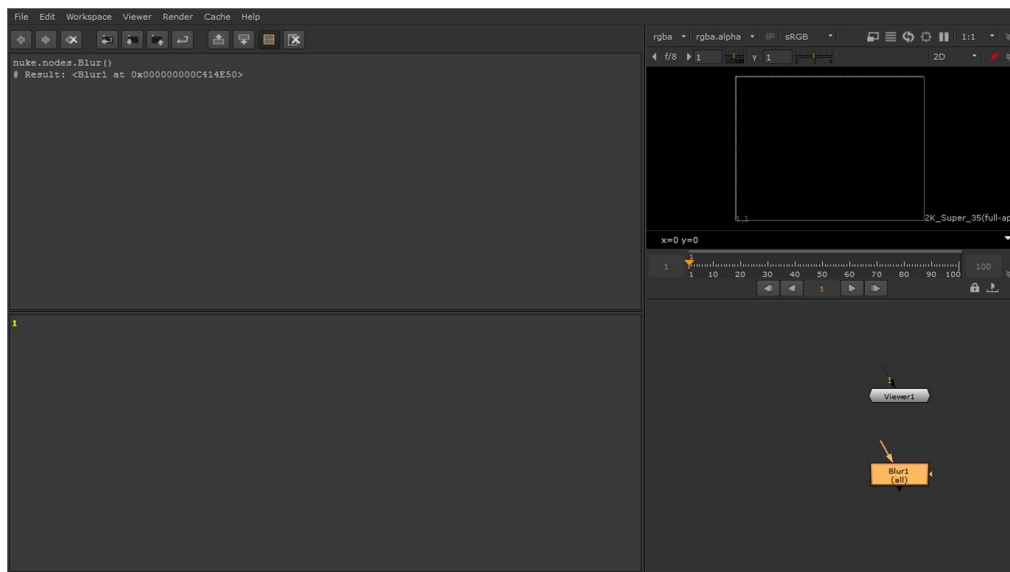



Рисунок 8 – Додавання нової ноди у вікно нодів NodeGraph

Для перегляду інформації про ноду достатньо її виділити і натиснути на клавіатурі клавішу “Г”.

Для управління рядками скрипту у вікні скриптіngu є органи управління рядками коду у формі двох стрілок (лівий верхній кут): . Після виконання скрипту його написання зникає у нижньому лівому вікні режиму. Для того, щоб це не виникало слід виділити рядок з командою і натиснути `ctrl+Enter`. При цьому текст з командою виконається і не пропаде.

Далі слід прописати зв'язок створеної ноди. Наприклад, необхідно нову ноду з'єднати з виділеною і створеною вже раніше у вікні NodeGraph автоматично. Для цього слід виконати наступну команду:

```
nuke.createNode('Blur')
```

де у дужках вводимо у одиночних лапках назву ноди (для прикладу запишемо ноду Blur). Зазначимо, що тут замість другої крапки використано велику літеру. В результаті буде додано ноду зі зв'язком з виділеною і буде додатково виведено вікно властивостей створеної ноди (рис.9).

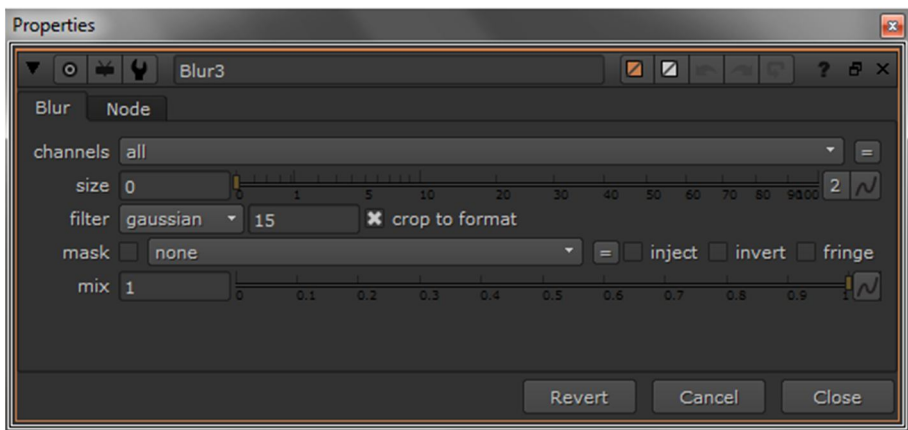
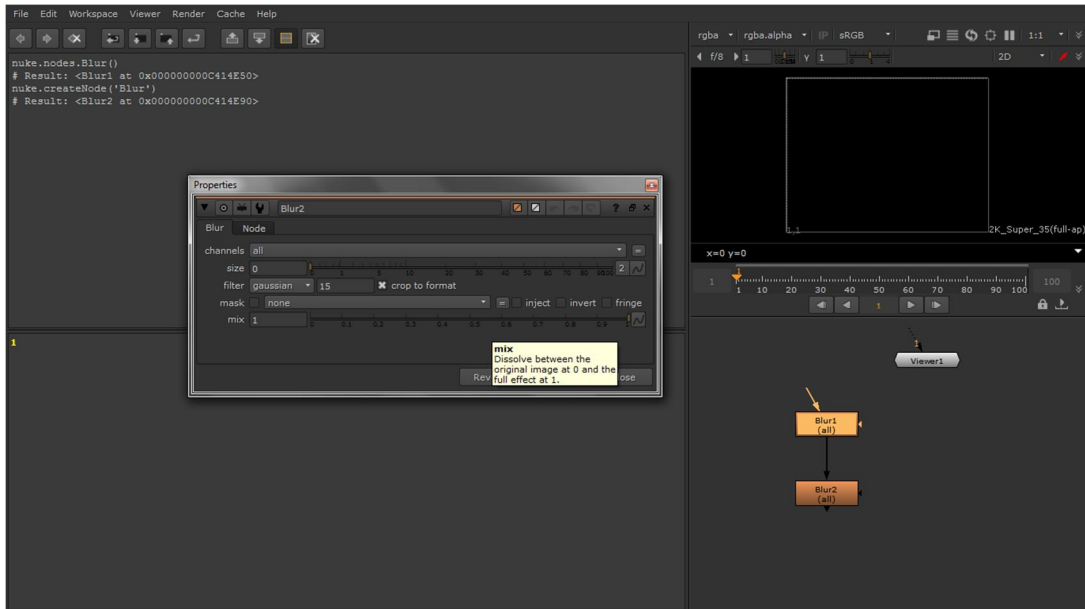


Рисунок 9 – Додавання ноди зі зв'язком та параметри ноду

Якщо ж нема необхідності у виведенні вікна з властивостями, то попередню команду слід доповнити таким чином:

```
nuke.createNode('Blur', inpanel=False)
```

і тоді матимемо новий нод без виводу вікна властивостей. Усі створені ноди були створені з параметрами за замовчуванням. Для зміни цих властивостей ще на етапі створення ноди можна використати наступний хід:

```
nuke.nodes.Blur(size=10)
```

де, у дужках слід вказати параметри, які слід задати (наприклад параметр size). Зазначимо, що не завжди назви параметрів співпадають з реальними назвами. Для того, щоб перевірити назви, треба просто мишею навести на область зміни параметри до появи пояснення. А при створенні нової ноди, треба після коми у дужках вказати значення параметру, який встановлюємо:

```
nuke.createNode('Blur', 'size 10')
```

– параметр далі пробіл і його значення.

Кожну ноду можна присвоїти певну змінну, яку потім можна змінювати. Наприклад, визначимо змінну b, як ноду

```
b=nuke.nodes.Blur()
```

і далі тепер можна переглянути параметри змінної b. Для цього використаємо команду

```
print b
```

Результат показано на рисунку 10:

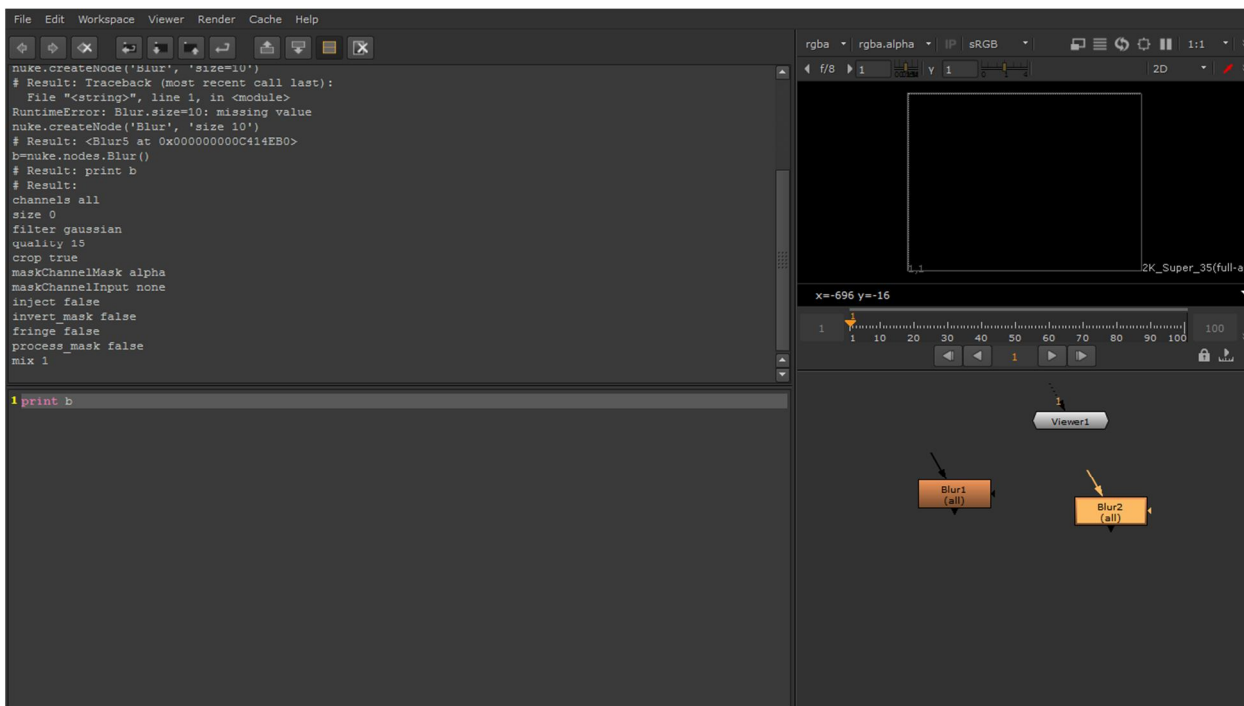


Рисунок 10 – Перегляд параметрів ноди

Для виведення на екран значення певного параметра ноди:

```
print b['size'].value()
```

де вказуємо який саме параметр треба вивести, його значення. Результат на рисунку 11.

Для того, щоб задати параметр, його конкретне значення, можна змінити попередню команду таким чином (20- нове значення параметру):

```
print b['size'].setValue(20)
```

Результат команди показано на рисунку 12.

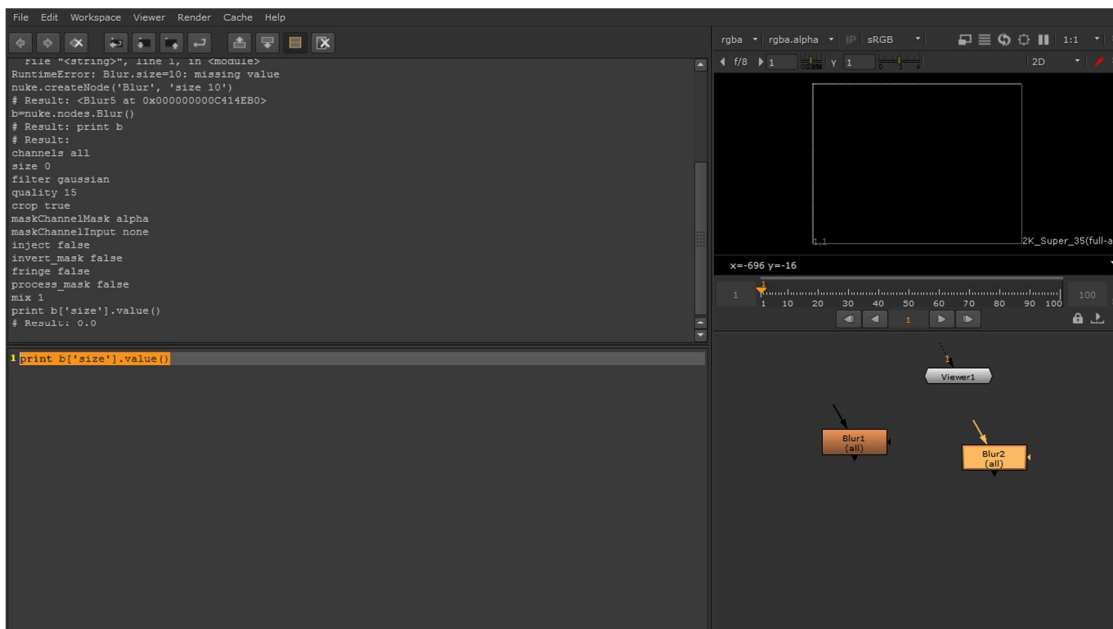


Рисунок 11 – Перегляд конкретного параметру ноди

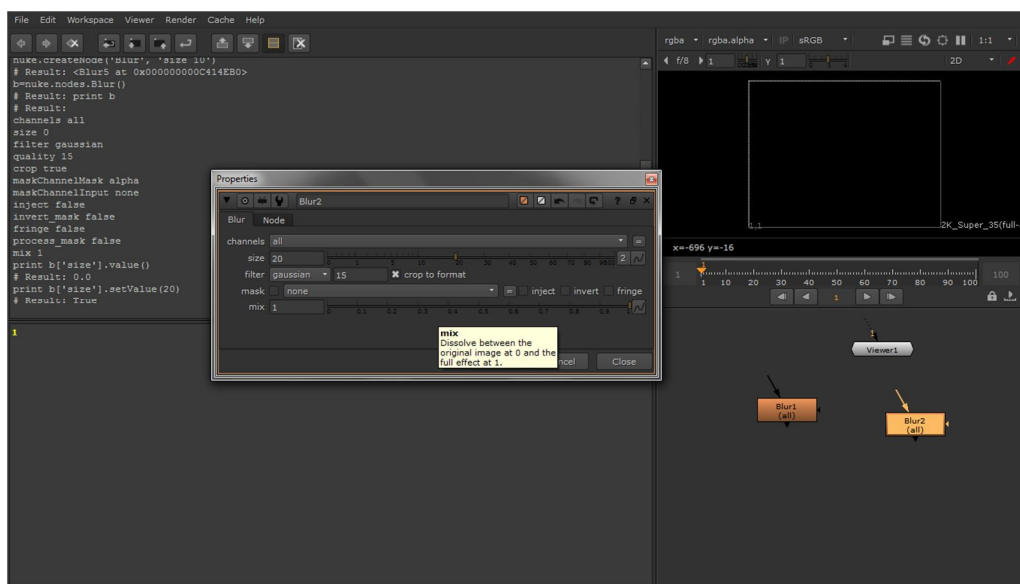


Рисунок 12 – Визначення параметру та його перевірка

2. Встановлення ключів анімації параметрів ноди

Кожний параметр ноди можна анімувати, тобто його значення буде змінюватись залежно від номера кадру на доріжці часового коду. Доріжка часового коду в режимі скриптіngu розміщено з правої сторони і має вигляд:

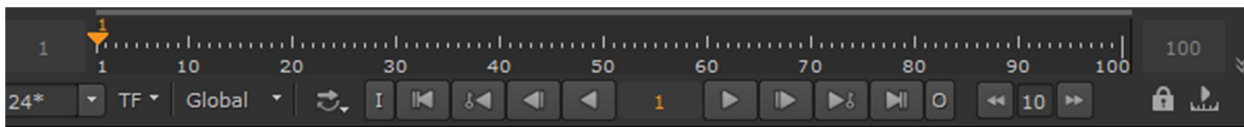


Рисунок 13 – Доріжка часового коду

Наведемо алгоритм анімації конкретного параметра ноди Blur. Спочатку створимо ще одну змінну:

```
b1=nuke.nodes.Blur()
```

для анімації параметра size визначеної ноди присвоїмо його новій змінній v

```
v=b1['size']
```

і для анімації цього параметру слід, по-перше прописати команду, тобто це означає, що будь-яка зміна його значення буде в програмі запам'ятовуватись

```
v.setAnimated()
```

і у вікні результату отримаємо такий запис:

```
b1=nuke.nodes.Blur()  
# Result: v=b1['size']  
# Result: v.setAnimated()  
# Result: True
```

А у властивостях ноди можна побачити, що відповідна комірка параметру ноди змінить свій колір (стане синім) (рисунок 14):

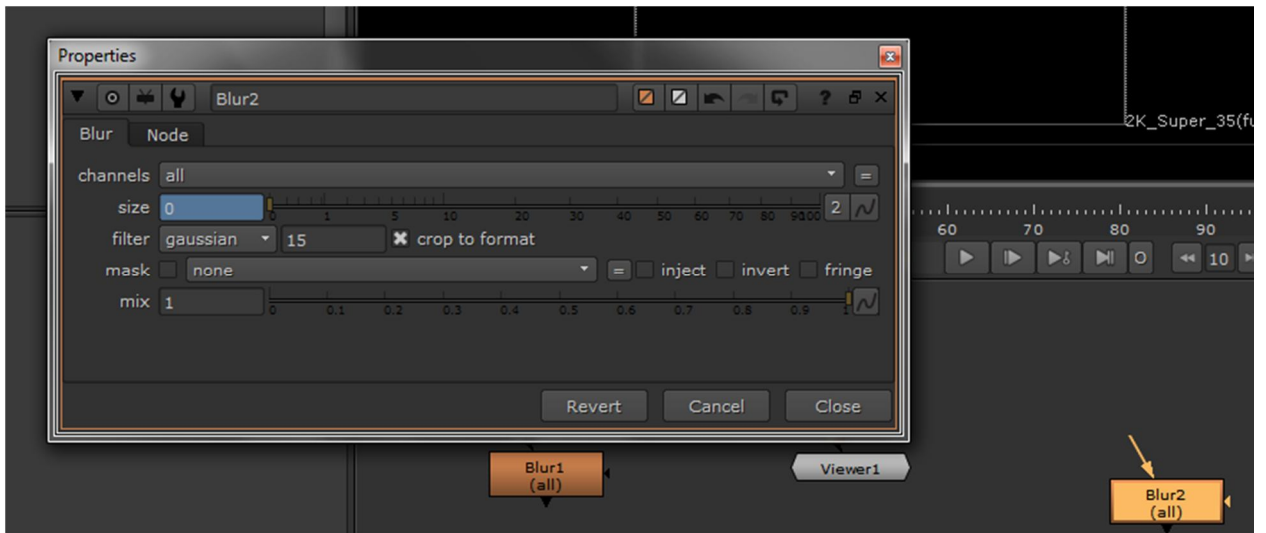


Рисунок 14 – Ознака анімації параметра ноди

І по-друге, далі треба цю анімацію задати, тобто змінити значення параметра на різних ділянках доріжки часового коду. Наприклад на першому кадрі встановимо значення параметра size=10

```
v.setValue( 10, time=1 )
```

зазначимо, що тут ми задаємо значення параметра на першому кадрі, яке дорівнює 10. І аналогічно змінимо значення на 30 у 50 кадрі. В результаті можна перевірити у властивостях ноди, що параметр зміниться (рисунок 15). Зазначимо про появу синіх штрихів на відповідних мітках часової доріжки.

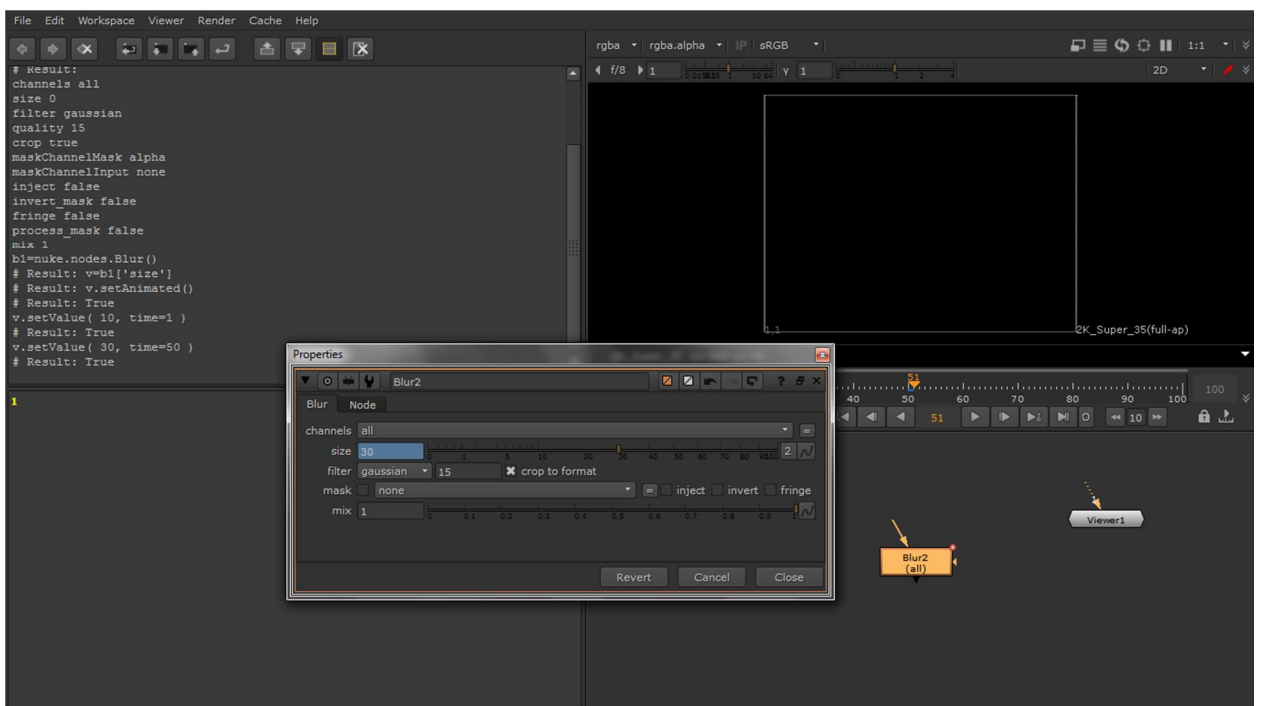


Рисунок 15 – Встановлення анімації

В результаті ми задали анімацію параметра ноди, значення якого змінюється між заданими кадрами (рис.16):

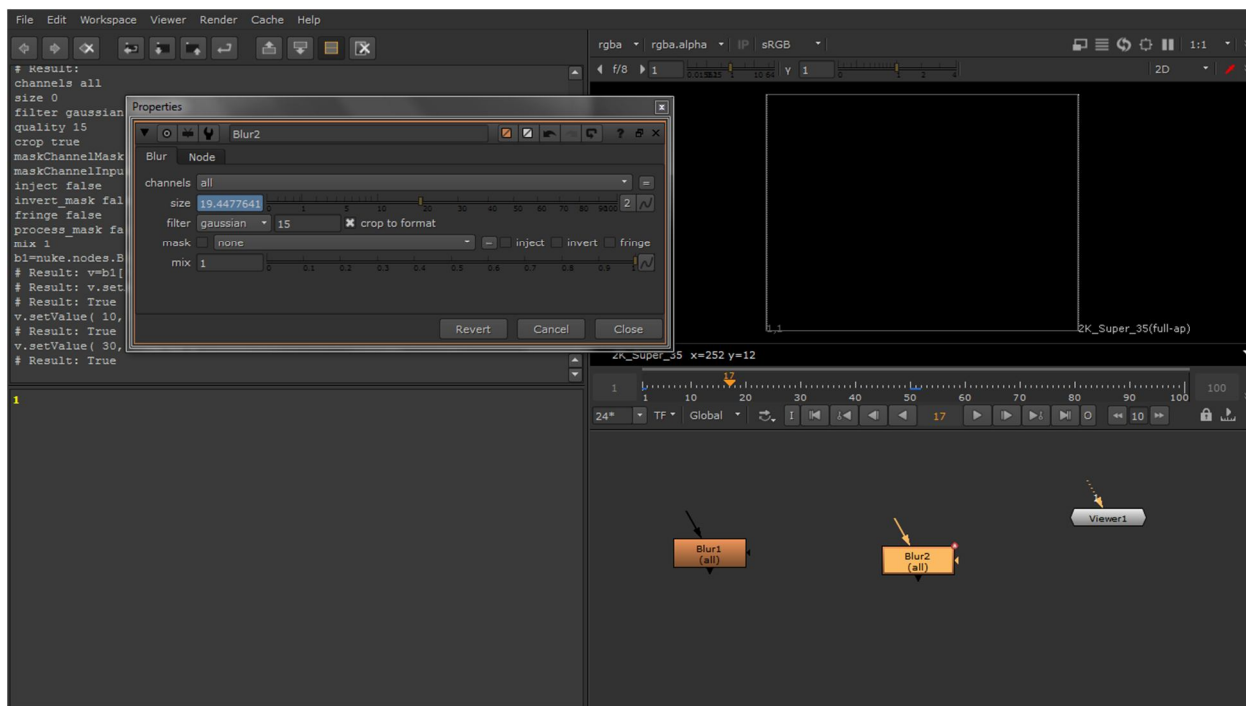


Рисунок 16 – Перевірка анімації

Але, крім цього, для створеної ноди користувач може додавати свої параметри. Це можна зробити за допомогою наступної команди, де присвоїмо цей параметр функцією Knob новій змінній k:

```
k=nuke.Array_Knob('Param', 'new_param')
```

де Array – тип комірки; (WH- бігунок, слайдер; Boolean – позначка у вікні властивостей за параметром) і ми лише в змінну внесли деякі параметри. Для включення цього параметра у ноду, наприклад, Blur

```
b.addKnob(k)
```

де b-змінна, де знаходиться нода і у дужках вказано , параметр, який буде додано, у даному випадку це k. Результат можна переглянути у властивостях ноди, де з'явилась нова вкладка (рисунок 17). Тут визначено новий параметр new_param, який називається Param.

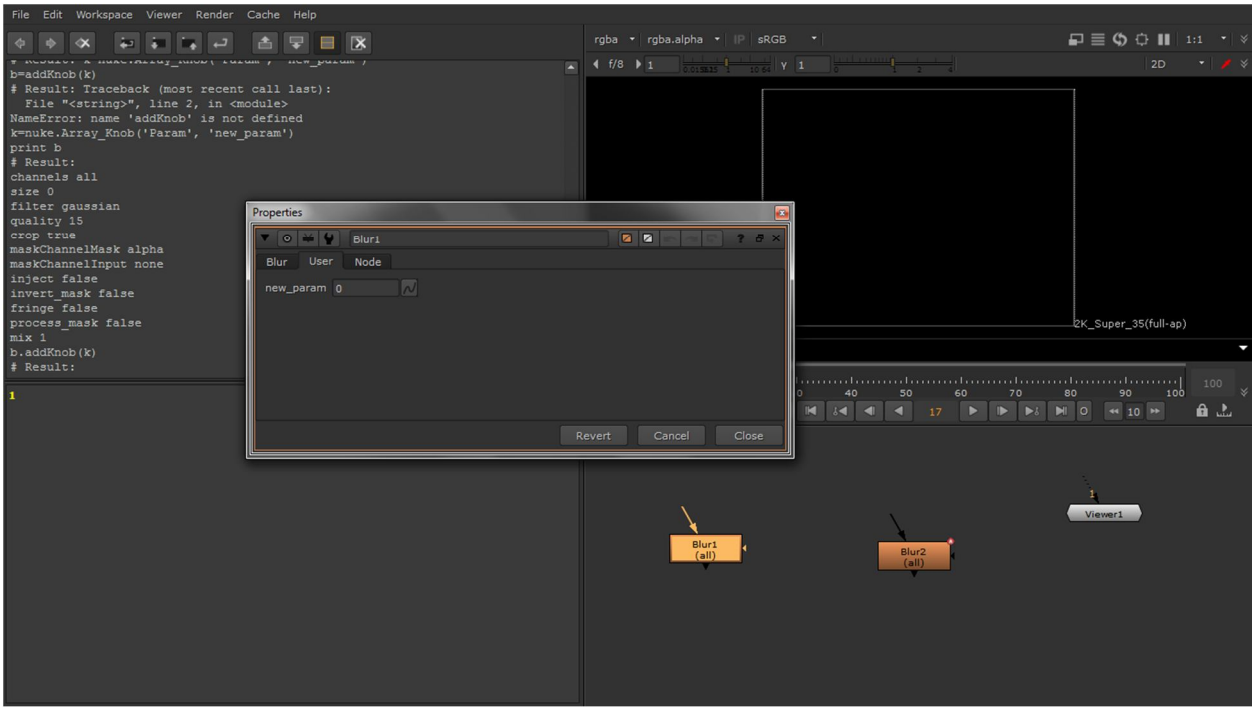


Рисунок 17 – Встановлення нового параметру

Зі встановленим слайдером (рисунок 18):

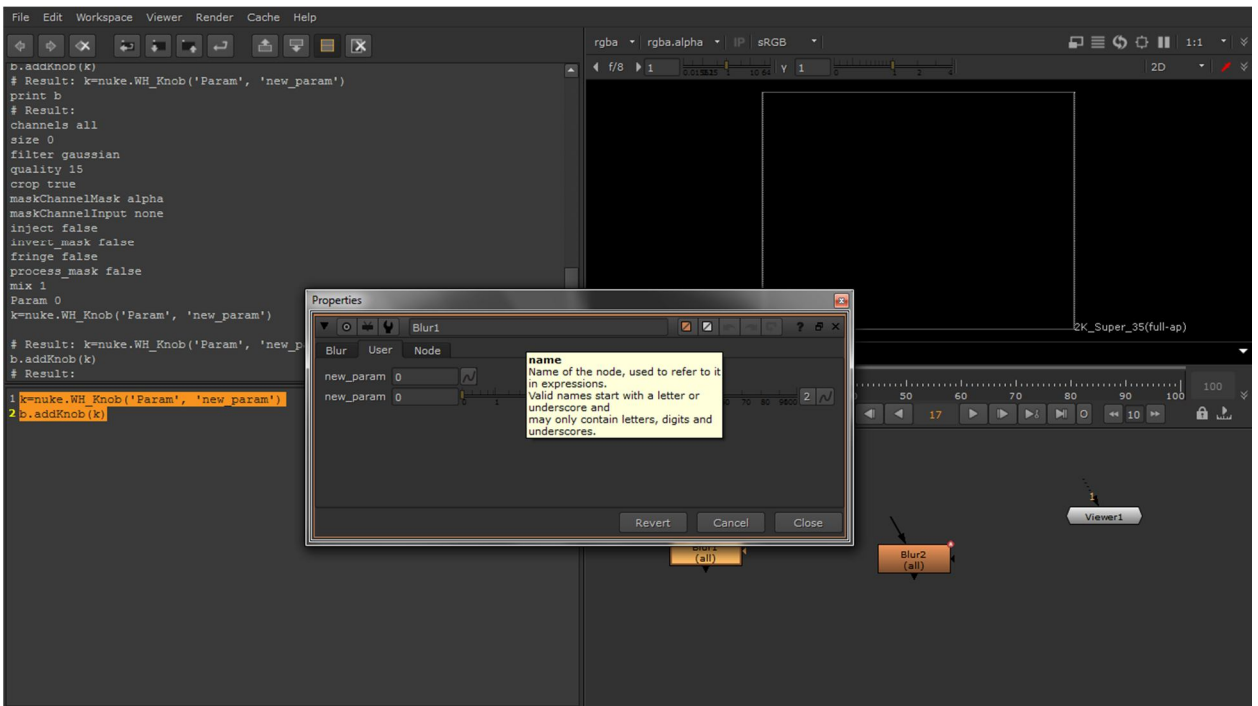


Рисунок 18 – Встановлення слайдеру

Зі встановленою позначкою (рисунок 19)

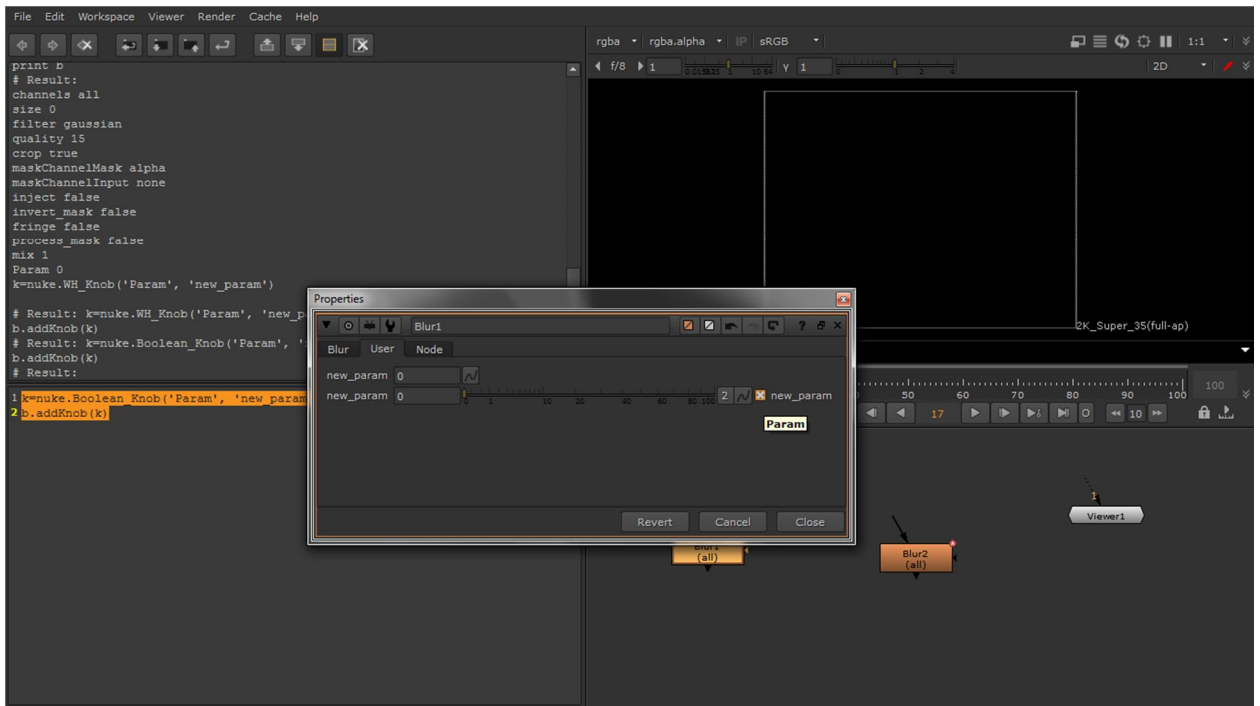
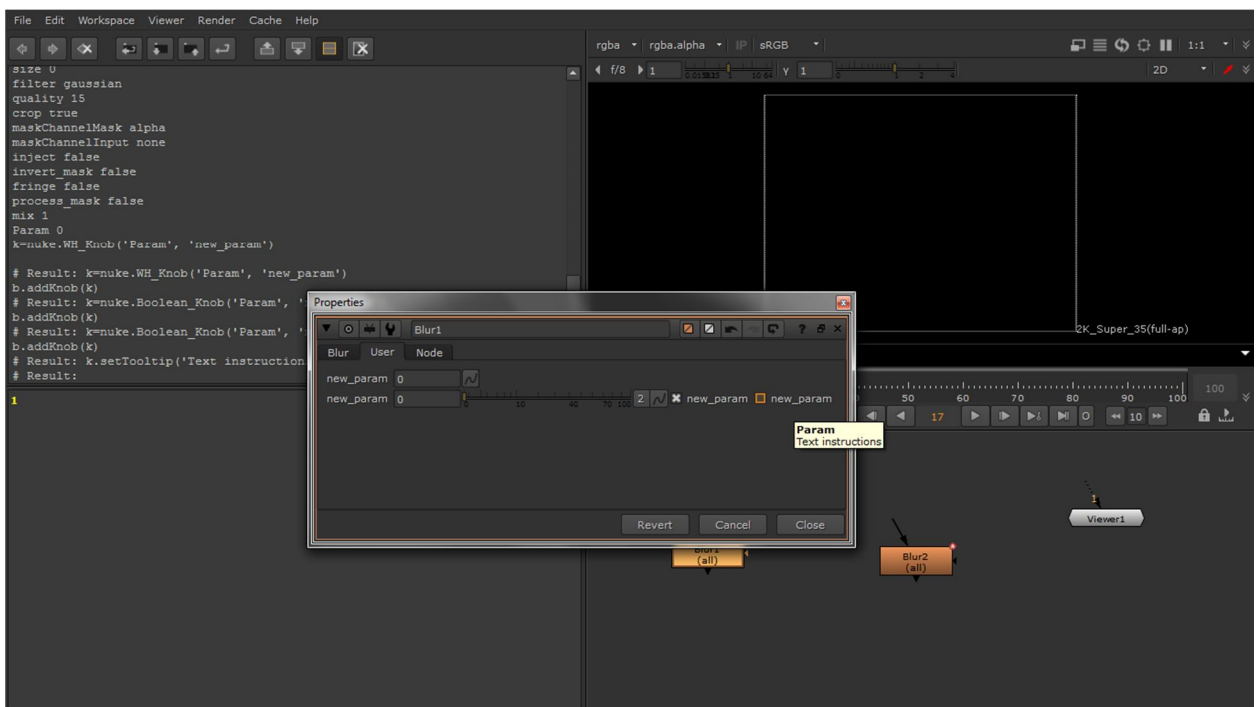


Рисунок 19 – Встановлення позначки

Для прописування нашому створеному параметру змінної k опису треба виконати команду:

```
k.setToolTip('Text instruction')
```

і в результаті можна перевірити:



3. Додаткові можливості при додаванні нод в панель NodeGraph

1) Місцезнаходження доданої ноди Grade в панелі NodeGraph:

```
g=nuke.nodes.Grade( xpos=0, ypos=0 )
```

де g-змінна; xpos, ypos – координати створеної ноди в панелі.

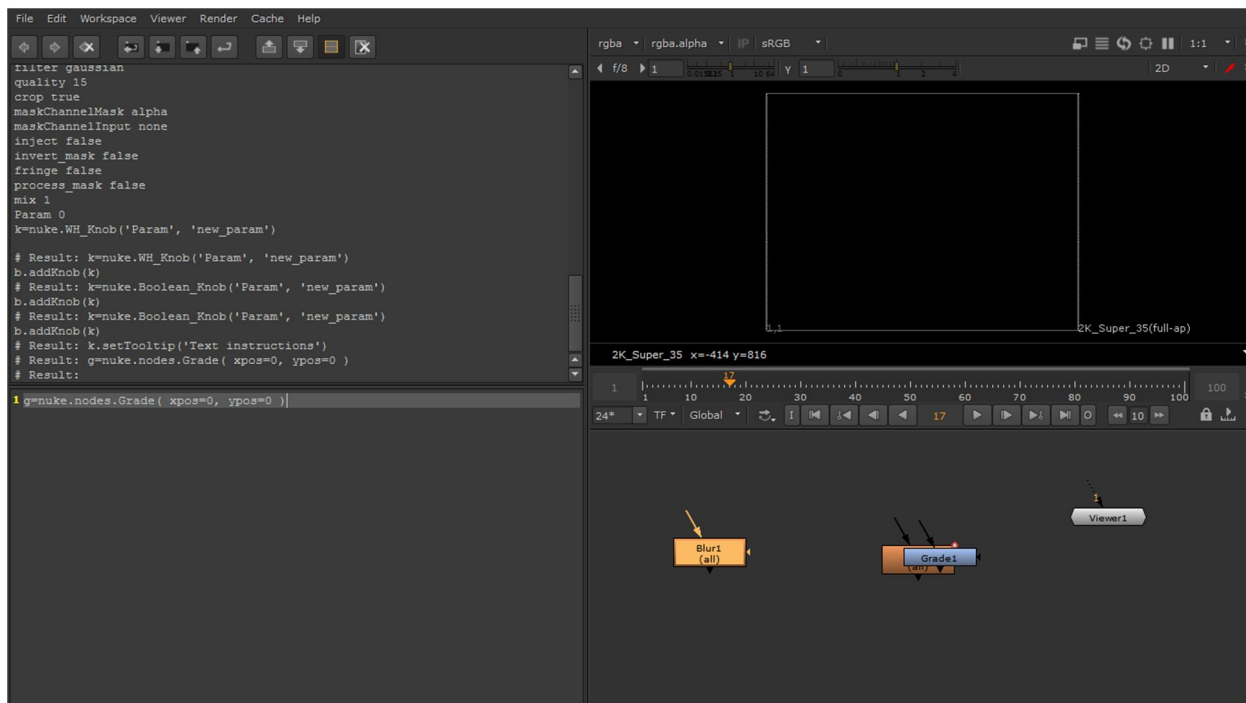


Рисунок 20 – Встановлення місцезнаходження ноди

Або можна ще простіше зробити, наприклад для ноди Blur (змінна b) на основі команди:

```
b.setXpos(50)
```

4. З'єднання нодів в панелі NodeGraph

Наприклад, необхідно додати ноду Merge (визначимо через змінну g), але так, щоб вона була з'єднана за входами з двома іншими нодами Blur та Grade, які раніше були визначені через змінні b та g відповідно. Для цього можна використати команду:

```
m=nuke.nodes.Merge(inputs=[g,b])
```

де g – для входу B; b- для входу A. Тобто, за списком вхід B (перший вхід), вхід A (другий вхід) (рисунок 21).

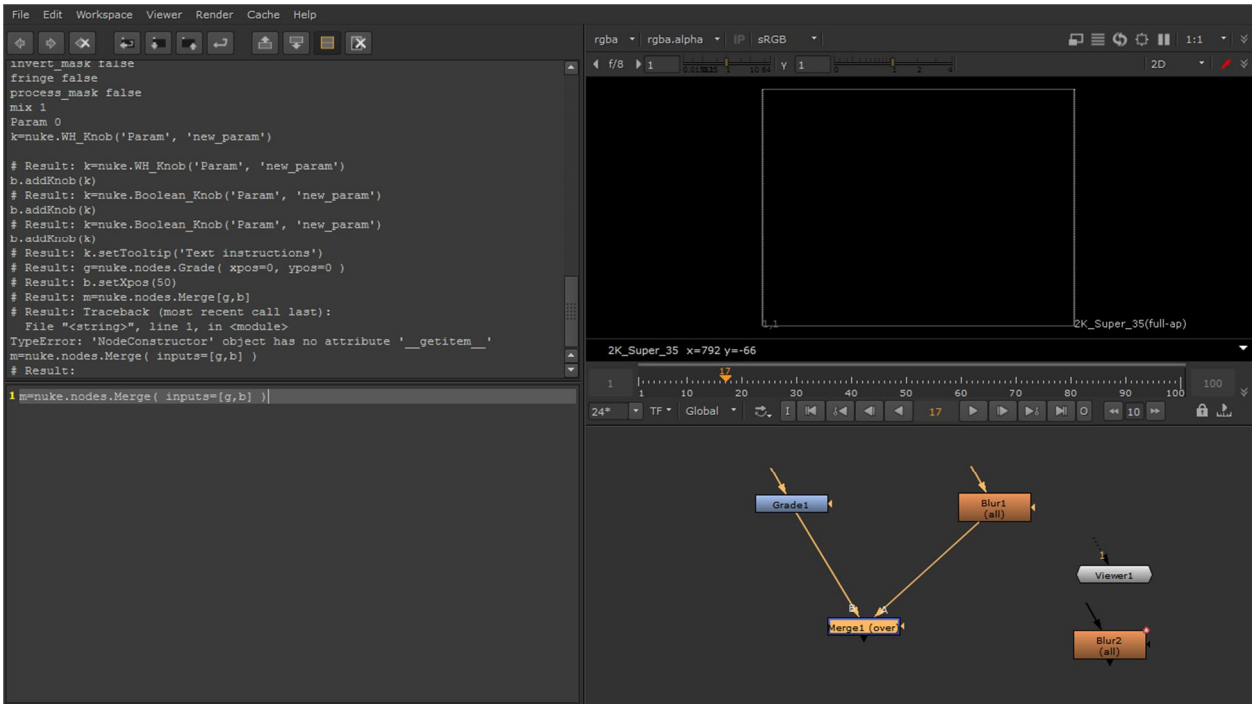


Рисунок 21 – Додавання нової ноди

Якщо ж у нас у ноді один вхід, то наприклад для визначеної змінної b1 (нода Blur)

`b1=nuke.nodes.Blur(inputs=[m])`

і тоді отримаємо з'єднання як на рисунку 22.

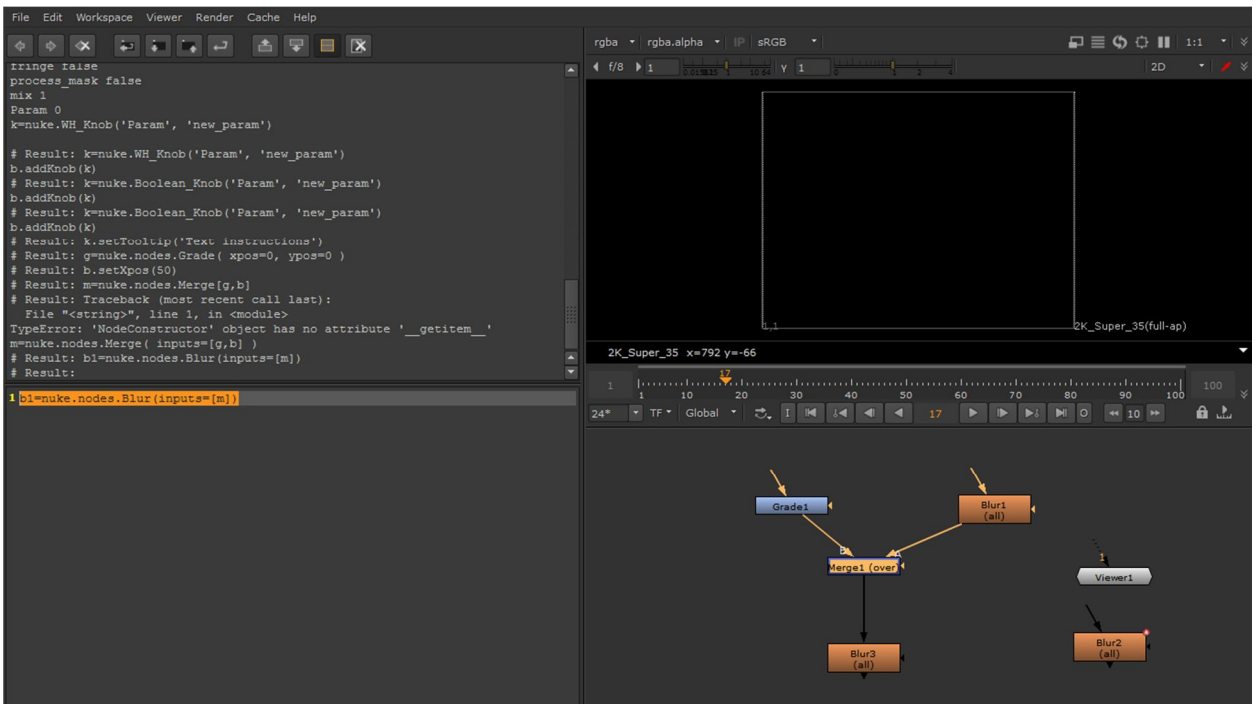


Рисунок 22 – Додавання ноди

Інша ситуація, коли є не підключена нода в панелі. Наприклад нода Blur 2 (визначена під змінною b1) і підключимо її до ноди Merge (змінна m) (рис.23):

```
b1.setInput( 0, m )
```

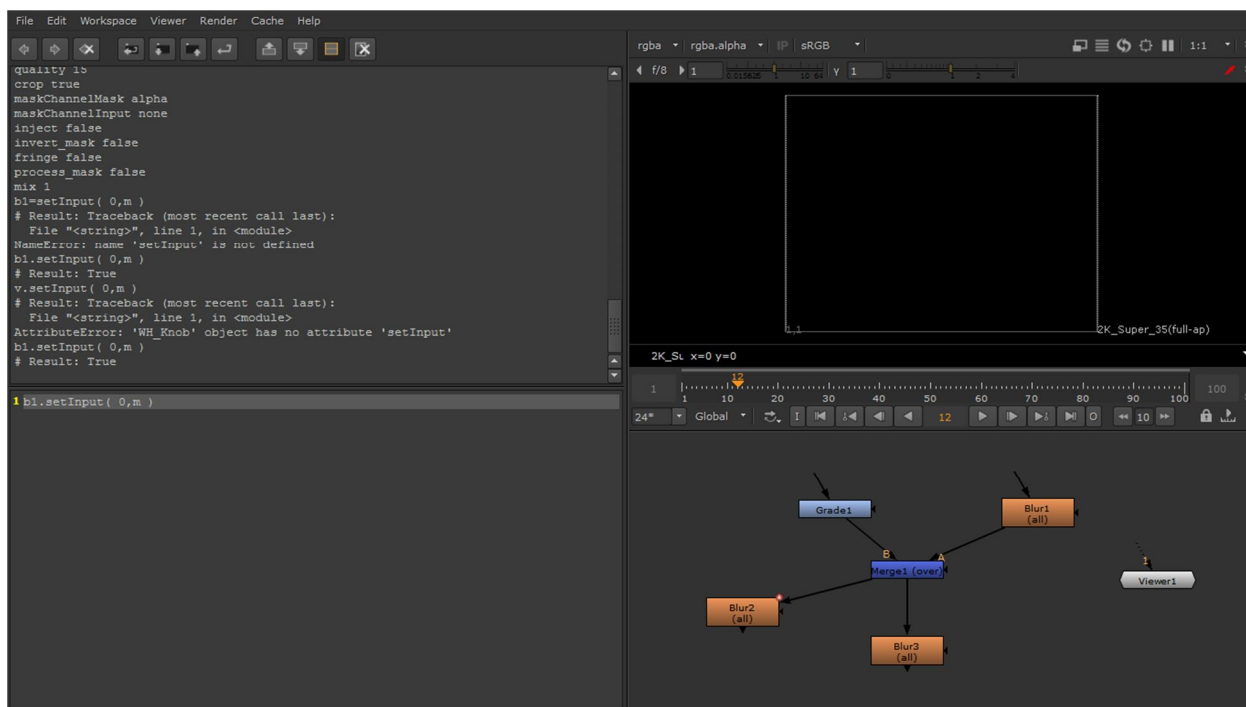


Рисунок 23 – Підключення ноди

Розглянемо ситуацію, коли необхідно виділити ноду, яка не підключена до інших елементів схеми. Наприклад під змінну b3 додамо ноду Transform

```
b3=nuke.nodes.Transform()
```

і далі виділяємо цю ноду

```
nuke.toNode("Transform1").setSelected(True)
```

де Transform1 – назва ноди на панелі NodeGraph. В результаті отримаємо виділену ноду (рисунок 24):

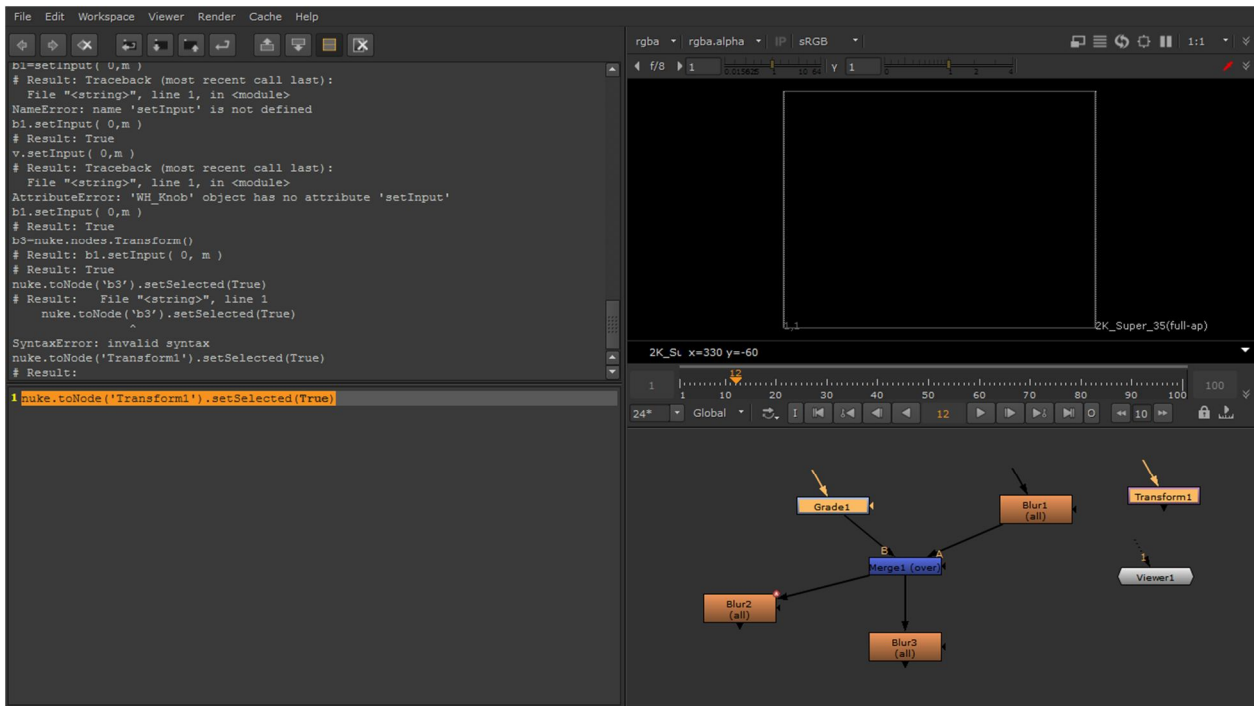


Рисунок 24 – Виділення ноди

І далі її слід присвоїти певній змінній

```
p=nuke.selectedNode()
```

і далі підключимо її до ноди Merge (рисунок 25)

```
p.setInput( 0, m )
```

де 0 – перший вузол нода.

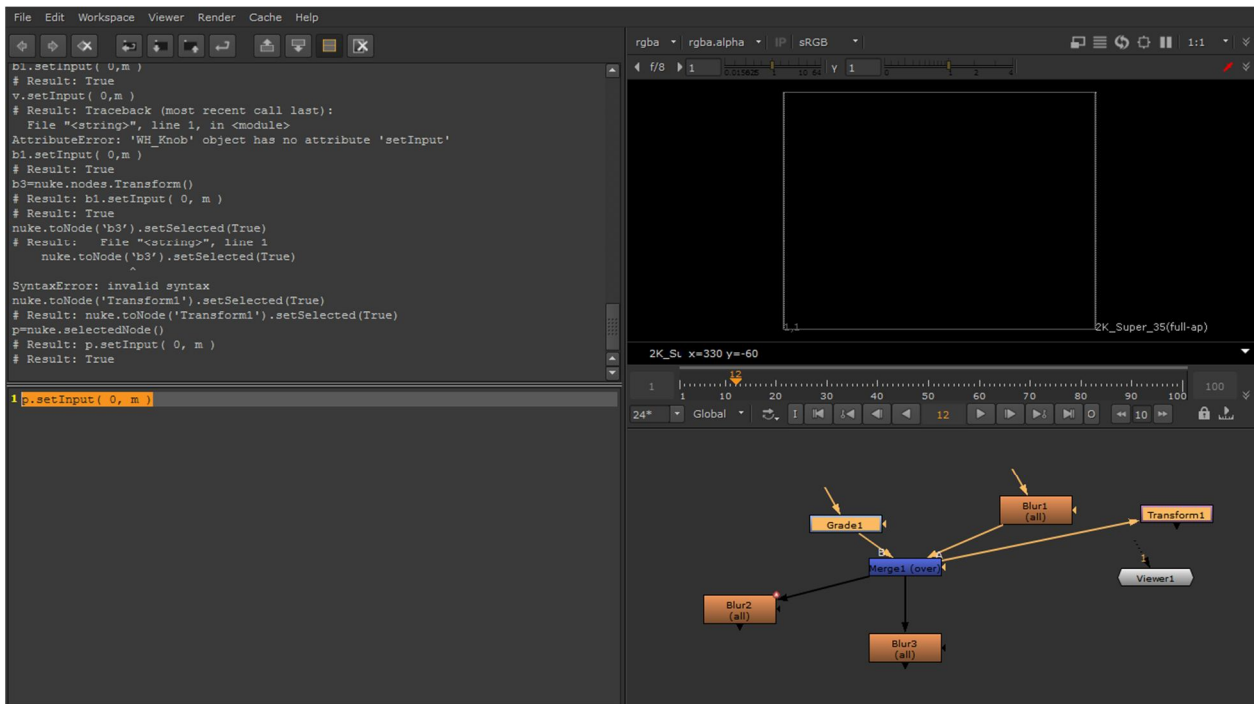


Рисунок 25 – Підключення ноди

Для того, щоб отримати ім'я виділеної ноди можна використати команду

```
nuke.selectedNode().name()
```

і у вікні результату буде показано ім'я виділеної ноди (рисунок 26)

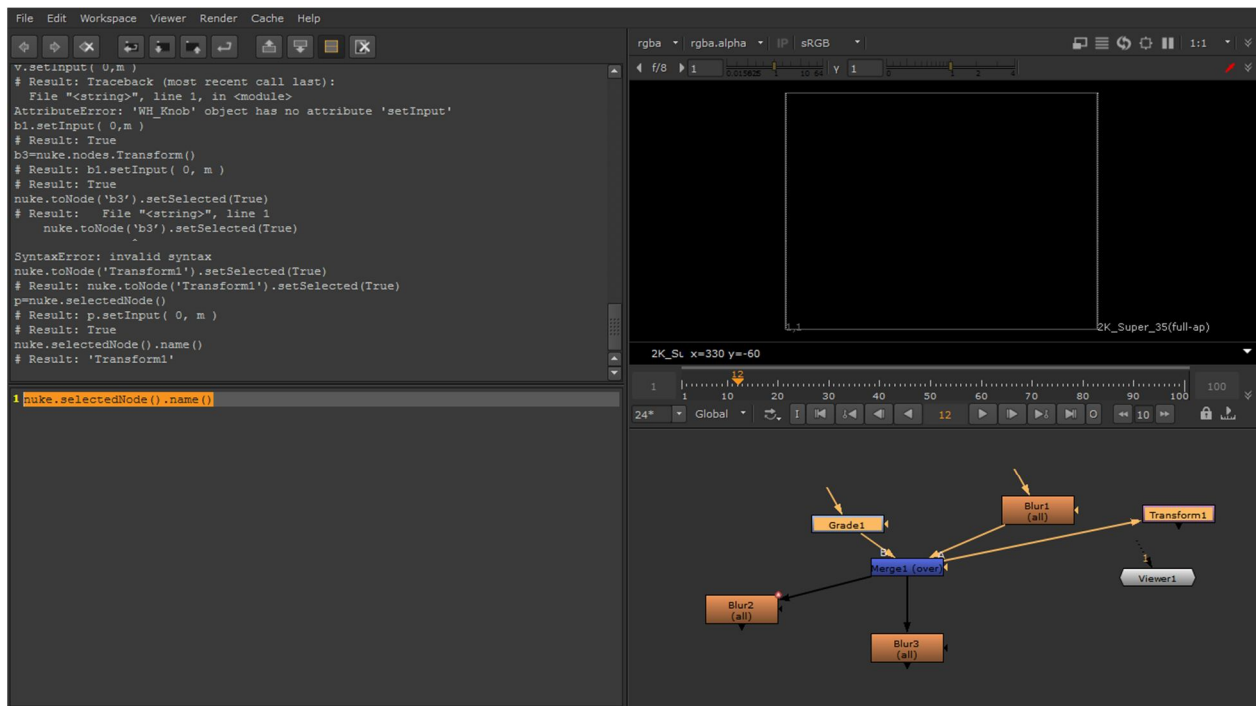


Рисунок 26 – Виділення ноди

і тут слід відмітити, що якщо виділено групу нодів, то буде показано лише ім'я останньої виділеної ноди. Для виділення усіх в групі треба записати таку команду:

```
nuke.selectedNodes()
```

і в результаті отримаємо (рисунок 27).

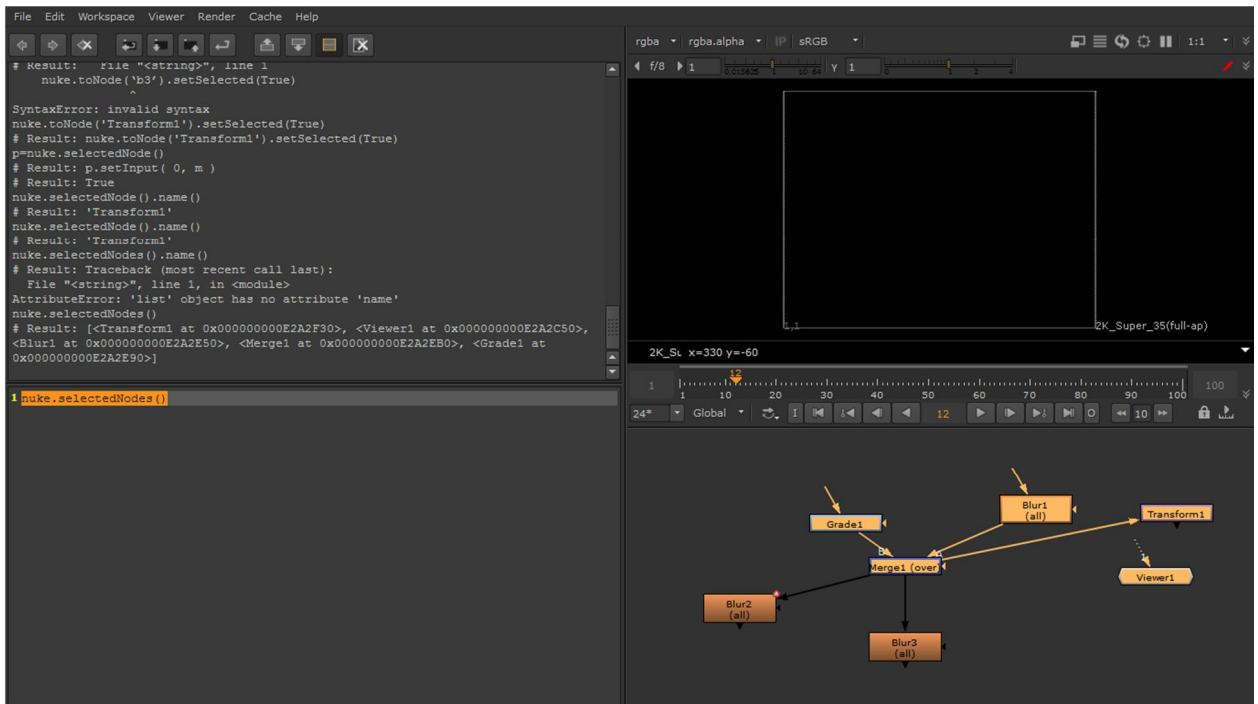


Рисунок 27 – Виведення назв групи нодів

5. Ноди Read та Write

Для додавання ноди Read використаємо вже відому команду, визначивши при цьому ще одну змінну:

```
r=nuke.nodes.Read(file='C:/Temp/1.png')
```

де вказуємо шлях до файлу, який хочемо вставити в блок-схему проекту. В результаті отримаємо в панелі NodeGraph нову ноду (рис.28).

Завдання. Розмістити ноду над нодою Blur 1 та підключити її до нодів Merge та Transform.

```

r.setXYpos( b.xpos(), b.ypos())
b.setInput( 0, r )
m=nuke.nodes.Merge()
m.setInput( 0, r )

```

Для того, щоб переглянути результат треба підключити ноду Read через ноду Merge до екрана viewer1. Тобто використаємо команду

```

nuke.connectViewer( 0, m )
де 0 – фон (рис.29).

```

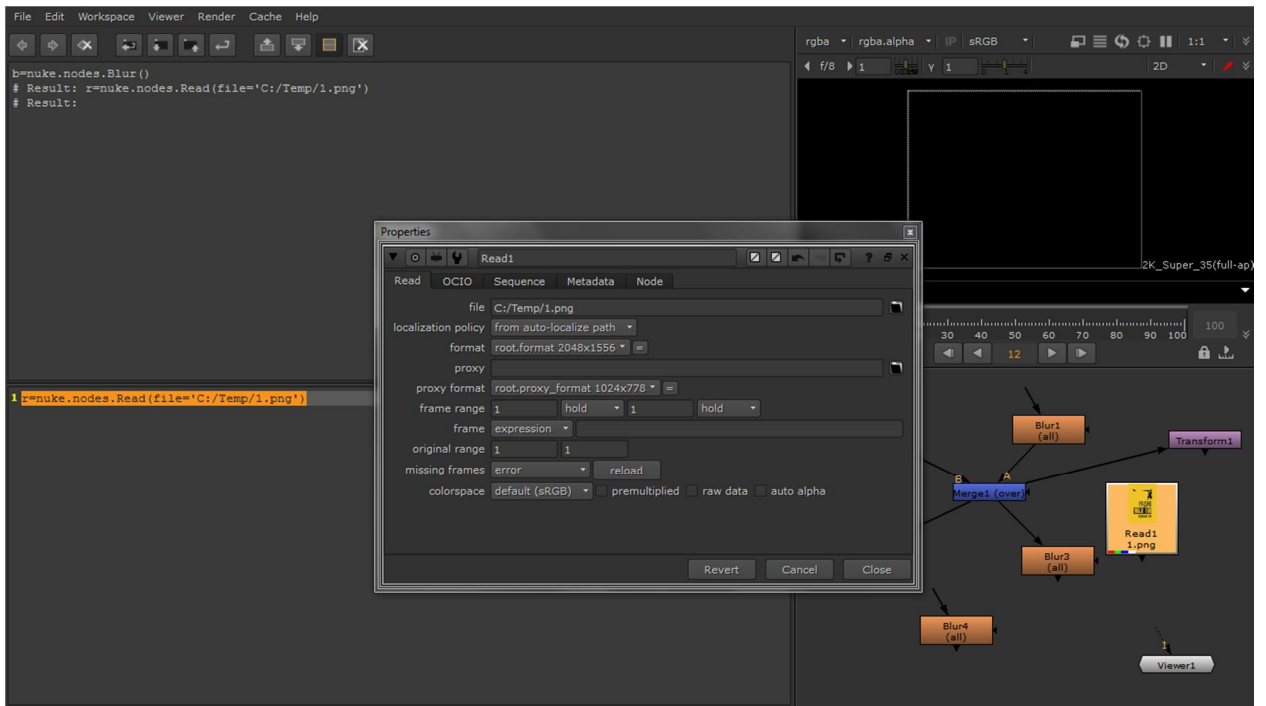


Рисунок 28 – Додавання ноди Read

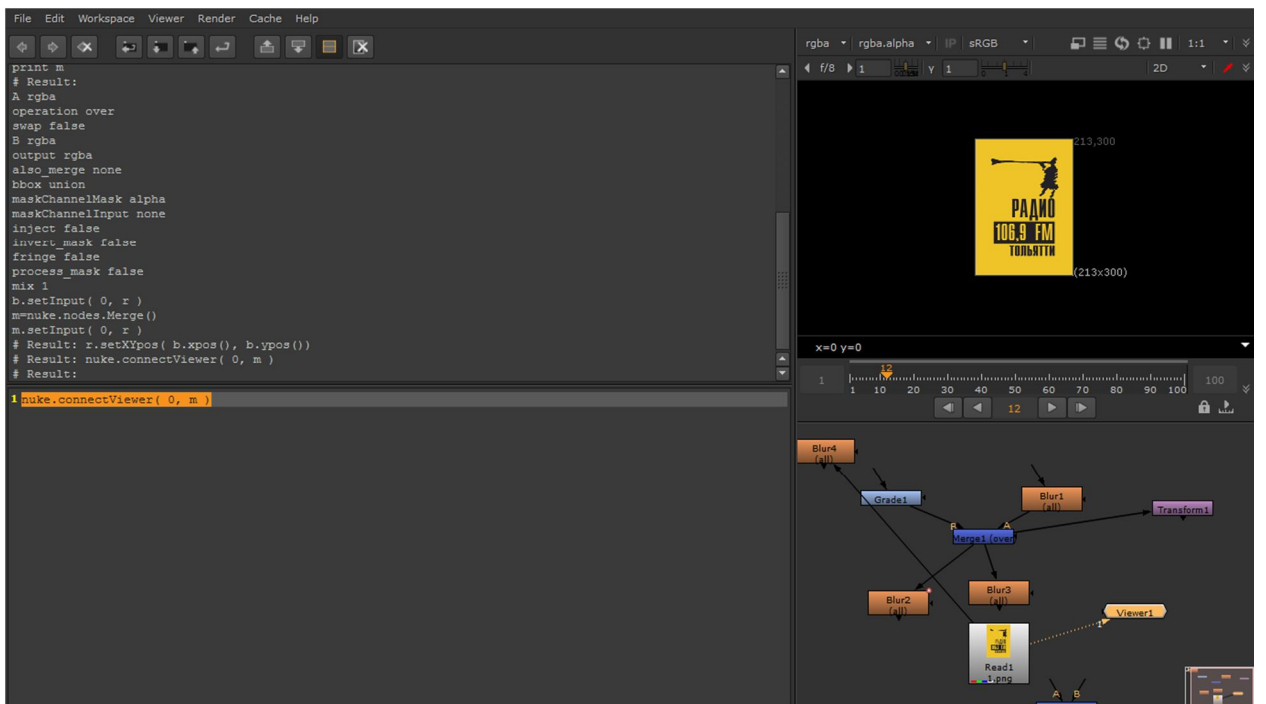


Рисунок 29 – Підключення зображення до екрану

Для запису цього зображення за певним напрямом (наприклад у ту саму папку) використаємо ноду Write. Для цього визначивши через змінну w зі збереженням, наприклад, секвенції файлів у форматі ##### (рис.30):

```
w=nuke.nodes.Write(file='C:/Temp/test.#####.jpg')
```

і підключимо її до ноди Blur (рис.31)

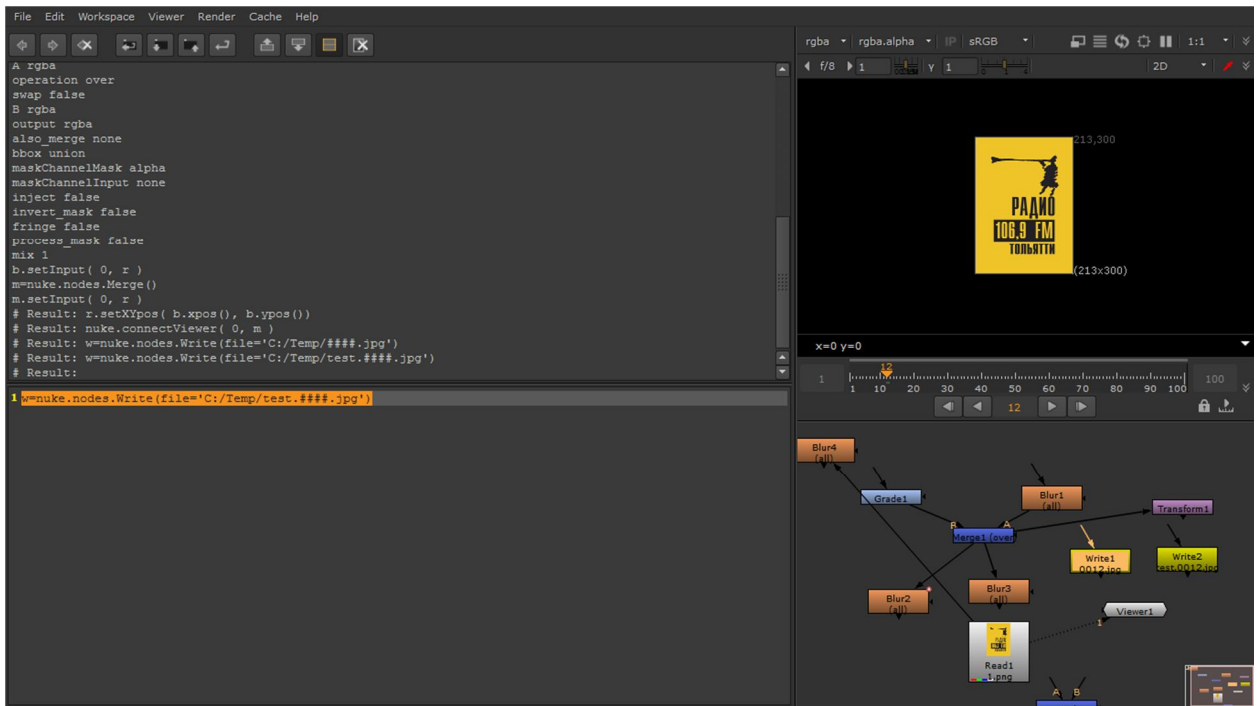


Рисунок 30 – Виведення ноди Write

```
w.setXYpos( b.xpos()-150, b.ypos())
w.setInput( 0, b )
```

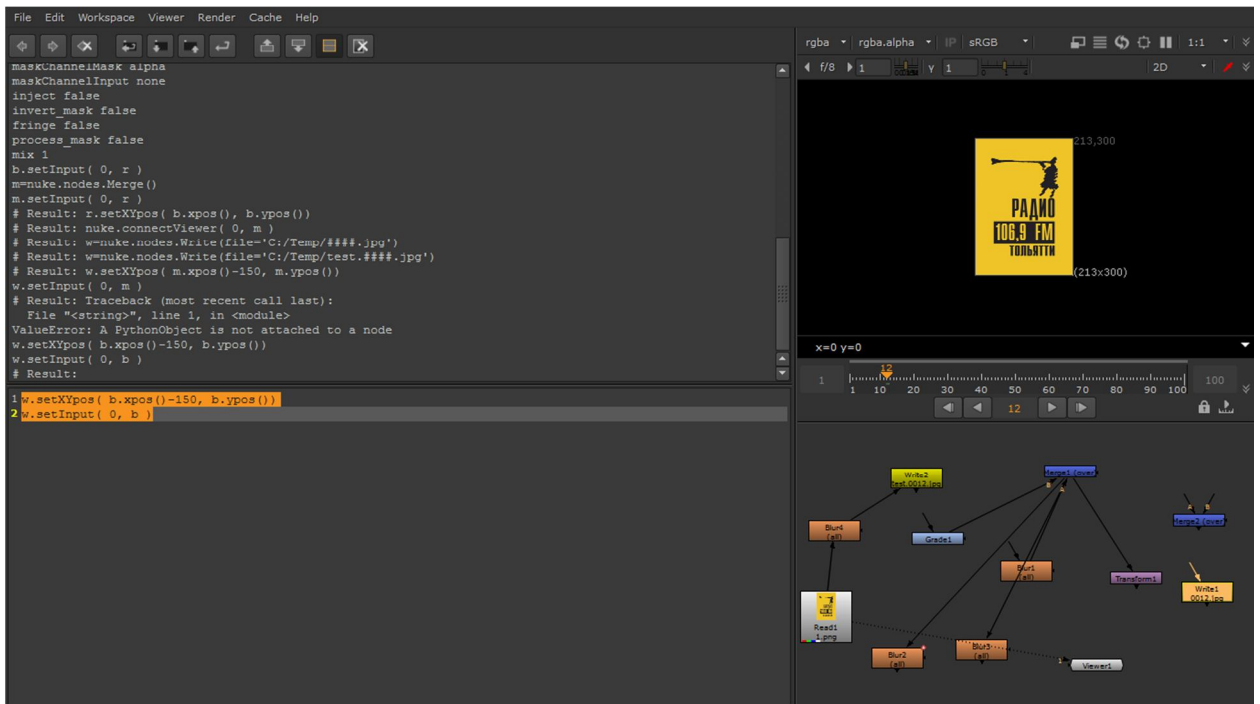


Рисунок 31 – Підключення ноди Write

І нарешті для записаного файлу у змінній w треба провести рендерінг сцени. Для цього використаємо команду execute або render, де у дужках спочатку вказуємо ім'я ноди, яка буде приймати участь у рендерінгу (у нас це нода Write2), далі треба вказати перший та останній кадр та необов'язково можна вказати по черговість

кадрів (крок рендерінгу, тобто які кадри будуть прорахуватись за номером починаючи з першого) (рисунок 32).

```
Nuke.render('Write2', 1, 50)
```

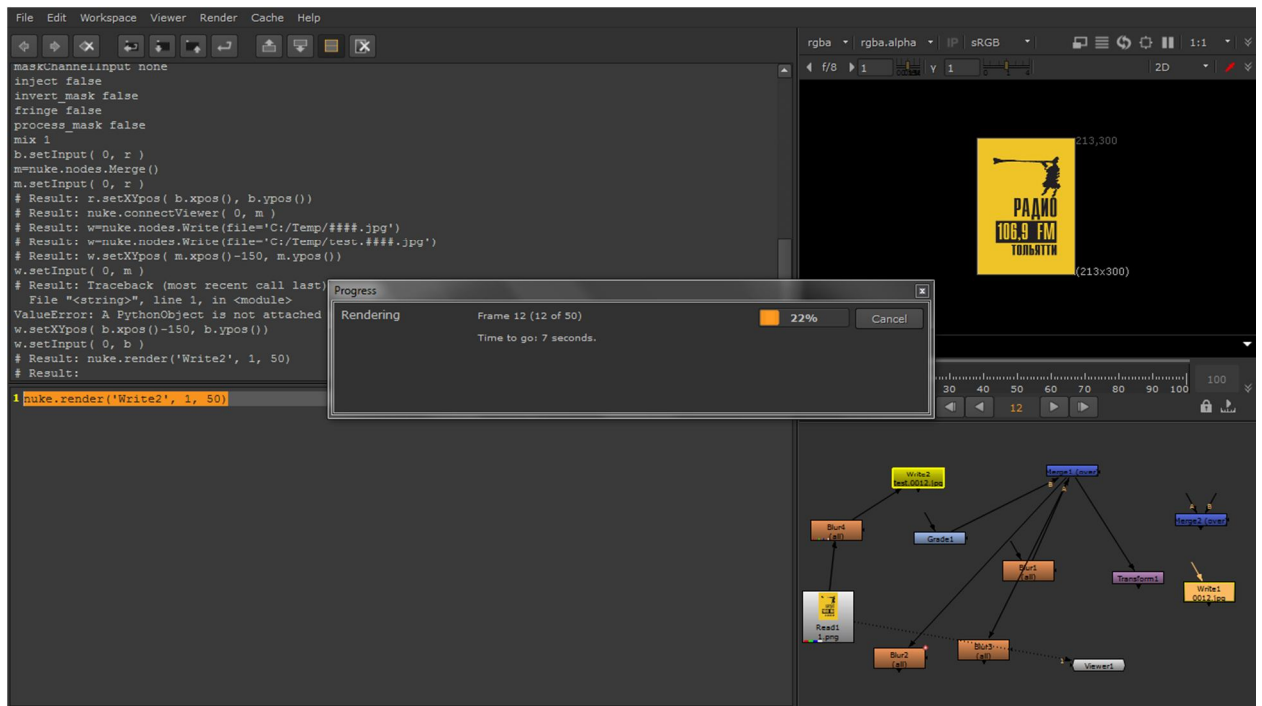


Рисунок 32 – Рендерінг

У папці яка була вказана буде створено розкадровані кадри (рис.33)

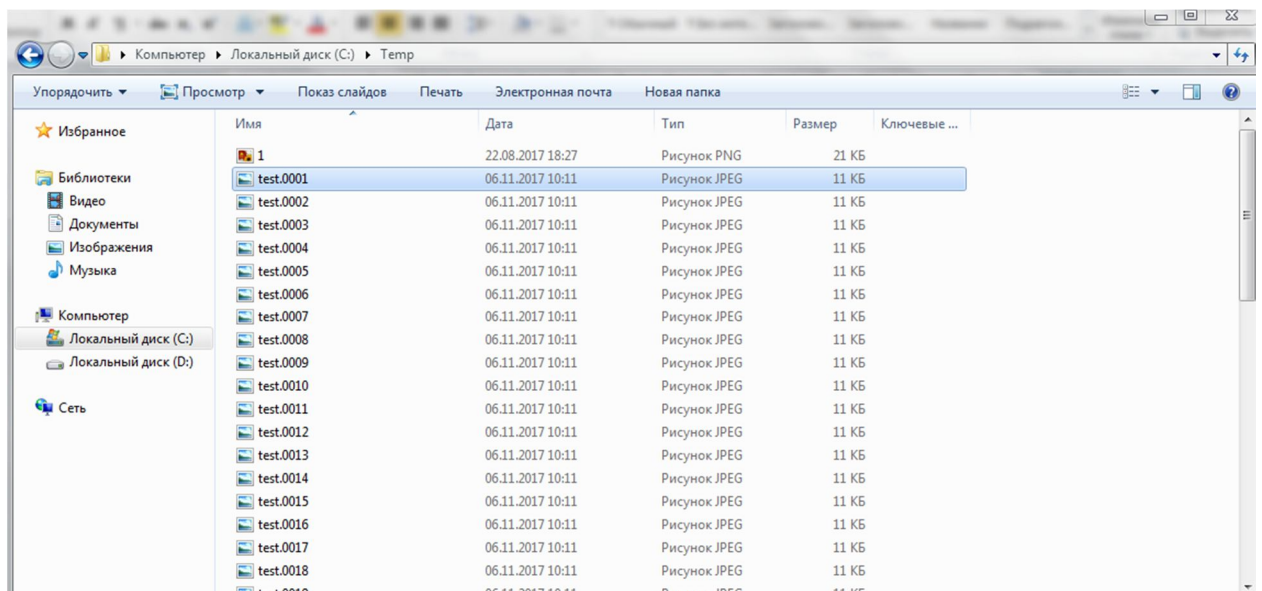


Рисунок 33 – Результат

Контрольні запитання

1. Які ви знаєте органи управління в режимі написання коду в програмі Nuke ? Як можна запустити один рядок коду, два рядки коду, видалити рядок, переглянути попередній рядок коду ?
2. Наведіть ключові особливості синтаксису команд в Nuke на основі мови програмування Python ?
3. Яке призначення має функція Knob ? Наведіть приклад скрипту з цієї командою.
4. Як при написанні скрипту можна використовувати змінні ?
5. Як для ноди можна змінити значення окремого параметру від того, який встановлено за замовчуванням ?
6. Яким чином проводять анімацію параметра ноди і як перевірити правильність коду ?
7. Яке призначення аргументу Inputs при написанні коду для окремо визначених вузлів ?
8. Як можна виділити декілька нодів автоматично ? Наведіть приклад скрипту.
9. Яке призначення має функція Read ? Наведіть приклад скрипту з цієї командою.
10. Яке призначення має функція Write ? Наведіть приклад скрипту з цієї командою.

КРИТЕРІЇ ОЦІНЮВАННЯ ТА ВКАЗІВКИ ПРО ПОРЯДОК ЗАХИСТУ ЛАБОРАТОРНИХ РОБІТ

Критерії оцінювання лабораторних робіт з дисципліни:

3 бали – під час захисту виконаної роботи студент відповідає правильно, у повному обсязі;

2 бали – під час захисту виконаної роботи студент допускає незначні неточності у відповідях;

1 бал – під час захисту виконаної роботи студент погано орієнтується в матеріалі;

0,5 бали – лабораторна робота виконана студентом у відведений розкладом час;

0 балів – лабораторна робота не виконана.

Під час захисту лабораторної роботи студент надає звіт, оформлений відповідно до вимог ДСТУ 3008-2015, який обов'язково повинен містити:

- титульний аркуш з номером та назвою лабораторної роботи;
- мету роботи;
- завдання на роботу;
- порядок виконання та результати роботи відповідно до завдання (індивідуально для кожної роботи ця частина звіту складається згідно з пунктом «Зміст звіту»);
- висновки.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Лутц М. Программирование на Python, 4-е издание / М. Лутц. – СПб.: Символ-плюс, 2011. – 992 с.
2. Ganbar R. Nuke 101. Professional compositing and visual effects / R. Ganbar. – NY.: Berkeley, 2011. – 404 p.
3. Кирьянов Д.В. Adobe Premiere CS3 и After Effects CS3 на примерах / Д.В. Кирьянов, Е. Н. Кирьянова. – СПб.: БХВ-Петербург, 2008. – 400 с.
4. Демидов Д.Г. Программные и аппаратные средства систем мультимедиа. Часть 1. Аппаратные средства / Д.Г. Демидов, А. М. Васьковский, А. Б. Николаев. – М.: МГУП им. Ивана Федорова, 2014. – 78 с.
5. Рейтц К. Автостопом по Python / К. Рейтц, Т. Шлюссер. – СПб.: Питер, 2017. – 336 с.