

Create and share your own interactive stories, games, music, and art

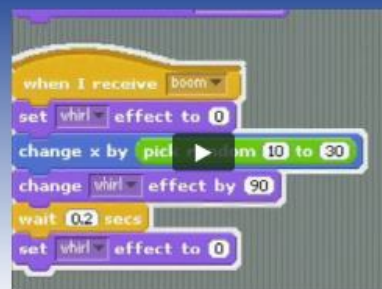
Check out the 1,424,007 projects from around the world!



To create your own projects:



Download Scratch



Настановва користувача

Скретч

версія 1.4

НАСТАНОВА КОРИСТУВАЧА



Зміст

Вступ	3
Основні поняття Скретч	5
Інтерфейс робочого вікна.....	6
Сцена.....	7
Нові виконавці	8
Список виконавців.....	9
Контейнер блоків та область Скрипти	10
Образи.....	12
Звуки.....	13
Поточні відомості	14
Стили повороту	15
Панель інструментів	15
Меню.....	16
Зелений прапорець.....	17
Вбудований графічний редактор	18
Величини.....	22
Типи даних	22
Організація даних	22
Вирази	25
Блоки.....	27
Типи блоків.....	27
Блоки реалізації базових структур алгоритмів	28
Події.....	30
Повний перелік команд.....	32

НАСТАНОВА КОРИСТУВАЧА



Вступ

Скретч (Scratch) – графічна мова та візуальне середовище об'єктно - орієнтованого програмування, які дозволяють легко створювати власні проекти: анімаційні та інтерактивні історії, ігри, моделі та розміщувати їх в мережі Інтернет.

Користувачам Скретч пропонує низьку підлогу (легко розпочинати), високу стелю (можна створювати складні комплексні проекти) та широкі стіни (підтримка великого спектру проектів).

Працюючи над Скретч - проектами, ви маєте можливість вивчити:

- принципи розробки сценаріїв ігрових, навчальних та інших проектів;
- основи об'єктно-орієнтованого програмування — поняття об'єкта, його властивості та дії над ним, взаємодія об'єктів;
- важливі алгоритмічні структури, такі як слідування, повторення та розгалуження;
- як створювати та опрацьовувати зображення з використанням вбудованого графічного редактора;
- роботу з веб-ресурсами — публікація, повторне використання, оцінювання;

Скретч, в інтерпретації команди розробників, — не тільки мова та середовище програмування, а й середовище для творчості, можливості самореалізації, тобто автори розглядають Скретч як навчальне діяльнісне середовище.

Скретч дозволяє реалізувати досить складні проекти з різних тем та предметних галузей — від найпростіших малюнків і презентаційних роликів до комплексних моделей фізичних та біологічних процесів. При цьому учні виступають в ролі вчених та винахідників: планують та здійснюють експерименти, розробляють моделі, висувають гіпотези та перевіряють їх на практиці. Це безперервна спіраль:

- генерація ідей (навчальних, пізнавальних, соціальних і т.п.) та визначення очікуваних результатів;
- створення проекту як моделі реалізації власних ідей;
- дослідження результатів власної діяльності;
- аналіз та обговорення результатів власної діяльності у спільноті (мережевій в тому числі);
- залучення до співпраці однодумців;

НАСТАНОВА КОРИСТУВАЧА



- генерація нових ідей, зокрема ідей нових проектів, набуття досвіду створення проектів, прийняття рішень, ефективної комунікації та роботи в команді.

Основні особливості Скретч:

- **Блочне програмування.** При створенні програм необхідно просто з'єднати різнокольорові графічні блоки (команди). Форма блоків розроблена таким чином, що їх можна зібрати лише в синтаксично правильні конструкції. Змінювати блоки можна й під час виконання програми, що дозволяє досліджувати роботу програми та більше експериментувати з новими ідеями.
- **Маніпуляції даними.** При створенні проектів можна одночасно використовувати графіку, анімацію, музику та звуки. У такий спосіб розширюються можливості щодо управління медіа-даними.
- **Спільна робота та обмін.** На сайті проекту Скретч (<http://scratch.mit.edu>) розміщено колекцію готових проектів, які можна переглянути, завантажити та використати у якості прикладів чи заготовок для розробки; на цьому ж сайті можна опублікувати власні проекти. У такий спосіб, навколо проекту Скетч, формується міжнародна спільнота користувачів.

В даному керівництві розглянуто основні можливості середовища Скретч версії 1.4, реліз якої відбувся в липні 2009 року. Адаптація та локалізація для України Галини Проценко (Печерська гімназія №75 м. Києва).

Завантажити середовище можна з сайту Скретч (<http://scratch.mit.edu>). На цьому ж сайті (<http://info.scratch.mit.edu/Support/>) ви можете знайти багато додаткових англomовних матеріалів: відео уроки, зразки проектів, картки, форум тощо.

На сайті ВікіОсвіта (http://eduwiki.urau.net.ua/wiki/index.php/Українська_спільнота_Scratch) ви можете познайомитися з україномовними матеріалами щодо використання Скретч в навчально-виховному процесі: настанови по інсталюванню та роботі з середовищем, методичні рекомендації, приклади проектів, список корисних посилань та ін.

НАСТАНОВА КОРИСТУВАЧА



Основні поняття Скретч

Проект (англ. projects) - об'єкти, скрипти, звуки, які зберігаються в окремому файлі та використовуються разом.



В Скретч за замовчуванням використовується спеціальний виконавець команд (англ. *sprite*, *спрайт*, *персонаж*, *герой*) - *Рудий кіт*. Він може рухатися, говорити та думати, змінювати зовнішній вигляд та взаємодіяти з іншими виконавцями на *сцені*.

Інших виконавців можна вибрати зі стандартної галереї, намалювати самостійно у вбудованому графічному редакторі (англ. *Paint Editor*), завантажити з комп'ютера або знайти в мережі Інтернет .

Зовнішній вигляд виконавців можна змінювати, використовуючи різні **образи** (англ. *costume*). Образи, як і виконавців, можна вибирати зі стандартної галереї, намалювати або завантажити.

При розробці проекту для кожного з виконавців необхідно створити скрипт (сценарій, метод), вказавши що і в якій послідовності робити, наприклад, як і куди рухатися, як реагувати на дії інших виконавців тощо.

Скрипт (англ. scripts) — послідовність команд, що визначає, які дії і в якому порядку слід виконати виконавцям.

Скрипт складається зі стеків та може містити кілька стеків.

Стек (англ. stack) - набір послідовно з'єднаних різнокольорових графічних блоків (цеглинка, команда, вказівка) в межах однієї події.

Блок (англ. blocks) – окрема команда, яка використовується в програмі.

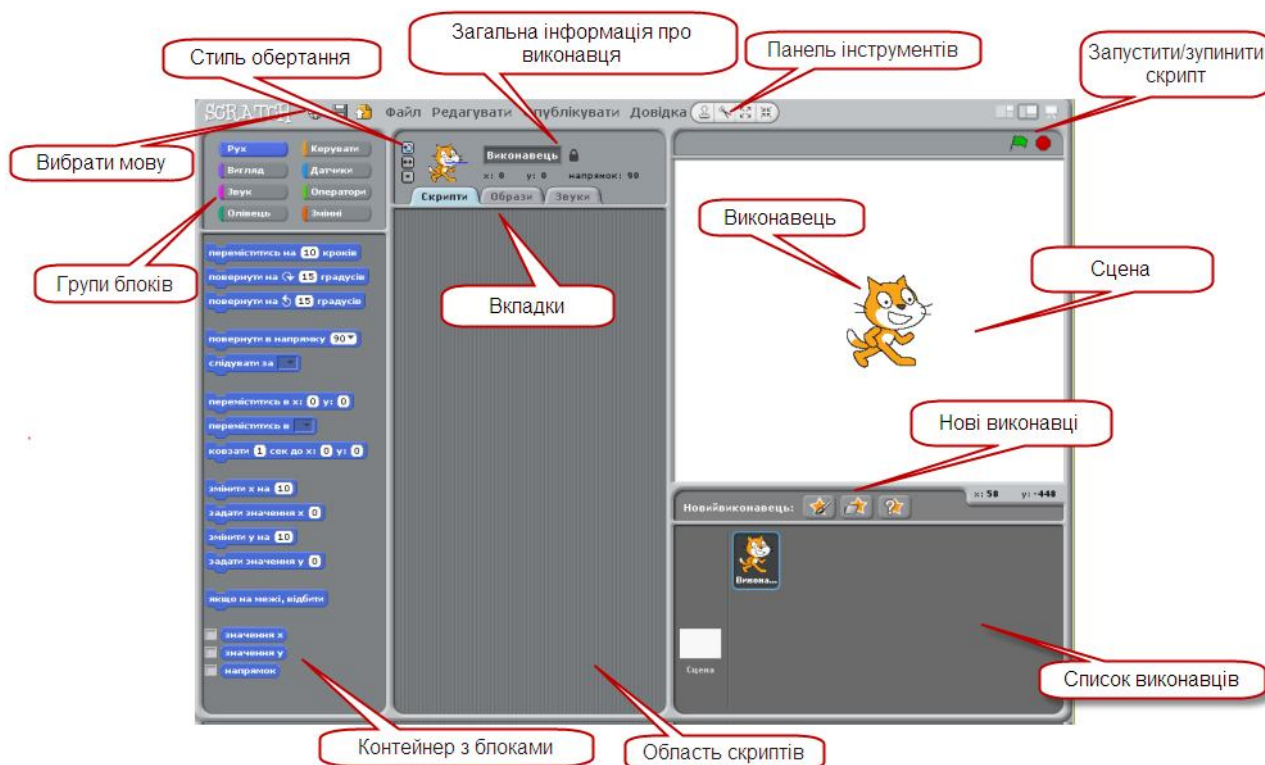
Події в середовищі Скретч - натискання на задану клавішу (наприклад, пропуск), клацання по виконавцю або по сцені, надходження повідомлення від іншого виконавця і т.п.

НАСТАНОВА КОРИСТУВАЧА



Інтерфейс робочого вікна

При запуску середовища Скретч відкривається програмне вікно з новим проектом, який містить одного виконавця.



В правій верхній частині робочого вікна розміщено сцену, на якій знаходиться Рудий кіт. Саме на сцені відбуваються всі події, на ній розміщуються інші виконавці, для кожного проекту добираються декорації (фон).

В правій нижній частині робочого вікна можна побачити усіх виконавців проекту, їх імена та кількість скриптів. У цій області всі виконавці проекту відображаються у вигляді ескізів.

Виконавцям і фону можна задати команди. Команди (блоки) зібрані в різнокольорові шухлядки (групи), які розташовані в верхній лівій частині робочого вікна. Якщо відкрити шухлядку (вибрати групу блоків), то у лівій нижній частині вікна можна побачити всі команди (контейнер з блоками).

Для того, щоб зібрати з окремих блоків сценарій (скрипт) слід перетягнути блоки в центральну частину робочого вікна.

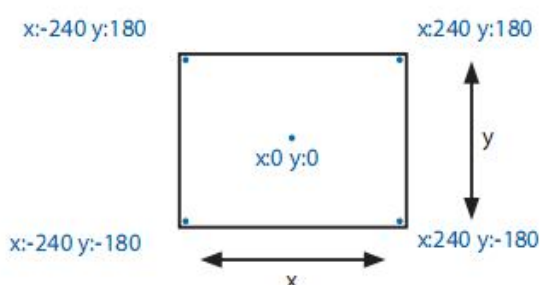
НАСТАНОВА КОРИСТУВАЧА



Сцена

Сцена (*stage*) – це місце, де розгортаються основні події проектів.

На початковому етапі створення проекту сцена являє собою прямокутну область білого кольору, але її вигляд (фон) можна змінити — намалювати власний малюнок у вбудованому редакторі, завантажити зображення, вибрати фон з стандартної колекції.



Розміри сцени 480 x 360 одиниць. Одиниця є умовною і не відповідає жодній одиниці вимірювання.

На сцені розміщуються виконавці, місце розташування яких можна змінювати перетягуванням мишкою.

Положення виконавця та вказівника мишки на сцені визначається координатами (x, y). Координати точки в центрі сцени мають значення $x=0$, $y=0$. Значення координат вказівника мишки відображаються в правому нижньому куті сцени.

x: 146 y: -99

Масштаб відображення сцени можна змінити.



Зменшити масштаб відображення сцени (використовується при роботі на нетбуках)



Відновити масштаб відображення сцени (стандартний режим роботи)

Передбачено можливість перегляду проекту в повноекранному режимі.



Перегляд проекту в повноекранному режимі. Для завершення перегляду слід натиснути клавішу *Esc* або вибрати кнопку (Закінчити перегляд).

НАСТАНОВА КОРИСТУВАЧА



Нові виконавці

Новий проект завжди містить одного виконавця Рудого кота. Існує кілька способів створення інших (нових) виконавців:



Намалювати нового виконавця. Використовується вбудований графічний редактор (Paint Editor).



Вибрати нового виконавця з файлу. Вибрати виконавця можна з стандартної галереї (папка *Новий виконавець*) або скористатися власними заготовками (графічні файли наступних форматів JPG, BMP, PNG, GIF).



Отримати нового виконавця випадковим чином. Новий виконавець вибирається випадковим чином зі стандартної галереї

Виконавці в стандартній галереї зібрані в наступні папки:

Animals (Тварини),

Fantasy (Фантастичні істоти),

Letters (Літери),

People (Люди),

Things (Речі),

Transportation (Транспорт).



У якості нового виконавця можна використовувати частину сцени. Для того, щоб створити такого виконавця слід викликати контекстне меню **Сцени** та вибрати команду **використати частину екрану для нового виконавця**. Після цього слід виділити потрібну область.

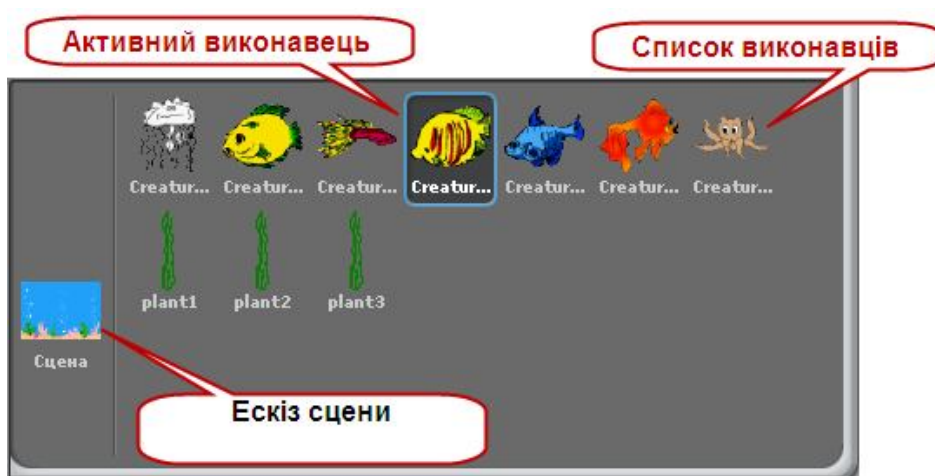
НАСТАНОВА КОРИСТУВАЧА



Список виконавців

Виконавці та сцена у проекті, подібно до персонажів у мультфільмах, представлені у вигляді ескізів в області *Список виконавців*.

Кожен ескіз супроводжується наступною інформацією: назва та кількість скриптів.



Для перегляду та редагування образів, скриптів, звуків окремого виконавця слід вибрати його в списку. Такий виконавець називається **активним**. Навколо ескізу активного виконавця з'являється синя рамка.

Змінити порядок слідування виконавців можна перетягуючи їх у даній області.

Для відображення прихованих виконавців, експорту, дублювання, вилучення слід скористатися відповідними командами контекстного меню. Відобразити прихованих виконавців також можна за допомогою комбінації Shift + клацнути по ескізу.

Інший спосіб перегляду та редагування - на сцені вибрати виконавця (підвести вказівник мишки до виконавця та двічі клацнути лівою кнопкою).

Змінити також можна і сцену. Для перегляду та редагування скриптів, фонів, звуків, які пов'язані зі сценою, слід вибрати ескіз сцени.

НАСТАНОВА КОРИСТУВАЧА






Контейнер блоків та область Скрипти

В верхній лівій частині робочого вікна середовища розташовано 8 різнокольорових кнопок, які дозволяють відкрити потрібну групу блоків. Блоки вибраної групи відображаються в контейнері блоків (ліва нижня частина робочого вікна).

Для того, щоб задати команду виконавцю, слід

1. Відкрити відповідну групу блоків.
2. У контейнері блоків (*Blocks Palette*) вибрати потрібний блок та перетягнути його в область *Скрипти*.

Деякі блоки мають поля введення (округлені прямокутники), наприклад, . Такі поля призначені для введення/редагування значень. Для того, щоб ввести значення з клавіатури, необхідно встановити курсор в поле та набрати значення. В таке поле також можна помістити інші блоки, які мають таку ж округлену форму.

В деяких блоках значення вибирається зі списку, що відкривається, наприклад,  (натисніть ▼ та серед списку значень виберіть потрібне). Значення в деяких блоках можна як вводити в поле, так і вибирати зі списку, наприклад, .

Запуск блоку - підведіть до блоку вказівник мишки і двічі клацніть лівою кнопкою.

Блоки об'єднуються в стек.

Стек (англ. *stack*) - набір послідовно з'єднаних різнокольорових графічних блоків в межах однієї події.

Події в середовищі Скретч – натискання на задану клавішу (наприклад, пропуск), клацання по виконавцю або по сцені, надходження повідомлення від іншого виконавця і т.п.

При додаванні блока в уже готовий стек з'являється біла смужка. Вона показує можливе місце розташування та дозволяє коректно об'єднати блоки.



НАСТАНОВА КОРИСТУВАЧА



Запуск стека – підведіть до будь-якого блоку в стеку вказівник мишки і двічі клацніть лівою кнопкою (блоки виконуються послідовно у порядку їх розташування зверху вниз).

Стек можна переміщувати в області *Скрипти*, утримуючи лівою кнопкою мишки верхній блок.

Якщо вибрати блок всередині стека, та перемістити його, то разом з ним перемістяться і всі блоки, що розташовані нижче.

Ви можете скопіювати стек та використати його для іншого виконавця – просто перетягніть стек на ескіз виконавця в області *Список виконавців*.

Скрипт (англ. scripts) — послідовність команд, що визначає, які дії і в якому порядку слід виконати виконавцям.

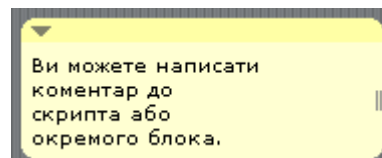
Скрипт складається зі стеків. Скрипт може містити кілька стеків.

Видалити блоки з області **Скрипти** можна просто перетягнувши їх назад в контейнер блоків.

Розглянемо команди контекстного меню області Скрипти:

- **очистити** – впорядкувати розташування стеків
- **зберегти зображення скрипта** – зробити та зберегти скриншот скрипта у форматі gif
- **додати коментар** – прокоментувати стек або окремий блок. З'явиться жовта область коментарів і ви можете ввести текст.

Щоб змінити ширину області коментарів, використовуйте позначку, що розташована праворуч.



Щоб згорнути/розгорнути область коментарів, використовуйте трикутник, який розташовано в верхньому лівому куті області коментарів.

Переміщувати коментарі можна перетягуванням.



НАСТАНОВА КОРИСТУВАЧА



Образи

Зовнішній вигляд виконавців можна змінювати, використовуючи різні **образи** (*costume*). За звичай різні образи використовуються для створення ефекту анімації. Зміна образів відбувається при виконанні скрипта.

Для перегляду, створення або редагування образів слід перейти на вкладку **Образи** (*Costumes*).



Рудий кіт за замовчуванням має два образи, які можна використовувати для імітації ходи.

Кожний образ має порядковий номер та назву. Порядок слідування образів можна змінити, переміщуючи їх на вкладці за допомогою мишки. При цьому автоматично змінюється нумерація образів. Назву образу також можна змінити – встановіть курсор у відповідне поле та введіть нове значення.

Навколо активного образу з'являється синя рамка. Для того, щоб зробити інший образ активним, слід його вибрати.

Для зміни або доповнення готового образу необхідно вибрати кнопку **Редагувати** (*Edit*). За допомогою кнопки **Копіювати** (*Copy*) створюється точна копія образу.

Команда контекстного меню **перейти до нового виконавця** дозволяє перетворити образ у виконавця – образ переміститься в область **Список виконавців**.

Команда контекстного меню **зберегти образ** дозволяє зберегти образ як окремий файл.

НАСТАНОВА КОРИСТУВАЧА



Створити новий образ можна кількома способами:


Малювати

Намалювати новий образ. Використовується вбудований графічний редактор.

Вставка

Вибрати новий образ з файлу. Вибрати образ можна з стандартної галереї (папка *Образи*) або скористатися власними заготовками (графічні файли наступних форматів JPG, BMP, PNG, GIF).

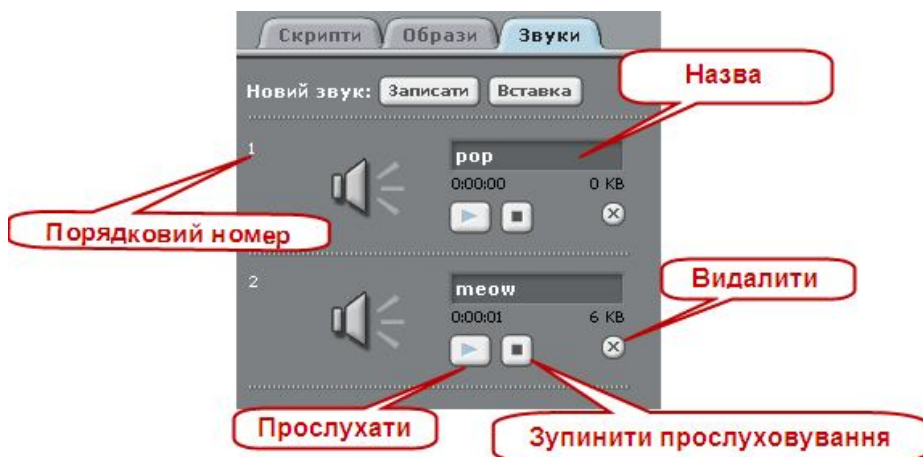
Камера

Створити новий образ з використанням веб-камери. Для того, щоб отримати зображення, у вікні Камера, що відкриється, слід вибрати кнопку  або натиснути клавішу *пропуск*.

Перетягнути зображення з Робочого стола

Звуки

Для прослуховування, запису або вставляння звуків слід перейти на вкладку **Звуки** (*Sounds*).



Виконавці можуть відтворювати звукові файли у форматі MP3 та не стиснуті файли у наступних форматах: WAV, AIF и AU.

НАСТАНОВА КОРИСТУВАЧА



Поточні відомості



Поточні відомості про виконавця містять наступні дані: ім'я, координати точки, в якій він знаходиться, напрямок руху, можливість переміщення перетягуванням мишкою при виконанні проекту.

Ім'я виконавця за потреби можна змінити.

Напрямок вказує куди і під яким кутом буде рухатися виконавець при виконанні блоків групи

Рух :

- 0 — виконавець рухається вгору;
- 90 — виконавець рухається праворуч;
- 180 — виконавець рухається вниз;
- 90 — виконавець рухається ліворуч.

Синя лінія біля виконавця також вказує напрямок руху виконавця. Змінюючи положення лінії, можна змінювати кут та напрямок руху.

У процесі перегляду проекту на веб-сайті або у повноекранному режимі, можна дозволити/заборонити перетягуванням мишкою змінювати місце розташування виконавця на сцені.



Заборонити, використовується за замовчуванням



Дозволити.

НАСТАНОВА КОРИСТУВАЧА



Стилі повороту

Стилі повороту (*Rotation Style*) дозволяють задати поведінку виконавця/образу при зміні напрямку руху.



Можна обертатись (*Can rotate*). При зміні напрямку руху виконавець може дзеркально відображатись (горизонтальне та вертикальне дзеркало).



Повертати тільки праворуч/ліворуч (*Left-right flip*). Виконавець може тільки горизонтально відображатись.



Не повертатись (*No-rotate*). Виконавець не може обертатись при зміні напрямку руху.

Панель інструментів

Для видалення, дублювання, зміни розмірів об'єкту використовуються кнопки панелі інструментів



Виберіть потрібний інструмент на панелі, а потім – відповідний об'єкт.

Призначення кнопок панелі інструментів:

Дублювати (штамп) - зробити копію виконавця, образу, звуків, блоків та скриптів.

Видалити (ножиці) – вирізати (видалити) виконавця, образ, звук, блоки та скрипти.

Збільшити – пропорційно збільшити розміри (Shift + вибрати виконавця та змінити розмір з більшим кроком)

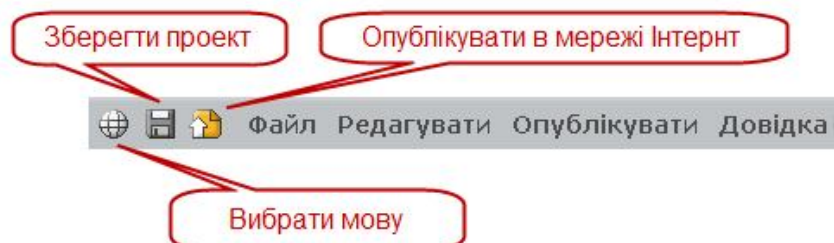
Зменшити - пропорційно зменшити розміри (Shift + вибрати виконавця та змінити розмір з більшим кроком)

При роботі з об'єктами можна також використовувати відповідні команди контекстного меню (права кнопка мишки).

НАСТАНОВА КОРИСТУВАЧА



Меню



Команди пункту меню **Файл**

Команди пункту меню **Файл** дозволяють створити новий проект та зберегти його, відкрити готовий проект, зберегти проект в іншій папці або з іншим ім'ям.

Імпортувати проект — перенести всіх виконавців та фони іншого проекту в поточний проект. Ця функція використовується для об'єднання виконавців з різних проектів в новому проекті.
(**Import Project**)

Зберегти виконавця — зберегти поточного виконавця в окремому файлі, який потім можна вставити в інший проект.

Опис проекту — зробити нотатки до проекту, наприклад, описати призначення проекту, як з ним працювати тощо.

Завершити роботу — вийти з середовища

Команди пункту меню **Редагувати** :

Відновити — відмінити останнє видалення (відновити блок, скрипт або виконавця).

Розпочати покрокове виконання — виконати скрипт покроково. Під час виконання відповідний блок підсвічується.

Задати одиничний крок — вибрати швидкість кроку (повільно, нормально, швидко, підвищена швидкість)

Показати блоки двигуна — використовується при підключеному до комп'ютера набору з робототехніки (LEGO® Education WeDo™
(<http://www.legoeducation.com>).

Стиснути звуки або картинки — зменшити розмір звукового або графічного файлу.

НАСТАНОВА КОРИСТУВАЧА



Команди пункту меню **Опублікувати**:

Опублікувати цей проект в мережі — Завантажити поточний проект на сайт Скретч (<http://scratch.mit.edu>). При завантаженні необхідно ввести логін та пароль, вказати назву проекту та теги (ключові слова для пошуку), описати.

На сайт Скетч — Перейти на домашню сторінку сайту Скретч.


Пункт меню **Довідка**:

Веб-сторінка довідки — відкрити довідкові матеріали на сайті Скретч

Довідка в малюнках — відкрити веб-сторінку з довідковими матеріалами

Про Скретч — переглянути короткі відомості про Скретч

Зелений прапорець

Кнопка  (Запустити скрип) розташована в верхньому правому куті сцени та використовується для одночасного запуску на виконання всіх скриптів.

Якщо її натиснути, то запустяться на виконання всі стеки першим блоком яких є блок



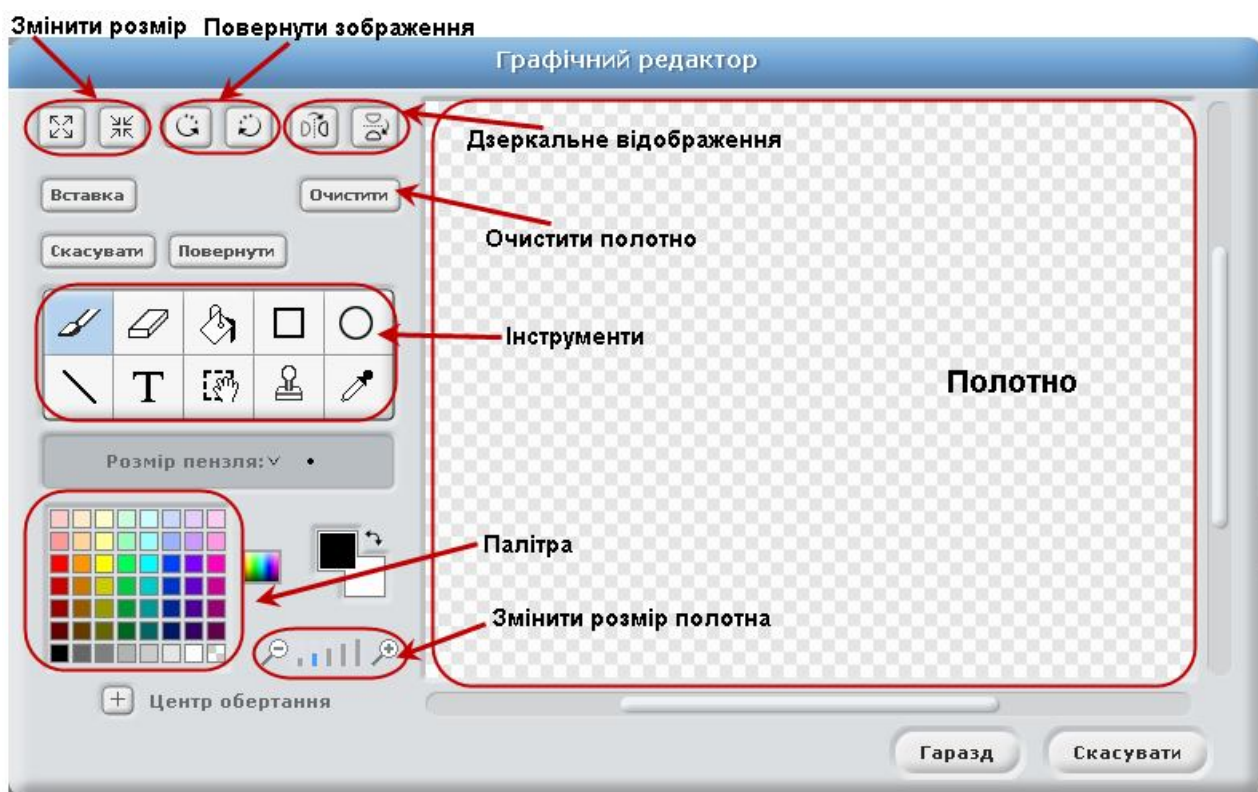
Якщо скрипт містить кілька стеків з таким блоком, то вони запускаються на виконання одночасно та виконуються паралельно.

НАСТАНОВА КОРИСТУВАЧА



Вбудований графічний редактор

Графічний редактор дозволяє створювати або змінювати виконавців, образи, фони.



Розглянемо призначення кнопок панелі **Інструментів**. Зазначимо, що при виборі деяких інструментів у нижній частині панелі з'являється додаткова панель, яка дає змогу змінити властивості вибраного інструмента.

Інструменти вільного малювання



Пензель
(Paintbrush)

Проведення ліній, товщина яких вибирається на додатковій панелі.

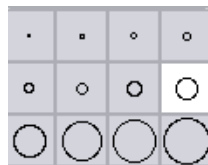


НАСТАНОВА КОРИСТУВАЧА



Стирчка (Eraser)

Стирання відповідної області зображення. Розмір стирачки вибирають на додатковій панелі.



Інструменти малювання графічних примітивів



Прямокутник (Rectangle)

Малювання прямокутника (квадрата при натиснутій клавіші Shift). Спосіб заповнення внутрішньої області вибирають на додатковій панелі:



- малювання контуру фігури і заповнення внутрішньої області основним кольором;
- малювання контуру фігури основним кольором без заповнення внутрішньої області.



Еліпс (Ellipse)

Малювання **еліпса** (кола при натиснутій клавіші Shift). Спосіб заповнення внутрішньої області вибирають на додатковій панелі



Лінія (Line)

Для проведення прямої лінії (горизонтальна або вертикальна при натиснутій клавіші Shift). Товщину лінії вибирають на додатковій панелі.

Інші інструменти



Виділити область (Selection)

Виділення прямокутної області зображення. Виділену область можна переміщувати, копіювати, видаляти (клавіша Delete) або вирізати (клавіші Shift + Backspace).

НАСТАНОВА КОРИСТУВАЧА



Текст (Text)

Дозволяє додавати до малюнка текстовий напис. При роботі з текстовими написами можна вибрати шрифт та розмір літер.



Штамп (Stamp)

Дублювання частини зображення. Спочатку необхідно вибрати інструмент, потім виділити область та штампувати.



Піпетка (Eyedropper)

Вибір кольору, який вже використовується на малюнку. Цей колір буде встановлено як основний при натисненні лівої кнопки мишки і як фоновий – при натисненні правої кнопки

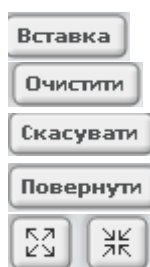


Заливачка (Fill)

Заповнення замкнутої області кольоровою фарбою. Спосіб заповнення внутрішньої області вибирають на додатковій панелі



Призначення кнопок робочого вікна графічного редактора:



Вставити зображення з файлу

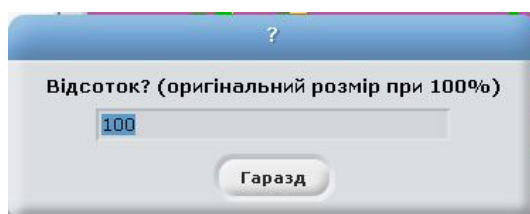
Очистити полотно (всі зображення видаляються)

Скасувати останню дію, якщо вона неправильна.

Повернути результат скасованої дії

Збільшити або зменшити зображення. Для цього слід виділити потрібну область полотна та вибрати відповідну кнопку.

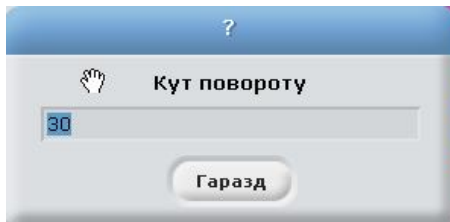
Ви можете задати значення на яке потрібно зменшити/збільшити виділену область. Для цього утримуючи клавішу SHIFT виберіть потрібну кнопку. У вікні, що відкрилося, задайте потрібне значення та виберіть кнопку Гаразд.



НАСТАНОВА КОРИСТУВАЧА



Повернути зображення. Ви можете задати значення кута повороту. Для цього утримуючи клавішу SHIFT виберіть потрібну кнопку. У вікні, що відкрилося, задайте потрібне значення та виберіть кнопку Гаразд



Дзеркальне відображення (горизонтальне та вертикальне дзеркало)

Збільшення або зменшення масштабу перегляду малюнка. Якщо малюнок повністю не поміщається у вікні, то з'являться повзунці, за допомогою яких можна переміщувати видиме зображення.



Задати для виконавця (образу) точку, яка буде центром обертання при виконанні відповідних команд.

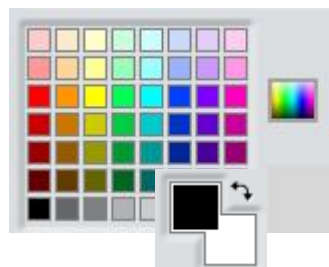


Закінчити малювання та перенести виконавця до Списку виконавців

Колір для малювання можна обрати за допомогою інструмента

Піпетка на одній з палітр кольорів.

Обраний колір малювання відображається на покажчику кольору (верхній квадрат — колір малювання, нижній — колір фону, стрілка дозволяє міняти кольори місцями). Щоб обрати колір фону треба при використанні Піпетки скористатись правою клавішею мишки.



НАСТАНОВА КОРИСТУВАЧА



Величини

Типи даних

В Скретч розрізняють наступні типи даних – символічні, графічні та аудіодані.

До символічних даних відносяться числа та тексти (рядки символів).

Числа можуть бути цілими та дійсними. Дійсні числа можуть записуватися у вигляді десяткового дробу (з крапкою, яка розділяє цілу та дробову частину числа) та у експоненціальному вигляді, наприклад, $1.56e4$.

Рядки символів (тексти) - це довільні послідовності літер, слів чи інших символів, наприклад, *яблуко, жовтень 2010 року, Ви виграли.*

До графічних даних відносяться:

- виконавці та фони сцени;
- малюнки, створені з використанням групи блоків **Олівець**.

До аудіо даних відносяться звуки зі стандартної галереї, готові аудіо файли та записані користувачем з використанням вбудованого програвача.

Організація даних

В мовах програмування використовують наступну організацію даних:

- скалярні дані: константи і змінні.
- структуровані дані: таблиці, масиви, списки і т.п. У Скретч використовуються тільки списки.

Змінні

Змінна (англ. *variable*) – це поіменована область в пам'яті комп'ютера.

Змінні використовуються для :

- збереження введених даних;
- збереження результатів обчислень.

Типізація існуючих змінних відбувається динамічно, за їх значенням.

НАСТАНОВА КОРИСТУВАЧА



Створення змінних

Змінні створюються з використанням команди **Створити змінну** в групі блоків **Змінні**.

При створенні змінної задається ім'я (використовується в командах для посилання на її поточне значення) та область дії змінної (всі виконавці проекту або один виконавець).

В залежності від області дії розрізняють локальні та глобальні змінні.

Локальна змінна – змінна, область дії якої поширюється лише на одного виконавця.

Глобальна змінна – змінна, область дії якої поширюється на всіх виконавців.

Імена локальних змінних можуть бути однаковими для різних виконавців, але не можуть співпадати з іменем глобальної змінної.

При наданні імен змінним слід дотримуватися певних правил:

- Ім'я змінної розпочинається з літери
- Ім'я змінної може містити літери, цифри та знак “_”.

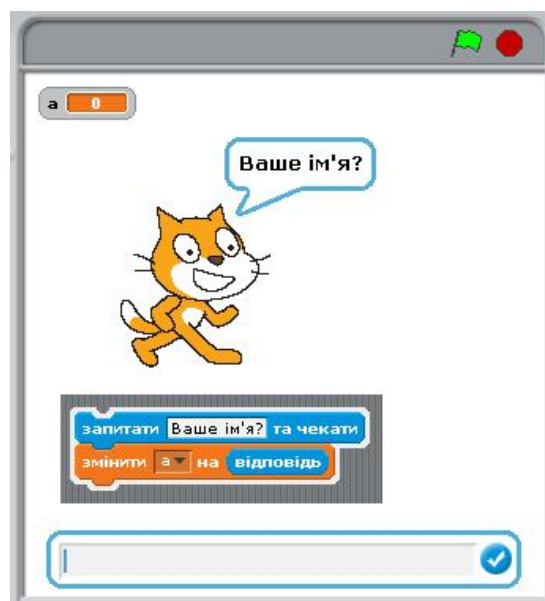
Надання змінній значення

Операція надання значень змінній називається присвоюванням. Початкове значення змінної рівне нулю. В кожний момент часу змінна може мати тільки одне значення. При наданні змінній нового значення її попереднє значення втрачається.

Змінні можуть набувати значень різних типів — це може бути число, текст або логічне значення.

Надати значення змінній можна одним з трьох способів:

- надання змінній фіксованого значення;
- надання змінній значення, що є результатом обчислень;
- використання команди **Запитати та чекати** для отримання в діалоговому режимі даних від користувача, після чого ці дані можна використовувати як значення змінної.



НАСТАНОВА КОРИСТУВАЧА



Значення змінної можна змінювати як під час створення проекту, так і під час його виконання. Для цього використовуються блоки

надати **a** значення **1**

та змінити **a** на **1**, ім'я змінної вибирається зі списку, що відкривається, значення - константа, інша змінна, арифметичний вираз.

Поточне значення змінної можна відобразити на Сцені, якщо поставити позначку прапорця біля імені змінної в області

Контейнер блоків. Значення змінної відображається у віконці – моніторі. Розрізняють наступні форми монітору - нормальний, великий, нормальний з повзунком. Для зміни форми віконця - монітору слід скористатися контекстним меню.



нормальний, відображається назва та відповідне значення



великий, відображається лише відповідне значення



нормальний з повзунком, відображається назва, відповідне значення. Повзунець дозволяє змінювати значення змінної перетягуванням мишкою при виконанні проекту.

Списки

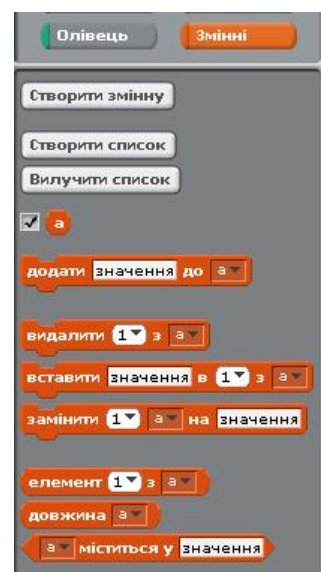
Список (англ. *list*) - це впорядкований набір значень (*елементів*) деякої величини.

Список позначається одним ім'ям, яке можна використовувати для посилання на нього. Кожне значення списку має свій порядковий номер (*індекс*), який використовується для посилання на цей елемент списку. Подібні набори в інших мовах програмування називаються *одновимірними масивами*.

Елементами списку можуть бути числа та тексти (рядки).

Створення списків відбувається за допомогою команди **Створити список** групи блоків **Змінні**.

Після створення списку з'являються блоки (команди), які відповідають за дії зі списками та їх елементами

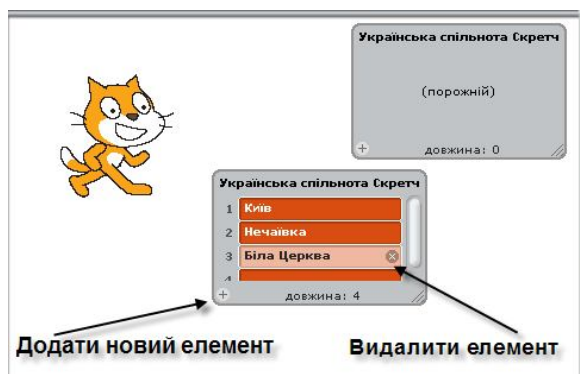


НАСТАНОВА КОРИСТУВАЧА



Елементи списку можна відобразити на Сцені, якщо поставити позначку прапорця біля імені списку в області *Контейнер блоків*. Елементи відображаються у віконці – моніторі. Розмір монітора можна змінювати, розтягуючи його за нижній правий кут.

Спочатку список порожній і його *довжина* (кількість елементів) рівна 0. Елементи списку можна додавати або змінювати, вводючи їх значення прямо у віконці-моніторі. Клацання по знаку "+" додає до списку новий елемент. Для додавання можна також використовувати блоки



та

Елементи списку можна імпортувати з текстового файлу формату .txt (елемент повинен розпочинатися з нового рядка, при збереженні файлу слід вибрати кодування UTF-8) та експортувати в текстовий файл.

Вирази

За допомогою виразу можна задати правило для знаходження значення деякої величини.

Вираз складається з операндів та знаків операцій. Операнди можуть бути константами, змінними і функціями. Операції поділяються на:

- арифметичні: +, -, *, / ;
- порівняння: <, =, > ;
- логічні: і, або, не.
- текстові: порівняння, об'єднання
- обробки подій

Для створення виразів використовуються блоки, що входять до складу групи **Оператори**.

Арифметичні вирази описують правила обчислення значення деякої величини. Операнди в арифметичних виразах з'єднуються символами арифметичних операцій.



додавання

віднімання

множення

НАСТАНОВА КОРИСТУВАЧА



ділення

Арифметичні вирази можуть містити математичні функції, наприклад, з . Деякі функції представлені окремими блоками, наприклад, та .

Для задання порядку виконання операцій у виразі в Скретч не можна використовувати дужки.

Логічні вирази призначені для записування різноманітних умов. Логічні вирази можуть набувати одного з двох можливих значень «істинно» (true) або «хибно» (false).

У Скретч надавати змінним логічного значення в явному вигляді неможна. Однак існує ряд операцій та виразів, результатом яких є логічні значення. Логічні значення є результатом арифметичних та рядкових операцій порівняння, деяких операцій опрацювання подій (наприклад, ,).

Прості логічні вирази – це два арифметичні вирази, з'єднанні символами відношення (порівняння).

Менше

Дорівнює

Більше

Складенні логічні вирази – це прості логічні вирази, до яких застосовуються логічні операції і (and), або (or), не (not).

Логічна операція **і (and)**. Повертає значення «істинно», коли обидва вирази істинні

Логічна операція **або (or)**. Повертає значення «істинно», коли хоча б один з виразів істинний.

Логічна операція **не (not)**. Повертає значення «істинно», якщо вираз хибний, та значення «хибно», якщо вираз істинний

Над тестовими даними визначені операції порівняння та наступні функції:

З'єднати Привіт світ

З'єднує рядки в один рядок

символ 1 у світ

Визначає символ, який знаходиться за вказаним номером

довжина світ

Визначає кількість символів в заданому рядку

НАСТАНОВА КОРИСТУВАЧА



Блоки

Типи блоків

Блоки розрізняються за призначенням, формою та способом заповнення. Типи блоків за призначенням відображено на рис.






Блоки стека. Блоки мають заглиблення зверху і виступ знизу, у такий спосіб блоки з'єднуються та утворюють стек. Деякі блоки мають поля введення, наприклад, **переміститись на 10 кроків**. Такі поля призначені для введення значень з клавіатури. В ці ж поля перетягуванням мишкою можна помістити блоки посилань. В деяких блоках значення вибирається зі списку, що відкривається, наприклад **грати звук: няв**. Значення в деяких блоках можна як вводити з клавіатури, так і вибирати зі списку, наприклад, **грати ноту: 60 0.5 тактів**.




Деякі блоки, наприклад **завжди**, мають C-подібну форму (**C-shaped «mouth»**), що дозволяє всередині конструкції розміщувати групу блоків.

НАСТАНОВА КОРИСТУВАЧА



Блоки заголовків. Блоки мають опуклий верхній край і виступ для з'єднання з відповідними блоками стеку, наприклад . При настанні відповідної події запускається на виконання стек, блоки якого розташовано під заголовком.

Блоки посилань. Блоки мають рівний край та розміщуються в полях введення блоків стеку, наприклад,  або .

Блоки з округлими кінцями (наприклад,  або ) поміщуються в блоки стеку, які мають округлі або прямокутні поля введення (наприклад,  або ). Блоки з загостреними кінцями (наприклад, ) поміщуються в блоки стеку, які мають загострені або прямокутні поля введення (наприклад,  або ). Перед деякими блоками посилань розміщується перемикач-прапорець, наприклад,  . Призначення таких блоків – відобразити/сховати на сцені поточне значення.

Блоки реалізації базових структур алгоритмів

Базові структури алгоритму — це структури, за допомогою яких можна скласти алгоритм розв'язання будь-якої задачі, якщо він існує.

Розрізняють наступні базові структури алгоритмів - *слідування* (лінійний алгоритм), *розгалуження* (прийняття рішень) та *цикли* (повторення).

Розгалуження. Характерною рисою таких алгоритмів є те, що деякі їх команди виконуються і при чому тільки один раз, а деякі команди зовсім не виконуються. Це залежить від результату перевірки умов – істина чи хибність. Блоки, які реалізують алгоритм з розгалуженням:

Повне розгалуження



Перевіряється умова (обчислюється логічний вираз). Якщо умова істинна, то виконуються блоки першої гілки (верхня частина конструкції); інакше виконуються блоки другої гілки (нижня частина конструкції)

НАСТАНОВА КОРИСТУВАЧА



Неповне розгалуження



Перевіряється умова (обчислюється логічний вираз). Якщо логічний вираз істинний, то виконуються команди, які розташовані всередині блоку, інакше виконуються блоки стеку, що розташовані нижче.

Цикли. Це алгоритм, в якому послідовність команд може виконуватися більше одного разу, тобто передбачено повторення деякої послідовності команд.

Нескінченні цикли



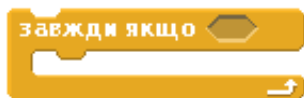
Постійно виконувати групу команд, які знаходяться всередині блоку

Цикли з параметром



Виконувати групу команд, які знаходяться всередині блоку, вказану кількість разів

Цикли з передумовою



Повторювати виконання групи команд, які знаходяться всередині блоку, до тих пір, поки виконується умова (значення логічного виразу істинне)

Перевіряється умова (обчислюється логічний вираз). Якщо умова виконується, то виконуються команди, які знаходяться всередині блоку. Знову перевіряється умова. Якщо значення виразу істинне, то команди виконуються повторно, якщо ж логічний вираз хибний, зазначені команди більше не виконуються, цикл завершується



Повторювати виконання групи команд, які знаходяться всередині блоку, до тих пір, поки умова не виконується (значення логічного виразу хибне)

Перевіряється умова (обчислюється логічний вираз). Якщо значення логічного виразу хибне, то виконуються команди, які знаходяться всередині блоку. Знову перевіряється умова. Якщо логічний вираз хибний, то команди виконуються повторно, якщо ж логічний вираз істинний, зазначені команди більше не виконуються, цикл завершується.

НАСТАНОВА КОРИСТУВАЧА



Події



При виконанні проекту виконавці можуть реагувати на різні події - клацання кнопки мишки, натиснення клавіші на клавіатурі, зіткнення з іншим виконавцем, надходження повідомлення від іншого виконавця тощо.

Розрізняють наступні види подій:


- події, які запускають скрипти;
- події, які можуть запускати певні дії у скриптах;
- посилання на величини, певне значення яких може бути визначено, як подія.

Після того, як ви визначили на які події повинні реагувати виконавці, слід продумати послідовність дій, які необхідно виконувати у випадку виникнення тої чи іншої події. Найчастіше це реалізується шляхом запуску певних скриптів за умови появи події з використанням блоків-заголовків. Інший спосіб полягає у перевірці наявності події або стану величини (у тому числі і параметрів виконавців).

Події, які запускають скрипти





	Запустити на виконання відповідні стеки, коли вибрано кнопку
	Запустити на виконання стек, коли вибрано виконавця
	Запустити на виконання стек, коли натиснуто клавішу на клавіатурі
	Запустити на виконання стек, коли вибрано сцену
	При отриманні повідомлення, запустити на виконання скрип. Повідомлення вибирається зі списку, що відкривається

Події, які можуть запускати певні дії у скриптах


	Чи доторкається виконавець вказівника мишки, краю Сцени, іншого виконавця? Повертає значення «істинно» або «хибно».
---	---

НАСТАНОВА КОРИСТУВАЧА



доторкається кольору  ?	Чи доторкається виконавець до вказаного кольору? Повертає значення «істинно» або «хибно». Для вибору кольору клацніть на палітрі кольорів, потім використовуючи піпетку, виберіть колір
колір  торкається  ?	Чи доторкається колір 1 до кольору 2? Повертає значення «істинно» або «хибно».
клавішу  натиснуто?	Чи натиснуто клавішу на клавіатурі? Повертає значення «істинно» або «хибно».
мишку натиснуто?	Чи натиснута кнопка мишки? Повертає значення «істинно» або «хибно».

Посилання на величини, значення яких може бути визначено, як подія

мишка x	Визначити (повернути) значення вказівника мишки по осі x
мишка y	Визначити (повернути) значення розташування вказівника мишки по осі y
відстань до 	Визначити (повернути) відстань між виконавцем та іншим об'єктом, наприклад, інший виконавець, вказівник мишки, край Сцени.
таймер	Визначити (повернути) значення таймера в секундах
гучність	Визначити (повернути) значення гучності відтворення звуків (1 ÷ 100)
значення y	Визначити (повернути) ординату розташування виконавця (-240 ÷ 240)
значення x	Визначити (повернути) абсцису координати розташування виконавця (-180 ÷ 180)
розмір	Визначити (повернути) розмір виконавця (у відсотках від початкового розміру)
образ №	Визначити (повернути) поточний номер образу виконавця
темп	Визначити (повернути) темп звучання (в ударах за хвилину)
гучність	Визначити (повернути) гучність звуку

НАСТАНОВА КОРИСТУВАЧА



Повний перелік команд

Рух	
	Перемістити виконавця на вказану кількість кроків (одиниць).
	Повернути за годинниковою стрілкою
	Повернути проти годинникової стрілки
	Повернути у вказаному напрямку. (0 - вгору, 90 - праворуч, 180 - вниз, -90 - ліворуч)
	Повернути у напрямі вказівника мишки або іншого виконавця
	Перемістити виконавця в точку з координатами (x,y)
	Перемістити виконавця в точку розташування вказівника мишки або іншого виконавця
	Перемістити (покроково) виконавця в точку з координатами (x,y) за вказану кількість секунд
	Змінити значення ординати на вказане число
	Задати значення ординати розташування виконавця
	Змінити значення абсциси на вказане число
	Задати значення абсциси розташування виконавця
	Коли виконавець доходить до краю Сцени, то повернути його назад
	Визначити (повернути) ординату розташування виконавця (-240 ÷ 240)
	Визначити (повернути) абсцису координати розташування виконавця (-180 ÷ 180)
	Визначити (повернути) напрямок руху виконавця (0 - вгору, 90 - праворуч, 180 - вниз, -90 - ліворуч)

НАСТАНОВА КОРИСТУВАЧА






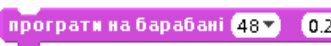









Вигляд

змінити образ на	Змінити образ виконавця на вибраний зі списку, що відкривається
наступний образ	Послідовно змінювати образи зі списку образів (якщо досягнуто кінець списку, то переходить до першого образу.)
образ №	Визначити (повернути) номер поточного образу виконавця
змінити фон на	Змінити фон сцени на вибраний зі списку, що відкривається
наступний фон	Послідовно змінювати фон зі списку (якщо досягнуто кінець списку, то переходить до першого фону)
фон #	Визначити (повернути) номер поточного фону сцени
говорити впродовж	Показати репліку задану кількість секунд. ("розмовна бульбашка")
говорити 	Показати репліку ("розмовна бульбашка")
подумати	Показати репліку задану кількість секунд ("бульбашка думки")
подумати 	Показати репліку ("бульбашка думка")
змінити ефект на	Змінити візуальний ефект на вказане число (ефект вибирається зі списку, що відкривається)
встановити ефект в	Задати значення для візуального ефекту (допустимі значення від 0 до 100.)
очистити графічні ефекти	Відмінити всі попередні команди щодо зміни зовнішнього вигляду об'єкта
змінити розмір на	Пропорційно збільшити/зменшити розмір виконавця
задати розмір	Задати розмір виконавця у відсотках від початкового розміру
розмір	Визначити (повернути) розмір виконавця (у відсотках від початкового розміру)
показати	Зробити виконавця видимим
сховати	Зробити виконавця невидимим
стати попереду всіх	Розмістити виконавця попереду всіх інших виконавців
стати позаду	Розмістити виконавця позаду вказаної кількості виконавців





НАСТАНОВА КОРИСТУВАЧА



Звук







	Розпочати програвати звук зі списку, що відкривається, та перейти до виконання наступного блоку (навіть якщо звук все ще грає)
	Програти звук зі списку, що відкривається, дочекатися його завершення та перейти до виконання наступного блоку
	Зупинити відтворення всіх звуків
	Програти на барабані (тип вибирається зі списку, що відкривається) вказану кількість тактів
	Програти музичну ноту вказане число тактів
	Зробити паузи на вказане число тактів
	Вибрати інструмент зі списку, що відкривається
	Змінити гучність відтворення звуків на вказане число
	Задати гучність відтворення звуків
	Визначити (повернути) гучність звуку
	Змінити темп відтворення музики на вказане число
	Задати темп відтворення музики в ударах за хвилину
	Визначити (повернути) темп звучання в ударах за хвилину

Олівець










	Очистити Сцену всі сліди олівця та штампи зі Сцени
	Увімкнути режим малювання - дозволити виконавцю малювати під час його переміщення
	Вимкнути режим малювання для малюванні
	Вибрати колір олівця з кольорової палітри

НАСТАНОВА КОРИСТУВАЧА



	Змінити колір олівця на вказане число (0 -червоний, 100 - синій)
	Змінити тінь олівця на вказане число (0 - дуже темний, 100 - дуже світлий)
	Задати розмір тіні олівця
	Змінити товщину олівця
	Задати товщину олівця
	Дублювати виконавця на Сцені

Контроль

	Запустити на виконання відповідні стеки, коли вибрано кнопку
	Запустити на виконання стек, коли натиснуто клавішу на клавіатурі
	Запустити на виконання стек, коли вибрано виконавця
	Запустити на виконання стек, коли вибрано сцену
	Затримати виконання наступного блоку на вказану кількість секунд
	Постійно виконувати групу команд, які знаходяться всередині блоку
	Виконувати групу команд, які знаходяться всередині блоку, вказану кількість разів
	Відправити повідомлення іншим виконавцям і продовжувати виконувати власний скрипт
	Відправити повідомлення іншим виконавцям і чекати закінчення виконання їх скриптів














НАСТАНОВА КОРИСТУВАЧА



	При отриманні повідомлення, запустити на виконання скрип. Повідомлення вибирається зі списку, що відкривається
	Повторювати виконання групи команд, які знаходяться всередині блоку, до тих пір, поки виконується умова (значення логічного виразу істинне). Перевіряється умова (обчислюється логічний вираз). Якщо умова виконується, то виконуються команди, які знаходяться всередині блоку. Знову перевіряється умова. Якщо значення виразу істинне, то команди виконуються повторно, якщо ж логічний вираз хибний, зазначені команди більше не виконуються, цикл завершується
	Перевіряється умова (обчислюється логічний вираз). Якщо умова істинна, то виконуються команди, які знаходяться всередині блоку
	Перевіряється умова (обчислюється логічний вираз). Якщо умова істинна, то виконуються блоки першої гілки (верхня частина конструкції); інакше виконуються блоки другої гілки (нижня частина конструкції)
	Не повторювати виконання групи блоків, до тих пір, поки умова не виконується (значення логічного виразу хибне). Перевіряється умова (обчислюється логічний вираз). Якщо значення логічного виразу хибне, то не виконувати блоки, які розташовані нижче. Знову перевіряється умова. Якщо логічний вираз хибний, то команди не виконуються повторно, якщо логічний вираз істинний, зазначені команди виконуються, цикл завершується
	Повторювати виконання групи команд, які знаходяться всередині блоку, до тих пір, поки умова не виконується (значення логічного виразу хибне) Перевіряється умова (обчислюється логічний вираз). Якщо значення логічного виразу хибне, то виконуються команди, які знаходяться всередині блоку. Знову перевіряється умова. Якщо логічний вираз хибний, то команди виконуються повторно, якщо ж логічний вираз істинний, зазначені команди більше не виконуються, цикл завершується.
	Зупинити виконання скрипта
	Зупинити виконання всіх скриптів для всіх виконавців

НАСТАНОВА КОРИСТУВАЧА



Датчики	
	Визначити (повернути) значення розташування вказівника мишки по осі x
	Визначити (повернути) значення розташування вказівника мишки по осі y
	Чи натиснута кнопка мишки? Повертає значення «істинно», якщо кнопку мишки натиснуто або «хибно», якщо кнопку мишки не натиснуто
	Чи натиснуто клавішу на клавіатурі? Повертає значення «істинно», якщо вказану клавішу натиснуто або «хибно», якщо клавішу не натиснуто
	Чи доторкається виконавець вказівника миші, краю Сцени, іншого виконавця? Повертає значення «істинно», якщо виконавець торкається (значення вибирається зі списку, що відкривається) або «хибно», якщо виконавець не торкається
	Чи доторкається виконавець вказаного кольору? Повертає значення «істинно», якщо виконавець торкається вказаного кольору або «хибно», якщо виконавець не торкається. Для вибору кольору клацніть по кольоровій палітрі, потім використовуючи піпетку, виберіть колір
	Чи доторкається колір 1 кольору 2? Повертає значення «істинно», якщо колір (в межах виконавця) торкається іншого кольору (фону, іншого виконавця) або «хибно», якщо колір не торкається
	Визначити (повернути) відстань між виконавцем та іншим об'єктом (виконавець, вказівник мишки, край Сцени)
	Почати відлік часу з нуля
	Визначити (повернути) значення таймера в секундах
	Повертає значення деяких властивостей (координати точки, розмір, гучність, напрямок, номер образу) виконавців. Властивості вибираються зі списку, що відкривається.
	Повертає значення гучності відтворення звуків (1 ÷ 100)
	Істинно, якщо комп'ютерний мікрофон виявляє гучність звуку, більше, ніж 30 (1 ÷ 100)

НАСТАНОВА КОРИСТУВАЧА



Оператори

	Знайти суму двох величин (константи, змінні)
	Знайти різницю двох величин (константи, змінні)
	Знайти добуток двох величин (константи, змінні)
	Знайти частку двох величин (константи, змінні)
	Згенерувати випадкове ціле число в межах вказаного інтервалу
	Порівняти величини. Істинно, якщо перше значення є менше ніж друге
	Порівняти величини. Істинно, якщо два значення рівні
	Порівняти величини. Істинно, якщо перше значення більше, ніж друге
	Логічна операція і (and) . Повертає значення «істинно», коли обидві умови справджуються
	Логічна операція або (or) . Повертає значення «істинно», коли хоча б одна з умов справджується.
	Логічна операція не (not) . Повертає значення «істинно», якщо умова не справджується, та значення «хибно», якщо умова справджується
	Обчислити значення вибраної функції
	Знайти залишок від ділення першого значення на друге
	Знайти найближче ціле число до вказаного значення

НАСТАНОВА КОРИСТУВАЧА



Змінні

	Створити нову змінну. При створенні змінної необхідно задати її ім'я та вказати область її дії (локальна, глобальна)
	Видалити змінну та всі блоки, пов'язані з нею
	Визначити (повернути) значення змінної
	Надати змінній нове значення. Змінна вибирається зі списку, що відкривається.
	Змінити значення змінної на нове. Змінна вибирається зі списку, що відкривається.
	Відобразити на сцені віконце-монітор змінної
	Не відображати на сцені віконце-монітор змінної
	Створити список. При створенні списку необхідно задати ім'я та вказати область дії (локальний, глобальний)
	Видалити список та блоки, пов'язані з ним
	Визначити (повернути) всі елементи списку
	Додати новий елемент в кінець списку. Елемент може бути числом або рядком символів
	Видалити елемент списку. Номер елемента вибирається зі списку, що відкривається, ім'я списку, також вибирається. Видалення елемента зменшує довжину списку на 1.
	Додати новий елемент у вказане місце списку. Номер елемента вибирається зі списку, що відкривається, ім'я списку, також вибирається. Довжина списку збільшується на 1
	Замінити значення елемента списку на нове. Довжина списку не змінюється
	Визначити (повернути) значення елемента списку із вказаними номером
	Визначити (повернути) кількість елементів списку