

# Podstawy Scratcha

**Autor:** Marcin Piotrowicz

## Lekcja 2: Pierwszy skrypt w Scratch

Drugie zajęcia z programowania w Scratch to kontynuacja pracy z animacją stworzoną podczas pierwszych zajęć. Uczniowie zaczynają korzystać z coraz większej ilości skryptów, pracują nad animacją tła. Wprowadzimy też pierwsze instrukcje warunkowe, które pozwolą nam tworzyć bardziej zaawansowane bloki skryptów.

### Cele zajęć:

Uczeń powinien:

- Znać różne sposoby uruchamiania skryptów
- Posługiwać się blokami z bibliotek ruch, wygląd, kontrola, zdarzenia
- Dokonywać zmiany tła, wprowadzać efekt animacji tła
- Posługiwać się instrukcją warunkową

### Pojęcia kluczowe:

- ⇒ Scena, duszek
- ⇒ Skrypt
- ⇒ Kontrola, zdarzenia, czujniki
- ⇒ Instrukcja warunkowa
- ⇒ oś „x” i „y”

### Metody pracy:

- Wykład, dyskusja, prowadzenie
- Ćwiczenia praktyczne przy komputerze
- Prezentowanie efektów pracy
- Burza mózgów
- „Uczenie się przez osobiste doświadczenie — ograniczamy liczbę podawanych informacji do niezbędnego minimum i dążymy do tego, aby uczeń mógł rozpocząć samodzielną pracę” [A.Wałat, *Wybrane problemy dydaktyki informatyki*]

### Materiały pomocnicze:

- Scratch - [www.scratch.mit.edu](http://www.scratch.mit.edu)
- Wyposażenie pracowni komputerowej (2 uczniów na 1 stanowisko komputerowe)

**Czas na realizację zajęć:** 45 min



### Treści programowe (związek z podstawą programową)

Podstawa programowa kształcenia ogólnego dla szkół podstawowych — II etap edukacyjny — klasy IV-VI. Treści szczegółowe zajęć komputerowych:

3.1 Wyszukiwanie i wykorzystywanie informacji z różnych źródeł. Uczeń: wyszukuje informacje w różnych źródłach elektronicznych (słowniki, encyklopedie, zbiory biblioteczne, dokumentacje techniczne i zasoby Internetu);

4.1 Opracowywanie za pomocą komputera rysunków, motywów, tekstów, animacji, prezentacji multimedialnych i danych liczbowych. Uczeń: tworzy rysunki i motywy przy użyciu edytora grafiki (posługuje się kształtami, barwami, przekształcaniem obrazu, fragmentami innych obrazów);

5.1 Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera. Uczeń: za pomocą ciągu poleceń tworzy proste motywy lub steruje obiektem na ekranie;

5.2 Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera. Uczeń: uczestniczy w pracy zespołowej, porozumiewa się z innymi osobami podczas realizacji wspólnego projektu, podejmuje decyzje w zakresie swoich zadań i uprawnień

7.2 Wykorzystywanie komputera i technologii informacyjno-komunikacyjnych do rozwijania swoich zainteresowań, zastosowanie komputera w życiu codziennym, opisywanie zagrożeń i ograniczeń związanych z korzystaniem z komputera i Internetu. Uczeń: szanuje prywatność i pracę innych osób;

7.3 Wykorzystywanie komputera i technologii informacyjno-komunikacyjnych do rozwijania swoich zainteresowań, zastosowanie komputera w życiu codziennym, opisywanie zagrożeń i ograniczeń związanych z korzystaniem z komputera i Internetu. Uczeń: przestrzega zasad etycznych i prawnych związanych z korzystaniem z komputera i Internetu, ocenia możliwe zagrożenia.

## Przebieg zajęć:

### 1. Wprowadzenie w tematykę i integracja grupy - 12 min.

Zajęcia zaczynamy od przypomnienia efektów naszej pracy z pierwszych zajęć.

Uczniowie logują się na swoje konta. Tych, którzy w domu pracowali nad swoim projektem, zachęcamy do zaprezentowania swojego mini-programu. Również ci, którzy zremiksowali cudzy projekt i dokonali w nim zmian, mogą pochwalić się swoim osiągnięciem. Poproście uczniów, aby zaprezentowali, w jaki sposób zaprogramowali poszczególne funkcjonalności programu.

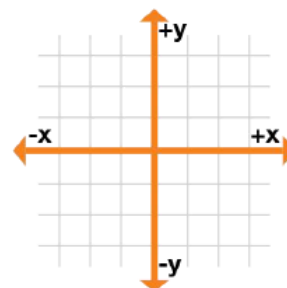
Zapewne dla części będzie to czas na przypomnienie sobie sposobu logowania się na stronie [scratch.mit.edu](https://scratch.mit.edu) lub dokończenia pracy z poprzednich zajęć. Pozwalamy uczniom na indywidualizację pracy.

### 2. Część zasadnicza - 25 min.

#### Sterowanie duszkiem

Na tym etapie pracy wracamy do naszej animacji z pierwszych zajęć.

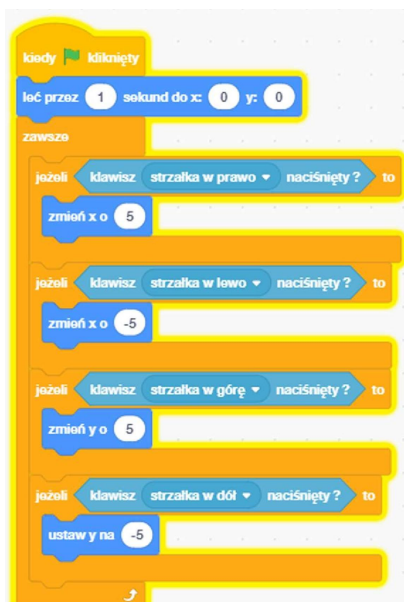
Zapytajmy uczniów czy znają grę Minecraft (tam występują koordynaty x,y,z), czy mają doświadczenie z grą w szachy/warcaby oraz dyktandami graficznymi. Niech wpadną na pomysł co łączy te wszystkie aktywności, skąd wiemy, gdzie położony jest pionek w warcabach lub określony kolor w dyktandzie graficznym. Aby móc lepiej sterować naszym duszkiem w Scratchu, wprowadzamy współrzędne kartezjańskie i tłumaczymy uczniom, że każdy punkt na scenie ma swój „adres” (podobnie jak każde pole na szachownicy lub dyktandzie graficznym). Można wyświetlić lub narysować kratownicę zawierającą oś x i y na tablicy i pozwolić uczniom odczytać położenie kilku obiektów.



Modyfikujemy skrypt z poprzednich zajęć tak, by możliwe było sterowanie duszkiem za pomocą strzałek klawiatury. Pozwalamy uczniom na burzę mózgow i zaproponowanie, jak mógłby wyglądać taki skrypt. Warto również dodać bloczek, który sprawi, że zawsze na początku gry duszek będzie startował ze środka ekranu. W zaproponowanym na pierwszych zajęciach skrypcie był to nurek. Zastanawiamy się, co powinno się stać, gdy nurek dotknie do krawędzi ekranu?

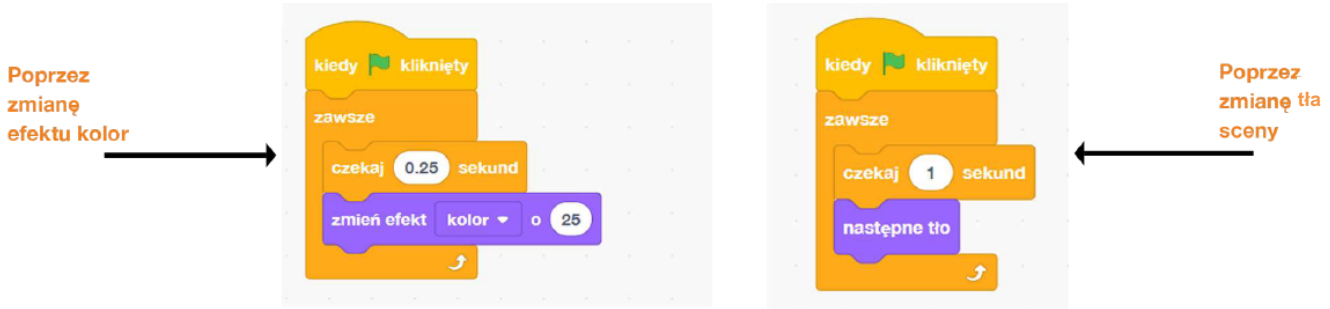
Tłumaczymy uczniom czym są bloczki instrukcji warunkowej (w skryptach „kontrola”).

Przykładowe rozwiązanie zadania może wyglądać następująco:

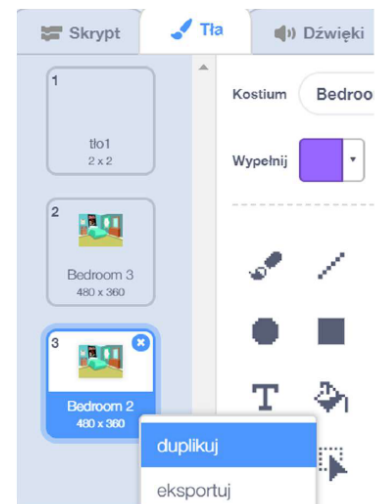


### „Animacja” tła

Warto podsunąć uczniom wątpliwość, czy stałe tło, które się nie zmienia ich zadowala? Czy można je jakoś animować? Uczniowie mogą wskazać dwa rozwiązania:



W tym przypadku warto pokazać sposób modyfikowania tła: klikamy na scenę, wchodzimy w zakładkę „tła”, następnie duplikujemy scenę kilka razy i w każdym duplikacie dokonujemy jakichś zmian (np. dorysowując jakieś elementy).



### 3. Podsumowanie i ewaluacja - 8 min.

Nauczyciel sprawdza postępy pracy uczniów. Zachęca do dokończenia zadań w domu. Prosi, by uczniowie przed zakończeniem pracy wpisali do opisu projektu krótką instrukcję obsługi gry.

Nauczyciel zadaje także pytania ewaluacyjne:

- kto nauczył się czegoś nowego w Scratch;
- kto samodzielnie odkrył coś nowego;
- co było ciekawe podczas dzisiejszej pracy;
- co można by było zmienić na następnych zajęciach.

### Uwagi/alternatywy:

Zapewne niektórzy uczniowie spotkali się z programowaniem w Scratchu. Jeżeli powyższe zadania wykonali szybko, mogą samodzielnie wprowadzić nowe elementy. W naszej grze może to być np. jakaś przeszkoda, którą należy omijać, a której prosty skrypt może wyglądać następująco:

