

Poziom

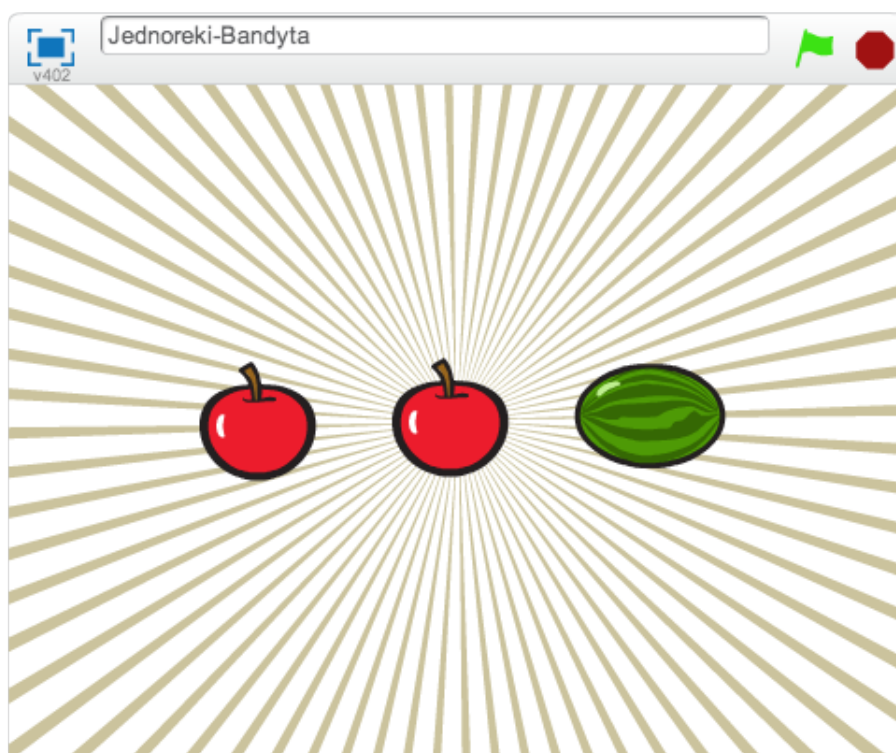
2

Jednoreki bandyta

{code
club}

Wstęp

Zrobimy prostą grę z trzema duszkami, które się zmieniają w trakcie gry. Twoje zadanie polega na zatrzymaniu tych duszków w taki sposób, aby wszystkie miały taki sam obrazek (jak w jednorekim bandycie!).



Zadania do wykonania

Wykonaj te **POLECENIA** krok po kroku



Przetestuj swój projekt

Kliknij na zieloną flagę, aby
PRZETESTOWAĆ swój kod



Zapisz swój projekt

Teraz **ZAPISZ** swój projekt

1

Krok 1: Tworzenie duszka, który zmienia kostiumy

✓ Zadania do wykonania

Zaimportujmy obrazki potrzebne do gry

1. Rozpocznij nowy projekt w Scratchu. Usuń kota, klikając na nim prawym przyciskiem myszki i wybierając 'usuń'. ☐
2. Dodaj nowe tło z biblioteki. Z kategorii Inne wybierz tło rays a następnie usuń puste tło sceny. ☐
3. Zaimportuj nowego duszka. ☐
4. Wybierz obrazek z dowolnego katalogu. My użyliśmy Przedmioty/bananas1, ale możesz wybrać dowolny obrazek, który ci się podoba. ☐
5. Kliknij na zakładkę 'Kostiumy' i wybierz dwa inne obrazki – potrzebujemy w sumie trzy różne kostiumy (my wybraliśmy Przedmioty/apple i Przedmioty/watermelon-a, ale możesz wybrać którekolwiek chcesz). ☐

Skoro mamy już kostiumy, teraz sprawmy, aby duszek nosił je wszystkie na zmianę.

Krok 2: Zmieniamy pojawiające się obrazki

✓ Zadania do wykonania

1. Przejdź do zakładki `Skrypty`. ☐
2. Kliknij na palecie `Zdarzenia` i przeciągnij blok `kiedy kliknięto zieloną flagę` na obszar ze skryptami. ☐
3. Dodaj blok `zawsze` z palety `Kontrola` i dołącz go do bloku z flagą. ☐
4. Kliknij na zieloną flagę w prawym rogu nad sceną. ☐
Zauważ, że nasz skrypt podświetlił się na żółto. Dzieje się

tak ponieważ kliknęliśmy na zieloną flagę, co spowodowało, że skrypt się uruchomił.

5. Kliknij na palecie **Wygląd** i dołącz do skryptu blok **następny kostium**. Kliknij zieloną flagę. ☐
6. Co możemy zrobić, aby obrazki nie zmieniały się tak szybko? Dodaj blok **czekaj 1 s** z palety **Kontrola**. ☐
7. Zmień czas, aby obrazki zmieniały się troszkę szybciej, ale nie za szybko - 0.5 s powinno być w sam raz. Co się stanie, jeżeli usuniemy ze skryptu blok **czekaj 1 s**? ☐



Przetestuj swój projekt

Kliknij zieloną flagę.

Czy kostiumy zmieniają się z rozsądną szybkością?



Zapisz swój projekt.

Rzeczy do spróbowania

Zmień ilość sekund w bloku **czekaj 1 s**. Jak myślisz, przy jakiej wartości gra stanie się zbyt prosta, a przy jakiej wartości będzie za trudna?

Krok 3: Zatrzymujemy obrazki kliknięciem klawisza myszki

Świetnie! Potrafimy już sprawić, aby kostiumy duszka zmieniały się w nieskończoność, ale jak zrobić, aby zatrzymały się, kiedy na niego klikniemy?

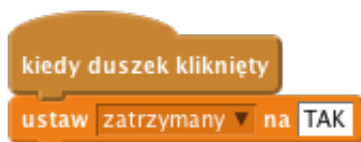
Można do tego użyć zmiennej, w której będziemy przechowywać stan Duszka. Przyda się ona także później...

✓ Zadania do wykonania

1. Kliknij na palecie `Zmienne` i `stwórz nową zmienną`. Nazwij ją `zatrzymany` i zaznacz, że ma być dostępna tylko dla tego duszka. Następnie odznacz haczyk obok niej, aby nie była widoczna na Scenie. ☐
2. Na początku gry duszek nie został kliknięty, więc ustawimy wartość zmiennej na "NIE". ☐



3. Teraz będziemy ustawiać wartość zmiennej `zatrzymany` na "TAK" kiedy ktoś kliknie na duszka. ☐



4. Na koniec chcemy, aby duszek przestał zmieniać kostiumy kiedy zmienna `zatrzymany` zmieni się na "TAK". Dodaj blok `jeżeli ... to` i użyj bloku do porównania `[] = []` (znajdziesz go w panelu *Wyrażenia*), aby sprawdzić czy ☐

zmienna **zatrzymany** ma nadal wartość "NIE".



🚩 Przetestuj swój projekt

Kliknij zieloną flagę, poczekaj chwilę i kliknij na duszka.

Czy duszek zmienia kostium, zanim na niego klikniesz?

Czy duszek zatrzymuje się po kliknięciu?

Uruchom skrypt jeszcze raz.

Czy duszek zatrzymuje się, jeżeli najedziesz na niego kursorem myszki, ale nie klikniesz? Czy zatrzymuje się, jeżeli klikniesz gdzie indziej na Scenie?



Zapisz swój projekt.

Krok 4: Dodajemy dodatkowe duszki

Teraz potrzebujemy stworzyć dwa inne duszki, abyśmy mogli zagrać w naszą grę!



Zadania do wykonania

1. Zduplikuj duszka, klikając na nim prawym przyciskiem myszki. ☐
2. Zduplikuj go jeszcze raz, tak aby były w sumie trzy duszki. ☐
3. Poprzesuwaj duszki na Scenie tak, aby były obok siebie. ☐
Jeśli chcesz, możesz je trochę zmniejszyć.

Przetestuj swój projekt

Kliknij zieloną flagę.

Wszystkie duszki powinny się zmieniać. Spróbuj zatrzymać je na tym samym obrazku!



Zapisz swój projekt.

Krok 5: Losowy kostium dla każdego duszka

Niech duszki zmieniają kostium na losowy po kliknięciu zielonej flagi.

Gdy zaczynasz grę, wszystkie duszki pokazują ten sam kostium i zmieniają się razem w taki sam sposób. Gra będzie ciekawsza (i trudniejsza) jeśli będą się zmieniać w mniej przewidywalny sposób.

Zadania do wykonania

1. Jeśli spojrzysz do zakładki `Kostiumy` dla duszka zobaczysz, że każdy kostium ma swój numer. Możesz określić, który kostium ma założyć duszek podając nazwę kostiumu lub jego numer. ☐
2. Aby duszek rozpoczął grę w losowym kostiumie dodajmy blok `zmień kostium na` z blokiem `losuj od (1) do (3)`, aby wybrać numer kostiumu. ☐

3. Możemy użyć dokładnie tego samego bloku w pętli **zawsze** tak, aby duszek zmieniał kostium na inny za każdym razem w ciągu gry.



4. Takie same zmiany zrób dla pozostałych duszków.



Przetestuj swój projekt

Kliknij zieloną flagę. Wszystkie duszki powinny zmieniać swoje kostiumy w różnej kolejności.

Jak powinniśmy zmienić nasz skrypt kiedy dodamy kolejny kostium?



Zapisz swój projekt.

Rzeczy do spróbowania

Zwiększ poziom trudności gry

Zwiększmy poziom trudności gry. Zwiększenie prędkości, z jaką zmieniają się obrazki, jest najprostym rozwiązaniem. Spróbuj czegoś bardziej wyszukanego. Oto kilka pomysłów, które mogą ci się spodobać:

- ☐ Zmień ilość kostiumów dla każdego duszka.
- ☐ Dodaj unikalne kostiumy niektórym duszkom.
- ☐ Spraw, aby kostiumy zmieniały się z różną prędkością dla różnych duszków.

Baw się wymyślając swoje własne modyfikacje!

Za każdym razem, gdy coś zmieniasz zastanów się, jaki ma to wpływ na poziom trudności gry. Czy gra robi się łatwiejsza czy trudniejsza? Jak możesz zmienić poziom trudności tak, aby był w sam raz?

Krok 6: Wyświetlamy wiadomość o końcu gry

Pokażmy naszemu graczowi wiadomość “Koniec gry” kiedy gra się skończy

Zadania do wykonania

Na początek, stwórzmy inne tło, które będziemy pokazywać kiedy gra się zakończy.

1. Kliknij w scenę, a następnie w zakładkę `Tła`. Zmień nazwę dotychczas używanego tła na “Gra”. ☐
2. Zduplicuj tło i do nowego tła dodaj tekst “Koniec gry”. ☐
Możesz zmienić rozmiar tekstu klikając na nim i

przeciągając jeden z narożników. Zmień nazwę tego tła na “KoniecGry”.

3. Przejdź do zakładki **Skrypty** dla sceny i spraw by tło “Gra” pokazywało się przy starcie gry.

4. Jak możemy sprawdzić, czy wszystkie duszki się zatrzymały? Pamiętaj, że używaliśmy zmiennej **zatrzymany**, aby sprawdzić dla każdego duszka czy został kliknięty? Sprawdźmy zmienną **zatrzymany** ostatniego duszka (Sprite3), aby stwierdzić, czy gra się skończyła. Zaznacz ostatniego duszka (Sprite3) i użyj bloku **pozycja x z Sprite3** z panelu **Czujniki**, ale zmień pozycja x na **zatrzymany**.



🚩 Przetestuj swój projekt

Kliknij zieloną flagę.

Czy wiadomość “Koniec gry” pokazała się po kliknięciu ostatniego duszka?

Co się stanie kiedy zatrzymasz ostatniego duszka zanim klikniesz na pozostałe duszki?

Zmieńmy nasz skrypt w taki sposób, żeby działał niezależnie od tego w jakiej kolejności duszki były zatrzymywane.

✓ Zadania do wykonania

1. Aby sprawdzić, czy wszystkie trzy duszki mają zmienną **zatrzymany** ustawioną na TAK, możemy użyć operatora **i**. To skomplikowany blok, który może być niełatwy do zmontowania, dlatego spróbuj i umieść wszystko razem w jednym kroku.



🚩 Przetestuj swój projekt

Kliknij zieloną flagę.

Czy wiadomość “Koniec gry” pojawia się wkiedy wszystkie trzy duszki są zatrzymane niezależnie od kolejności w jakiej na nie klikaliśmy?



Krok 7: Powiedz graczowi czy wygrał

Celem gry jest kliknięcie na każdym duszku w takim momencie, aby wszystkie miały ten sam kostium. Dobrze by było pokazać wiadomość, która będzie informować gracza, czy udało mu się wygrać.

✓ Zadania do wykonania

1. W kroku 6 napisaliśmy kod, który sprawdza czy gra się skończyła, więc wszystko co musimy teraz zrobić to sprawdzić, czy gracz wygrał. Przejdź do zakładki **Tła** i zmień tekst tła “KoniecGry” tak, by pokazywał także “WYGRANA!”. Potem zmień nazwę tła na “Wygrana”.
2. Skopiuj tło, aby utworzyć nowe z napisem “Przegrana”. Nadaj mu też nazwę “Przegrana”.
3. Teraz potrzebujemy kawałek kodu, który sprawdzi, które tło wybrać kiedy gra się skończy. Możemy użyć bloku **jeżeli ... to ... w przeciwnym razie**, by sprawdzić czy gracz wygrał czy przegrał. Porównamy każdy blok **kostium #** (numer kostiumu) używając bloku **pozycja x z Sprite** tak, jak to robiliśmy wcześniej. Tym razem zamiast zmiennej **zatrzymany**, wybierzemy blok **kostium #** i sprawdzimy czy pierwszy duszek ma ten sam kostium co drugi i czy drugi duszek ma ten sam kostium co trzeci.



Przetestuj swój projekt

Kliknij zieloną flagę.

Czy pokazuje się poprawna wiadomość na koniec gry? Co się stanie jeśli numery kostiumów się nie zgadzają (na przykład kiedy kostium nr 3 pierwszego duszka to jabłko a kostium nr 3 drugiego duszka to arbuz)?



Zapisz swój projekt.

Wyzwanie: Dodaj zmienny poziom trudności gry

Różni ludzie różnie sobie radzą z grą. Jak możesz zmodyfikować grę, aby dopasowywała swój poziom trudności do zdolności gracza?

Jednym z rozwiązań jest dopasowanie prędkości, z jaką zmieniają się kostiumy. Możesz dodać zmienną o nazwie `opóźnienie`, aby kierować blokiem “czekaj”. Jeżeli gracz wygra rundę, opóźnienie może się trochę zmniejszyć (co podniesie nieco poziom trudności gry). Jeżeli gracz przegra rundę, opóźnienie może wzrosnąć, co ułatwi nieco grę.

Prawdopodobnie będzie trzeba pomyśleć o użyciu innego sposobu na rozpoczęcie gry niż `kiedy kliknięto zieloną flagę`. Możesz wtedy przechowywać wartości w zmiennych, które będą zappamiętane pomiędzy różnymi rundami gry.



Zapisz swój projekt.

Brawo! To by było na tyle, teraz możesz się cieszyć swoją grą!

Nie zapomnij, że możesz podzielić się swoją grą ze swoimi przyjaciółmi i rodziną. Żeby to zrobić, kliknij menu Udostępnij.