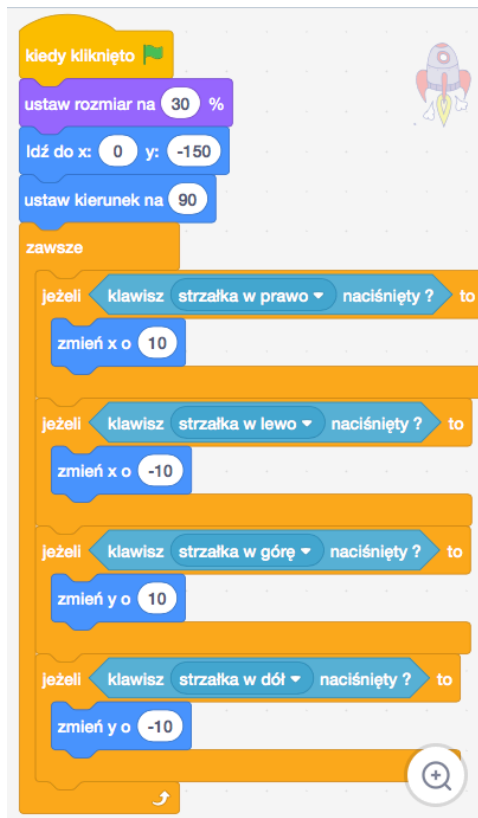
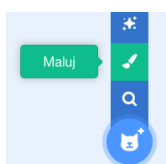


Projekt 7 – Rakieta

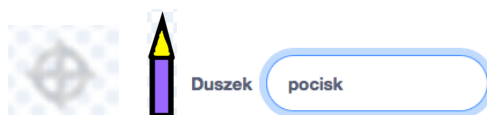
1. Usuń kota. Wybierz tło.
2. Dodaj raketę poruszaną strzałkami (**jeżeli klawisz strzałka w górę naciśnięty to zmień y o 10**). Pamiętaj o ustawieniach początkowych.



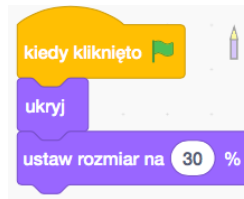
3. Dodaj pocisk rakiety.
 - Wybierz opcję Maluj w prawym dolnym rogu ekranu



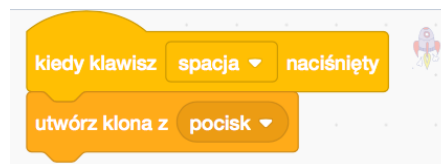
- **PAMIĘTAJ**, aby środek namalowanego duszka był w szarym kółku znajdującym się na środku pola do rysowania. Namaluj pocisk. Zmień nazwę duszka.



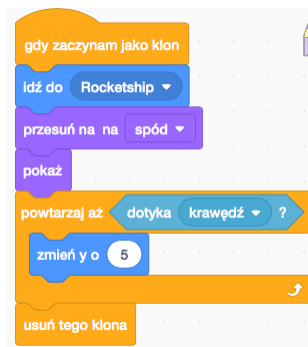
- Napisz skrypt, który po naciśnięciu zielonej flagi ukryją pocisk. Dostosuj rozmiar pocisku.



- W skryptach **dla rakiety** dodaj skrypt, który będzie klonował pocisk po naciśnięciu spacji. Przypominam, że jeśli zamiast tego umieścimy w skryptach dla pocisku bloczek **sklonuj siebie**, klonować będą się wszystkie istniejące w danej chwili klony. To sprawi, że gra będzie wolniej chodzić i będzie dostępnych „mniej” pocisków. Może też być kłopot z innymi klonami. Sprawdź to.



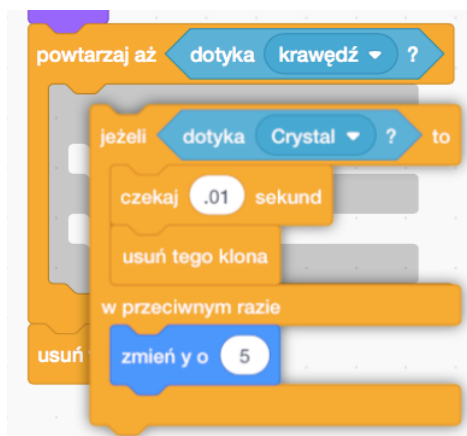
- Dodaj skrypt rozpoczynający się od **gdy zacznym jako klon**, który sprawi, że nowy klon pocisku **ustawi się tam, gdzie rakieta (pod rakieta), pokaże się, polecą do góry, oraz zniknie (klon zostanie usunięty) gdy dotknie krawędzi**. Ważne jest, aby klon został **USUNIĘTY**, a nie tylko ukryty. W Scartchu jednocześnie może występować tylko 300 klonów.



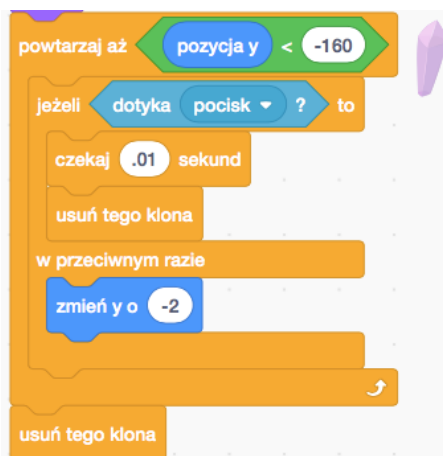
4. Dodaj spadające asteroidy (narysuj je, lub wybierz coś z istniejących duszków), które gracz będzie mógł zestrzelić. Asteroidy **pojawiają się w losowym punkcie górnej części ekranu, lecą w dół i znikają, kiedy dotkną krawędzi ekranu lub pocisku**.



- W skrypcie dla pocisku, dodaj w pętli **powtarzaj aż** instrukcję warunkową **jeżeli dotyka asteroidy to czekaj .01, usuń tego klona, w przeciwnym razie zmień y o 5**. Ponieważ po wystąpieniu kolizji pocisku z asteroidą, oba duszki znikną, klony nie mogą być usuwane zaraz po wykryciu kolizji. W przeciwnym razie kolizję wykryje tylko ten duszek, który zniknie jako pierwszy – sprawdź to.

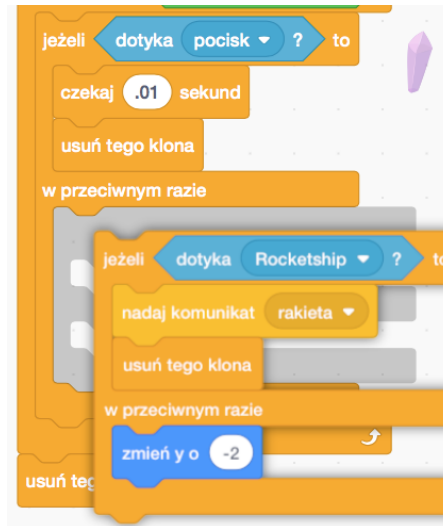


- Podobnie przekształć fragment kodu odpowiedzialny za wykrywanie kolizji asteroidy z pociskiem.

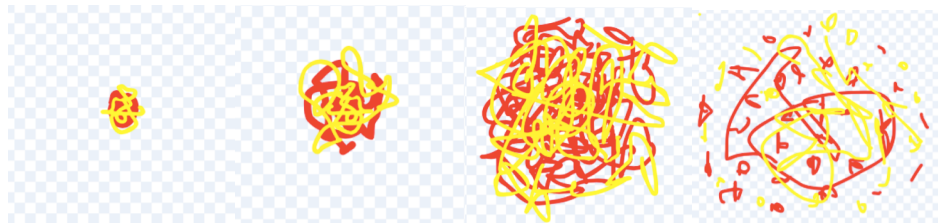


5. Zaprogramuj kolizję asteroidy z rakietą.

- W skrypcie asteroidy dodaj fragment kodu odpowiedzialny za wykrywanie kolizji asteroidy z pociskiem. Ponieważ po dotknięciu rakiety asteroida zniknie, zamiast wykrywać kolizję również w skryptach rakiety, asteroida po wykryciu kolizji **nada komunikat rakietą**. Rakieta oraz wybuch będą reagować na komunikat, a nie na wykrytą kolizję z asteroidą.



- Zaprogramuj wybuch. Narysuj kilka etapów wybuchu.



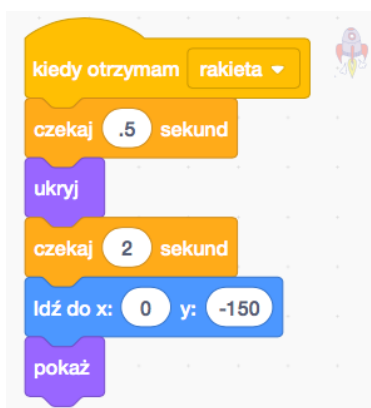
- Dodaj skrypt, który po rozpoczęciu gry ukryje wybuch. Po otrzymaniu komunikatu **rakieta** wybuch pójdzie na miejsce rakiety i pokaże się, kostium po kostiumie. Na koniec ukryje się. Aby móc ponownie wykorzystać ten sam fragment kodu, gdy wybuchnie coś innego, zaprogramujemy własny bloczek. Bloczek będzie przyjmował jeden parametr, który pozwoli nam ustawić rozmiar wybuchu.

W grupie instrukcji **Moje bloki** kliknij przycisk, **Utwórz blok**, wpisz nazwę bloku, następnie kliknij przycisk **Dodaj dane wejściowe**, kliknij **ok**. Na ekranie pojawi się nowy bloczek, który możemy teraz zdefiniować. Zmienną **number or text** bierzemy z czerwonego bloczku definiującego wybuch.

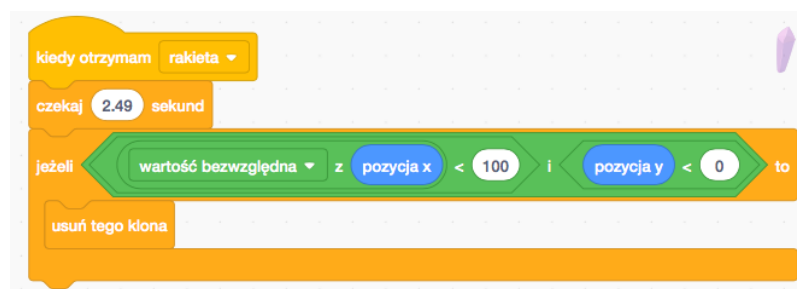




- Dodaj skrypt dla rakiety, który sprawi, że po otrzymaniu **komunikatu rakieta** rakieta ukryje się, poczeka 2 sekundy, pójdzie na środek dolnej części ekranu, i pokaże się. Ponieważ w trakcie trwania gry rakieta będzie znikać, dodaj bloczek **pokaż** w ustawieniach początkowych.



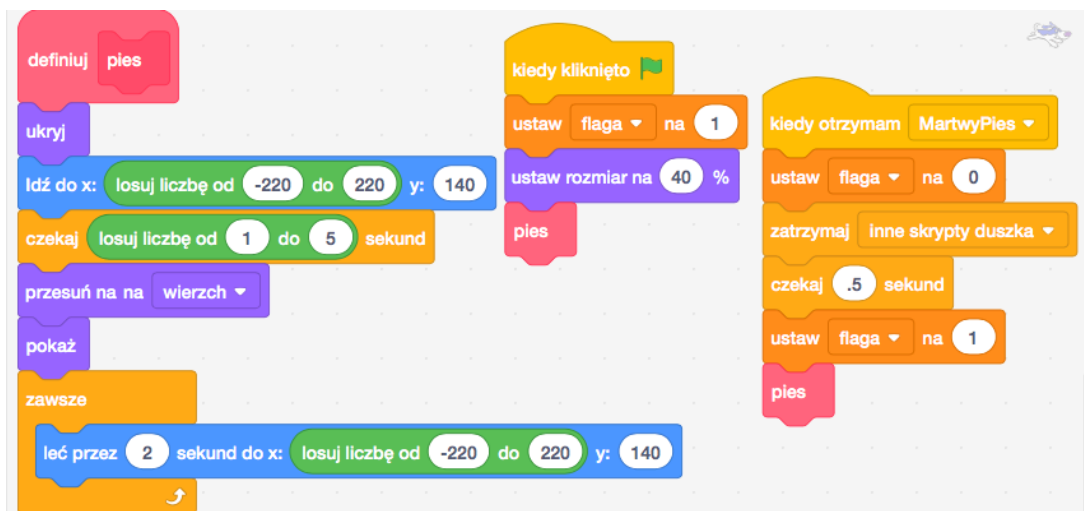
- Aby zapobiec pojawieniu się rakiety w miejscu, w którym są już asteroidy, co spowodowałoby jej ponowny wybuch, krótko przed pojawieniem się rakiety usuń wszystkie asteroidy znajdujące się w pobliżu rakiety. Wykorzystaj do tego komunikat **rakieta**, oraz bloczki **jeżeli wartość bezwzględna z pozycja x < 100 i pozycja y < 0 to usuń tego klona**.



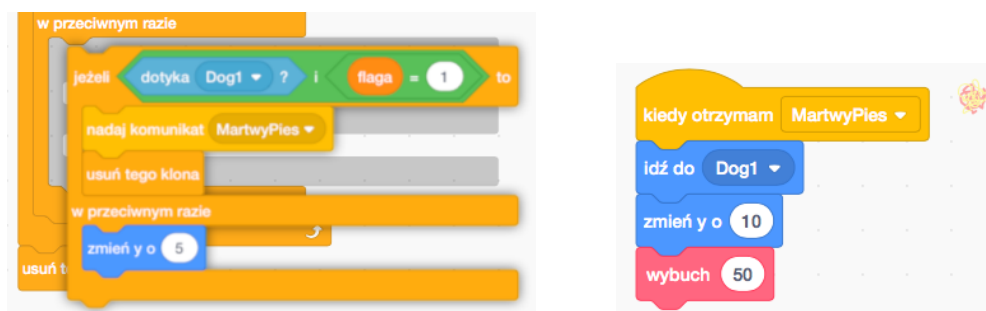
6. Dodaj wrogi statek kosmiczny. Aby dwa razy nie pisać tego samego kodu, zdefiniuj nowy bloczek, u mnie jest to **pies**. W definicji tego bloku **ukryj wrogi statek, idź do losowej pozycji w górnej części ekranu, czekaj losowy czas od 1 do 5 sekund, przesuń się na wierzch, pokaż się**. Następnie w pętli **zawsze leć przez 2 sekundy do losowego miejsca w górnej części ekranu**.

Dodaj skrypt, który po kliknięciu zielonej flagi ustawi odpowiedni rozmiar a następnie wywoła funkcję **pies**.

Kiedy pocisk wykryje kolizję z psem, nada komunikat **MartwyPies**. Kiedy wrogi statek otrzyma komunikat **MartwyPies**, zatrzyma inne swoje skrypty (trafiony statek musi przestać latać), poczeka chwilę, tak, aby miał czas wybuchnąć, wywoła funkcję **pies**. Ponieważ trafiony statek będzie jeszcze chwilę stał, aby zapobiec ponownemu trafieniu go przez pocisk, ustawimy **flagę na 0**. Poza tym **flaga** jest przez cały czas ustawiona na **1**. Pocisk może trafić we wrogi statek, tylko wtedy, gdy **flaga** jest ustawiona na **1**. Sprawdź co się stanie, kiedy nie ustawimy flagi, a wrogi statek zostanie w krótkim czasie wielokrotnie trafiony.



7. Zmień fragment kodu w skryptach dla pocisku, tak aby po wykryciu kolizji z wrogim statkiem wysyłał komunikat **MartwyPies**. Do wybuchu dodaj skrypt, który zareaguje na komunikat **MartwyPies**.



8. Naucz wrogi statek strzelać.
9. Teraz rakieta może uciec, nawet jeśli już zacznie wybuchać. Spraw, aby w momencie wybuchu rakiety klawisze strzałek nie działały. Możesz użyć do tego flagi.
10. Dodaj licznik punktów oraz żyć. Obie zmienne powinny być widoczne na ekranie. Ustaw warunek końca gry.