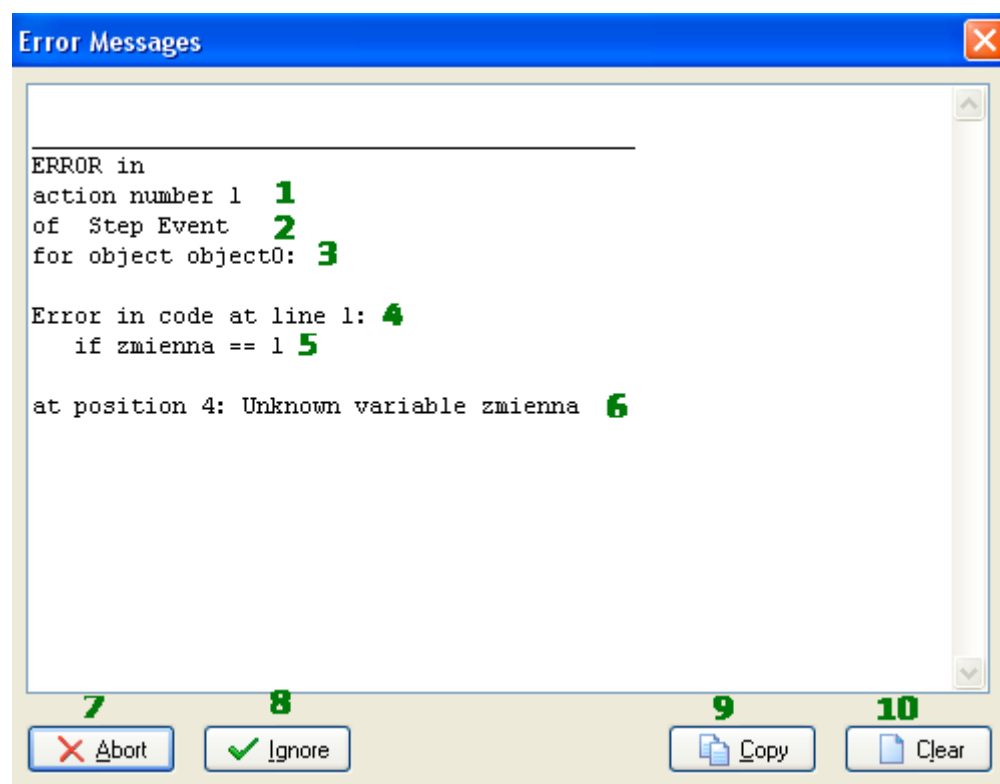


O błędach i ich przyczynach

Podczas pracy z Game Makerem, możemy natrafić na różnego rodzaju błędy. Niektóre możemy zignorować podczas wyświetlenia – jednak niektóre wymuszają wyłączenie gry. W tym artykule postaram się opisać wyskakujące błędy, ich przyczynę oraz jak je naprawiać.

Error Messages

W większości przypadków, okno z informacją o błędzie wskaże nam miejsce gdzie takowy wystąpił – przez co nie powinniśmy mieć problemów z wyszukaniem problemu.



- 1 – numer akcji, w której wystąpił błąd („1” to ta na samej górze eventu).
- 2 – Event (wydarzenie), w jakim wystąpił błąd. W tym wypadku jest to step.
- 3 – Informacja, w jakim obiekcie wystąpił błąd (w tym wypadku object0).
- 4 – Linia w jakiej wykryto błąd. W tym wypadku jest to pierwsza linia kodu. Używając wbudowanego edytora, można bez problemu „skoczyć” do szukanej linii używając „Go to a particular line”.
- 5 – Część kodu, w którym wykryto błąd
- 6 – Opis błędu. W tym wypadku nieznana zmienna.
- 7 – Przerwywamy działanie gry.
- 8 – Ignorujemy napotkany błąd - nie zawsze jest to jednak skuteczne.
- 9 – Kopiujemy cały raport do schowka
- 10 – Czyścimy okno błędu.

Rodzaje Błędów

Każdemu, nawet najlepszemu mistrzowi nie raz wyskoczą różnego rodzaju błędy. W większości będą to problemy łatwe do naprawienia, drobne przeoczenia. Zdąży się też jednak i takie, nad którymi będziemy siedzieć i głowić się jak go rozwiązać ;d.

Błąd	Tłumaczenie	Opis	Możliwość Zignorowania
<u>Unknown variable</u> <nazwa>	Nieznana zmienna <nazwa>	Musimy pamiętać, że przed jakąkolwiek operacją z użyciem zmiennej, musimy ją uprzednio zadeklarować. Zazwyczaj robimy to w create, lecz nie jest to konieczne. Zasada jest taka, że przed użyciem zmiennej musimy ją zadeklarować, nadać jakąś początkową wartość. Robimy to w taki sposób : nazwa = dowolna_liczba. Innym rozwiązaniem problemu jest zaznaczenie w ustawieniach Game Makera opcji „Treat uninitialized variables as value 0”. Znajdziemy ją w : Global Game Settings >> Errors.	ISTNIEJE. Ale w tym wypadku ignorowanie błędu jest bez sensu – gdyż błąd będzie się wyświetlał tak długo, dopóki zmienna nie zostanie wykryta. Może to nastąpić po jakimś czasie, gdy mamy ustawiony np. alarm, gdzie została zadeklarowana zmienna.
<u>Division by 0</u>	Dzielenie przez zero	W jakiejś operacji matematycznej, chcemy podzielić dowolną liczbę przez zero. W programowaniu nie jest to dopuszczalne. Pozostaje nam tylko zmiana dzielnika liczby.	NIE ISTNIEJE.
<u>Unknown variable „nazwa” or array index out of bounds.</u>	Nieznana zmienna „nazwa”, lub indeks tablicy poza jej granicami.	Tworząc tablice, ustalamy również jej granice. Możemy ją powiększać, dopisując kolejne wartości, ale nie możemy bezpośrednio odwoływać się do indeksu poza jej granicami. Przykładowo : Tablica o nazwie „array” ma rozmiar 8, więc nie możemy odwołać się do indeksu 9 : zmienna = array[9] . – przed wykonaniem tej operacji musi być dopisana tam jakaś wartość np. array[9] = 5 . Aby powiększyć tablicę.	ISTNIEJE. Po zignorowaniu, Game Maker nie wykona po prostu obliczeń.

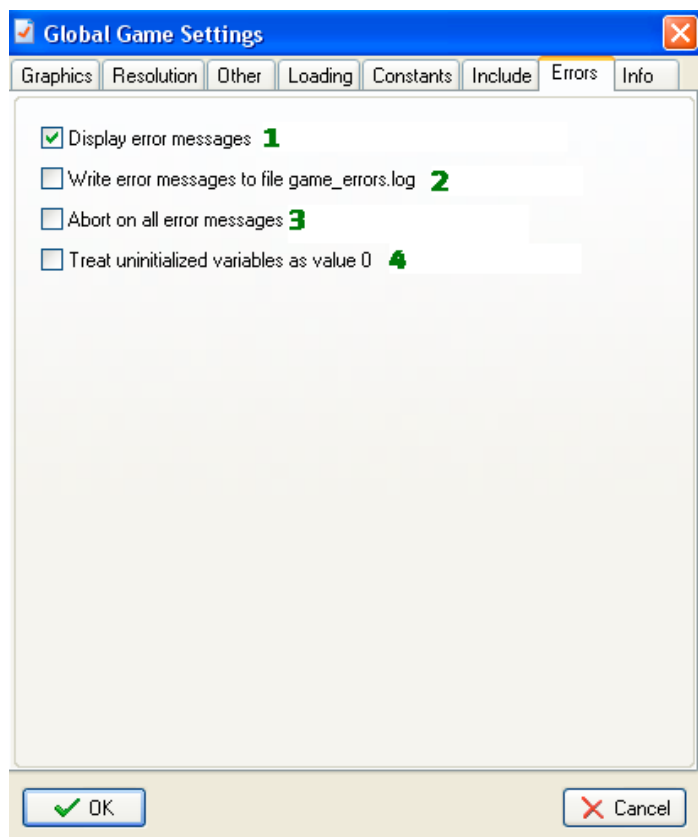
<u>Negative array index</u>	Negatywny indeks tablicy	Tablica nie może mieć ujemnego indeksu – przeszukaj kod. Często zdarza się w przypadku, gdy jakaś zmienna posiada wartość ujemną, a my próbujemy za jej pomocą odwołać się do jakiegoś indeksu tablicy (tablica[zmienna]).	ISTNIEJE. Po zignorowaniu, Game Maker nie wykona obliczeń.
<u>Error defining an external function.</u>	Błąd w definiowaniu zewnętrznej funkcji.	Błąd ten głównie występuje, gdy korzystamy z funkcji biblioteki dll., a nie umieściliśmy jej w tym samym folderze co gra. Większość bibliotek posiada również tzw. Funkcję inicjalizującą, którą musi być wykonana przed używaniem reszty funkcji. Możliwe, że o niej zapomniałeś. Błąd może występować również w innych wypadkach – ale jak nie jesteśmy twórcą danego dll'a nic więcej nie możemy zrobić.	ISTNIEJE. tak jak w wypadku „ Unknown variable <nazwa>”, Game Maker nie będzie dawał nam spokoju póki problem nie będzie rozwiązany. Krótko mówiąc ‘Ignore’ nic nam nie da ;]
<u>Wrong number of arguments to function or script.</u>	Zła liczba argumentów w funkcji bądź skrypcie	Do większości funkcji możemy wpisać tylko określoną liczbę argumentów, niezbędną do wykonania obliczeń. Jeśli jest ona nieprawidłowa – zostaniemy o tym poinformowani.	NIE ISTNIEJE.
<u>Cannot compare arguments.</u>	Nie można porównać argumentów	Pamiętajmy, że porównywać można ze sobą zmienne tylko tego samego typu. W GML istnieją dwa typy : real (liczby rzeczywiste) oraz string (ciąg znaków), jeśli zmienna „a” będzie liczbą, a zmienna „b” ciągiem znaków, kod : if a == b { game_end(); } będzie nieprawidłowy.	NIE ISTNIEJE.
<u>Unexpected symbol in expression.</u>	Niepożądany symbol w wyrażeniu	Przyczyną ich występowania może być pomyłka bądź też niewystarczająca wiedza GML. Możliwość jest bardzo wiele więc nie będę ich wszystkich wymieniał. Unikniemy ich – jeśli dobrze poznamy składnię oraz zasady panujące w tym języku.	NIE ISTNIEJE.

		Symbole takie jak !, ^, ^^, & itd. Używamy w konkretnych sytuacjach – nie możemy natomiast stawiać ich wszędzie gdzie nam się podoba. Odsyłam do kursów GML.	
<u>Wrong type of arguments to <symbol></u>	Złe typy argumentów dla <symbolu>	Przy operacjach takich dodawanie (+), odejmowanie (-), dzielenie (/), mnożenie (*), dzielenie całkowite (div) oraz reszta z dzielenia (mod) – używać możemy tylko liczb rzeczywistych (tzn. po lewej i po prawej stronie tego symbolu muszą znaleźć się zmienne typu real. Oprócz tego znakiem „+” możemy łączyć ciągi znaków	NIE ISTNIEJE.
<u>Symbol <symbol> expected</u>	Wymagany symbol <symbol>	Pojawia się, kiedy zapomnieliśmy dopisać jakiegoś znaku np. zamknąć nawiasu podczas pisania funkcji, Zamknąć blok klamrą.	NIE ISTNIEJE.

Może też się zdarzyć, że twój kod będzie poprawnie napisany – a gra będzie się ciągle wyłączać. Powodem tego mogą być zbyt obciążające obliczenia bądź funkcje, z którymi Game Maker nie może sobie poradzić. Przykładem może być funkcja `draw_getpixel()`, którą należy stosować z umiarem.

Kontrola nad błędami

Istnieje kilka opcji, które mogą pomóc nam w kontrolowaniu błędów. Znajdziemy je wchodząc do Global Game Settings w zakładce ‘Errors’.



- 1** – Jeśli to odznaczymy, nie będą wyświetlane okienka z błędami. Jeśli błąd nie może być zignorowany – gra zostanie wyłączona.
- 2** – Po zaznaczeniu, każdy błąd zostanie zapisany do pliku game_errors.log.
- 3** – Po wystąpieniu jakiegokolwiek błędu, Game Maker przerywa działanie gry.
- 4** – Wszystkim nieznanym zmiennym, które wystąpiły w kodzie – zostaje przypisana wartość 0 (patrz tabelka - Unknown variable „nazwa”).