

SilverCoders

DIGITAL LITERACY IMPROVEMENT THROUGH EFFECTIVE
LEARNING EXPERIENCES FOR ADULTS



CHALLENGE #18 **ADVANCED COIN FETCHER**

**CODING TRAINING PROGRAMME
FOR +55 ADULTS**



SILVER CODERS

ERASMUS+ No. 2020-1-SE01-KA227-ADU-092582



**Co-funded by
the European Union**

This document reflects only the author's view and the National Agency and the European Commission are not responsible for any use that may be made of the information it contains

STRUCTURE OF THE CHALLENGE

DESCRIPTION

This lesson takes the game developed in the previous challenge and develops it further, making it more complex and attractive.

GENERAL GOAL

This lesson continues to promote the understanding of the Gdevelop environment and how it can be used to code. It focuses on additional relevant GDevelop concepts like variables. It also explains how to use TextBoxes.

LEARNING OBJECTIVES

In the end of this challenge, the learner will be able ...:

- To have experience with a visual programming suite and to code a small piece of software with it.
- To know what statements and command lines are.
- To write instructions using correct syntax.
- To be able to use If statements correctly to execute code according to a certain defined fixed condition.
- To use the Gdevelop editor
- To understand the concepts of scenes, events and objects
- To understand the concept of variables.

INSTRUCTIONS

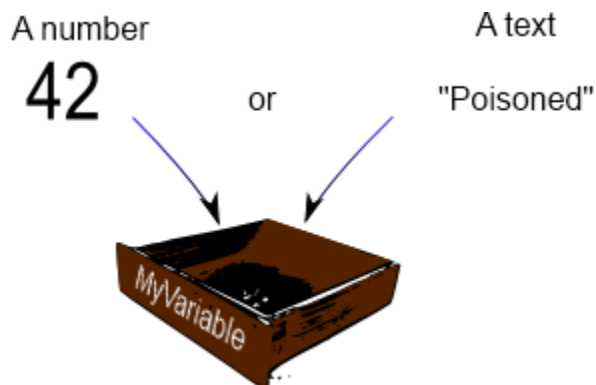
Welcome back to the coding and programming challenges.

In this challenge we will develop further the game with Kenney while we learn about variables and TextBoxes.

- Open the Challenge 18 – Advanced Coin Fetcher – Initial. This is the same file that you finished in the last challenge.

Variables

A Variable is a name or identifies that represents some data. For example, a variable can store a number or a text. We might compare them to drawers or boxes where we can file notes. Any data can be stored in a variable, as long as it is in text or number format. Such things as the number of lives a player has remaining, a player's high score, the number of bullets left, and/or the number of enemies killed are all examples of what can be stored in a variable. You are probably going to store numbers in variables as a common practice.



Actions and conditions test or change a variable value.

Let's create a wallet for Kenney so that he knows how many Coins he already collected. Let's start by creating a variable Wallet. This is an integer value that tells us how many Coins Kenney has.

We will create a scene variable, that is a variable that can only be used in this scene.

On the layout, right-click and choose »Open scene properties« and then »Edit scene variables«

Scene properties

Window title

☒ Stop music and sounds on startup

Scene background color
209;209;209

Any additional properties will appear here if you add behaviors to objects, like Physics behavior.

Edit scene variables

Ok

Do »Add a variable«

Call it Wallet, as Number and with initial value 0 (Kenney has no coins in the beginning).

Scene Variables

Copy Paste Delete Undo Redo Search variables + Add variable

Wallet 123 Number 0

Now, every time Kenney catches a Coin we increment (add 1) the value of the variable Wallet. On the same condition (Kenney collides with the Coin), we are adding a second action.

Search objects or actions

Objects Other actions

GENERAL

- Objects
- Conversion
- Variables
- Global variables
- Scene variables

Value of a scene variable

Change the value of a scene variable.

Variable: Wallet

Modification's sign: + (add)

Value: 1

This is the code

if kenney is in collision with the coin, delete the coin and add 1 to the Wallet

Kenney is in collision with Coin

Add condition

Delete Coin

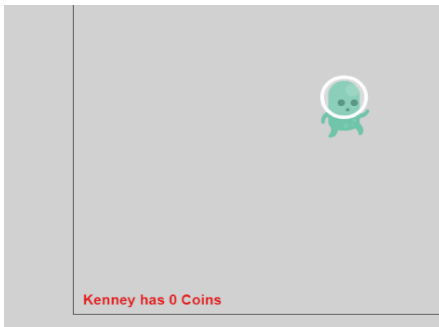
Change the scene variable Wallet: add 1

Add action

But we need something to show us how many Coins Kenney has. Let's create a Text Box for that. A Text Box is an object that shows text on the screen. On the "Object Window", select »Add a new object«, then »New object from scratch« and »Text«

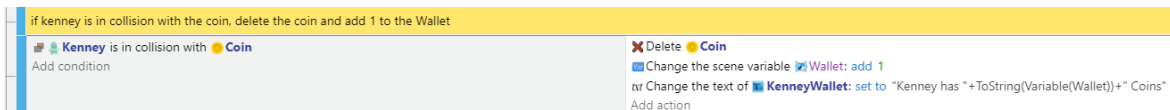
Lets' call it KenneyWallet. The initial text is »Kenney has 0 Coins«

Let's put the Text Box in the bottom of the layout (just drag it).



Now, every time Kenney picks a Coin, the number of coins changes and this must be reflected on the Text Box. Again, let's add a condition to the previous condition.

Here is the code



So, we are changing the text of the box. To do that we create a String (a sequence of characters) that results from the concatenation (we use the “+” sign for that) of the “Kenney has “ (the commas represent a String) with the value given by Variable(Wallet) (ToString() converts that into a String). Finally we concatenate with the string “ Coins”.

HOPE YOU ENJOYED! SEE THE FINAL VERSION OF THE CHALLENGE AS IT HAS SOME ADDITIONAL FEATURES.

Note: The scope of a variable determines the place a variable can be accessed. In GDevelop, there are three variable scopes are available:

- Global variables are accessible from all the game scenes. For instance, they can be used to store the player's score across different levels/scenes.
- Scene variables are only accessible from the scene they are created in. They can be used for data that only concerns one scene. A Scene variable would be able to access the time remaining to complete the level/scene.
- Object variables only concern one object. For example, a hero can have a “Health” or “Ammo” variable.

RESOURCES

Challenge 18 (Initial)