



Move: Player Gravity


Part 1 of 3


In a *platform* game, make a player object move when arrow keys are pressed






Object_Player

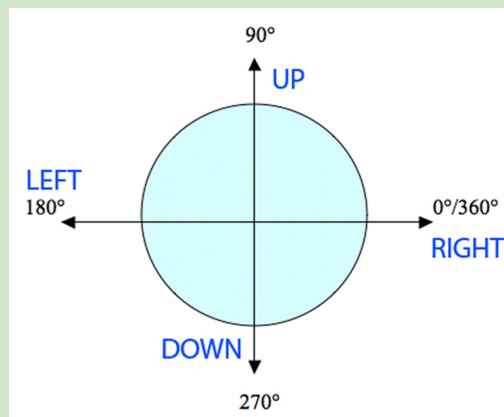
 **Event: Step <Step>**
 Action: Check Empty
 Applies to: Self
 x: 0
 y: 1
 Objects: Only Solid
 Check Relative
 No NOT

 **{same event}**
 Action: Set Gravity
 Direction: 270
 Gravity: 1
 Check Relative

 **[same event]**
 Action: Else

 **[same event]**
 Action: Set Gravity
 Direction: 270
 Gravity: 0
 Not Relative

Events:	Actions:
 Step	 If a position is collision free  Set the gravity  Else  Set the gravity




When setting gravity, Game Maker uses degrees to set direction because the gravity does not relate to (x,y) coordinates in the game room.

This will set the gravity for the object_player. **Check Empty** will check if there is anything under the player and set the gravity to 0 if there is a solid object. **Else** means if there isn't a solid object underneath, set the gravity to 1 to drop the player.

To make your object_player move left, right, and up- you will need a **solid** object_wall

Object_Player


 **Event: Keyboard <Left>**
 Action: Check Empty
 Applies to: Self
 x: -1
 y: 0
 Objects: Only Solid
 Check Relative
 No NOT





Events:	Actions:
 Step  <Left>	 If a position is collision free  Jump to position {-5,0}

Continued on Part 2 of 3




Move: Player Gravity


Part 2 of 3


 **{same event}**
 Action: Jump to Position
 Applies to: Self
 x: -5
 y: 0
 Check Relative


Events:	Actions:
 Step  <Left>	 If a position is collision free  Jump to position [-5,0]







 **Event: Keyboard <Right>**
 Action: Check Empty
 Applies to: Self
 x: 1
 y: 0
 Objects: Only Solid
 Check Relative
 No NOT

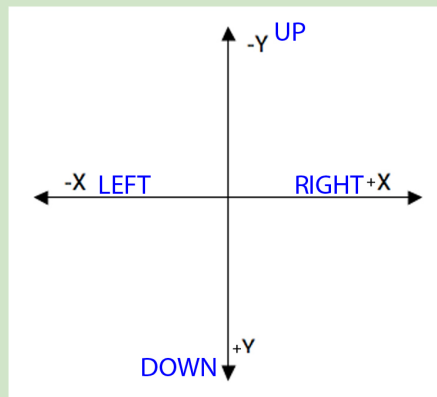
Events:	Actions:
 Step  <Left>  <Right>	 If a position is collision free  Jump to position [5,0]

 **{same event}**
 Action: Jump to Position
 Applies to: Self
 x: 5
 y: 0
 Check Relative

 **Event: Key Press <Up>**
 Action: Check Collision
 Applies to: Self
 x: 0
 y: 1
 Objects: Only Solid
 Check Relative
 No NOT

 **[same event]**
 Action: Speed Vertical
 Applies to: Self
 Vert. Speed: -14
 Not Relative

Events:	Actions:
 Step  <Left>  <Right>  press <Up>	 If there is a collision at a position  Set the vertical speed



When setting player movement, Game Maker uses (x,y) coordinates. Remember that the y-axis is flipped, so positive y-coordinates go down instead of up.

Move: Player Gravity

Part 3 of 3

Event: Collision <Wall>

Action: Move to Contact

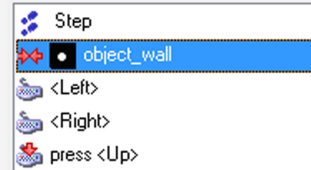
Applies to: Self

Direction: direction

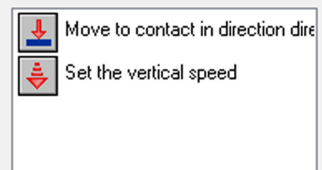
Maximum: 12

Against: Solid Objects

Events:



Actions:

**{same event}**

Action: Speed Vertical

Applies to: Self

Vert. Speed: 0

Not Relative

Important Notes:

When you place your object_player in the game room, it MUST be above the wall. If the game starts with object_player touching object_wall, the player will become stuck and immobile.

Test out how the player moves between the platforms. How high can he jump? How many blocks can he jump? Platform games are harder to design than free form or maze, because it is possible to get stuck without a way to complete a level.

The design of the levels is more important than the height or movement speeds. Be careful not to design your game and then change your vertical speed in a way that the player can't access the platforms. You should also be careful not to design the game and then make it too easy by increasing his movement speed.

The other cards in this packet will apply to platform games, but you should ignore any player movement changes unless you are using a card that states PLATFORM GAME ONLY at the top