

Mack Tang  
February 28th, 2018  
Programming Projects  
Project (Homework 3)

Link to Version Control: <https://github.com/macktang/AtomJumper.git>

### **Planned for Today's Assignment:**

Wk #	Required Learning	Task	Due Date and Time Allotted	Deliverables for Due Date
1	pygame, python OOP	Learn Pygame, Work through Pygame tutorials		Code for pygame tutorials
2	pygame, python OOP	GUI functioning, Basic movement working	Feb 28 (HW 3) [2 weeks]	Code for basic GUI functioning

### **Actual Accomplished for Today's Assignment:**

1. Used tutorial videos 1-5 of the following series (<https://www.youtube.com/watch?v=uWvb3QzA48c>), as a reference and evolved this into my own game. I also researched about pygame on the official pygame docs to know about the capabilities of the pygame functions and classes I was using. I did not follow the tutorial's code exactly, but instead used the tutorial code as a reference for making my own game.
2. **Code for basic GUI functioning, with player movement working.** Using what I learned from the pygame tutorials, I made a game that is now very close to the "Amazing Brick" mobile game:

#### Features of Current Code

- Physics act on player (such as falling with gravitational acceleration), uses pygames built-in vector objects to store the player's velocity.
- Functionality: spawn the player in, and also spawn 2 gates, which the player can jump past.
- Pacman style wrapping from left to right (moving too far right will teleport the player to the left of the screen, and vice versa)
- Game scrolls vertically along with the player.
- All code implemented in Object Oriented style

## Plan for Next Deadline

Wk #	Required Learning	Task	Due Date and Time Allotted	Deliverables for Due Date
3,4		Game working		Code for final game
5		Unintelligent bot 1 moves working (moves left or right randomly)	Mar 21 (HW 4) [3 weeks]	Including unintelligent bot moving

I would not like to make any changes to the above plan, but here are some more detailed requirements:

Task: Game Working

- Make gates infinitely spawn (instead of only spawning 2)
- Add score on-screen
- Add menu page
- Add sound
- Tune up player controls (make jump bigger, remove up arrow key “center jump”)
- Make player die if they hit gates

-Add Testing Framework

Task: Unintelligent bot 1 moves working

- Avoids jumping if jumping will make player sprite collide with an obstacle

## Screenshots of Project

