2P Tank Fight

1	Start a new project and delete the Cat sprite

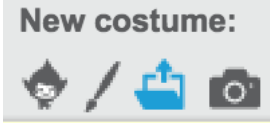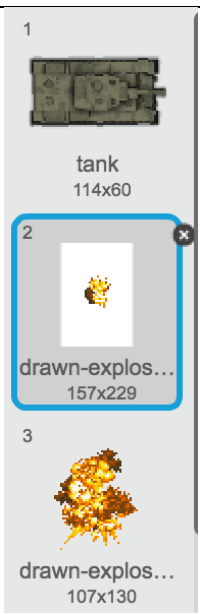| 2	For this game, we need to download some resources. These resources are pictures and sounds. **Go** to these resources using your **browser** and **download** to your local computer. Remember the folder you put them in so you can find them later.	• Tanks - https://opengameart.org/sites/default/files/i3ulu8_strip8_0.png • Shell - http://pixelartmaker.com/art/db779e6958ac762.png • Explosion - http://moziru.com/explore/Drawn%20explosion%20animated%20gif%20transparent%20background/#gal_post_1273_drawn-explosion-animated-gif-transparent-background-2.gif • Tank Firing Sound - http://soundbible.com/1326-Tank-Firing.html • Bomb Sound - http://soundbible.com/1234-Bomb.html

3	Click the **Stage** icon beside the **Sprites** bar. Click the **Backdrops** tab. Use the painting tools to draw the following stage. Use the **Fill with Color** and **Rectangle** shapes to paint the background and make the walls. Make sure the colours are very different. You should have something similar to this.	

4	Use the **Upload from file** icon from the **New sprite** bar. Find the <u>Tank</u> file you have downloaded earlier and open. Name this sprite *Tank1*.	

4	Click the **Costumes** tab Your *Tank1* costume should have a number of tanks. Use the **Select** tool to draw rectangles around the tanks you don't want and delete until you have one left. Use the **Set costume centre** tool to make sure the sprite is centred. Don't forget to name this costume *tank*	
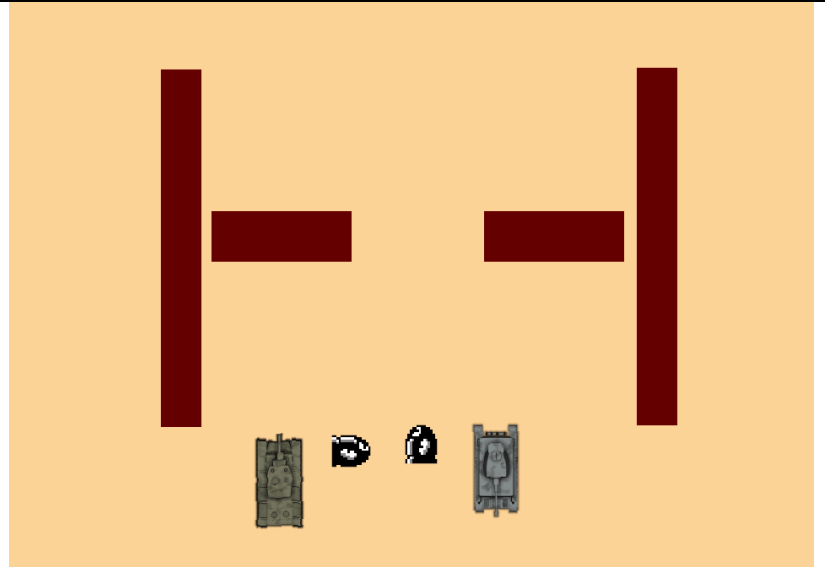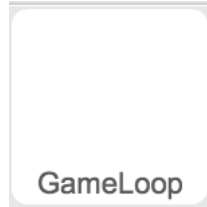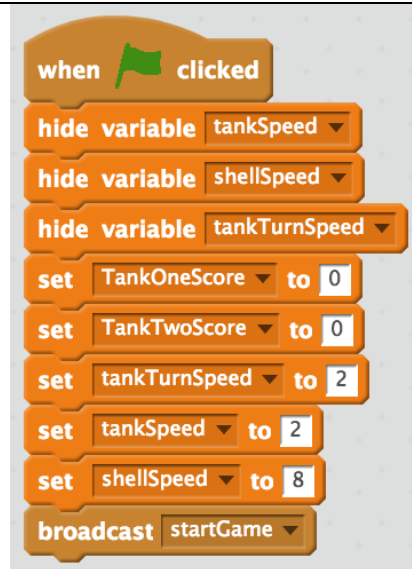
2P Tank Fight

5	On the **Costumes** tab, use the **Upload from file** to select the <u>Explosion</u> file that you downloaded earlier. You should now see a number of new costumes. Delete any that don't have a transparent background New costume:	
6	Click the **Sounds** tab and remove the existing pop sound. Use the **Upload from file** and select the <u>Tank Firing</u> sound file. Do the same action again and select the <u>Bomb</u> sound file.	
7	Duplicate the *Tank1* sprite and rename to *Tank2*. Click the **Costumes** tab and delete the existing *tank* costume. Use the **Upload from file** to select the <u>Tank</u> file from earlier. Repeat step 4.	

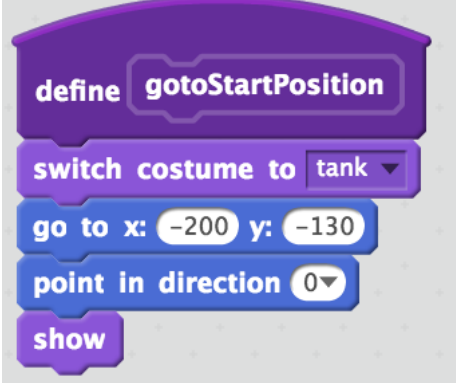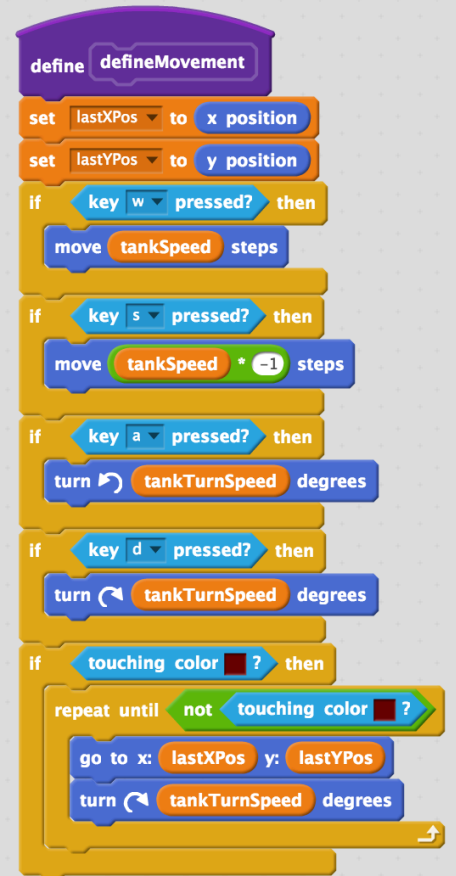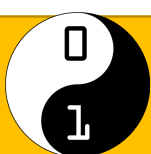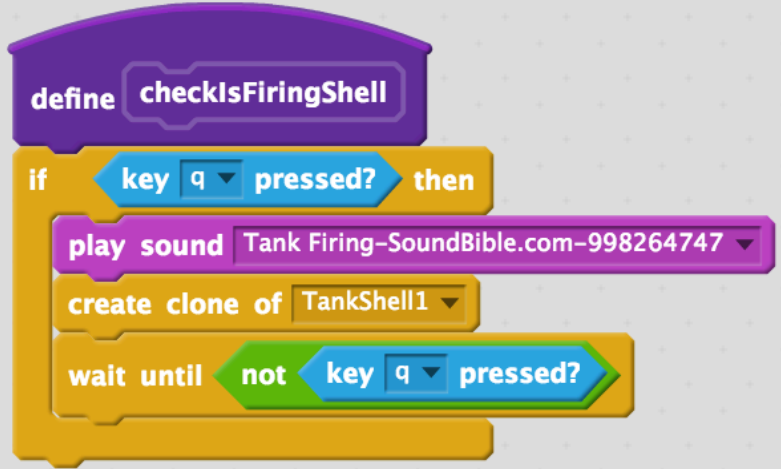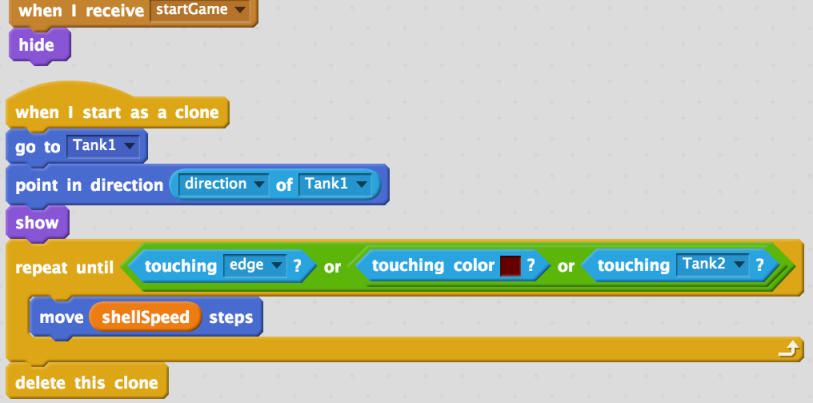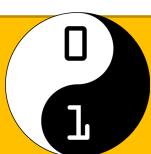|		
8	From the **New sprite** bar, use the **Upload from file** and select the <u>Shell</u> file you downloaded earlier. Name this sprite *TankShell1*. Duplicate this sprite and rename to *TankShell2*	TankShell1 TankShell2
9	Use the **Shrink** tool to size all your sprites relative to your stage. Your stage should look something similar to this.	
10	We have finished importing all our resources. We can now start coding behaviour!	
11	From the **New sprite** bar, select **Paint new sprite**. Name this sprite *GameLoop.* In complex Scratch games, it's best to use a sprite that manages our game variables and sends a signal to everyone when to start the game. In that way we avoid timing problems with our sprites.	GameLoop
12	Place the following script blocks on the *GameLoop* sprite. It will involve creating 5 variables (for all sprites) that will manage our tank and shell speed as well as scores. We will also hide the variables we don't want to see, and send a message to all the other sprites to start the game.	when 🚩 clicked hide variable tankSpeed hide variable shellSpeed hide variable tankTurnSpeed set TankOneScore to 0 set TankTwoScore to 0 set tankTurnSpeed to 2 set tankSpeed to 2 set shellSpeed to 8 broadcast startGame

2P Tank Fight

13	Select *Tank1* sprite and click the **Scripts** tab. Place the following script block. Use the **More Blocks** category and **Make a Block** to create the 4 purple blocks. This will be our main tank behaviour loop that will last the entire game.		when I receive startGame ▼ gotoStartPosition forever defineMovement checkIsHitByEnemy checkIsFiringShell
14	Place the following script blocks under the **gotoStartPosition** definition block. This will make our tank sprite to show its tank costume, move to a certain position and appear every time we use the **gotoStartPosition** block.		define gotoStartPosition switch costume to tank ▼ go to x: −200 y: −130 point in direction 0 ▼ show

15	Place the following script blocks under the **defineMovement** definition block. This will make our tank move when we press our direction keys (WASD You will need to create two variables <u>for this sprite only</u>, lastXPos and lastYPos. This and the **If touching color** (matching the color of the wall zones) will try stop the tank sprite from crossing the wall zones.		define defineMovement
			set lastXPos ▼ to x position
			set lastYPos ▼ to y position
			if key w ▼ pressed? then
			 move tankSpeed steps
			if key s ▼ pressed? then
			 move tankSpeed * −1 steps
			if key a ▼ pressed? then
			 turn ↺ tankTurnSpeed degrees
			if key d ▼ pressed? then
			 turn ↻ tankTurnSpeed degrees
			if touching color ■ ? then
			 repeat until not touching color ■ ?
			 go to x: lastXPos y: lastYPos
			 turn ↻ tankTurnSpeed degrees

CoderDojo Kells

2P Tank Fight

16	Place the following script block under the **checkIsHitByEnemy** definition block. This will tell our tank what to do if it is hit by the enemy shell (*TankShell2*). It will play the explosion sound and loop through all the costumes (19 comes from the number of costumes in the **Costumes** tab). We will change the score for *Tank2*, wait some seconds, and go back to the start position.	define checkIsHitByEnemy if touching TankShell2 ? then play sound Bomb-SoundBible.com-891110113 repeat until costume # = 19 next costume hide change TankTwoScore by 1 wait 2 secs gotoStartPosition
17	Lastly for *Tank1*, we place the following script blocks under the **checkIsFiringShell** definition block. This will tell our tank what to do when the key to fire (Q) is pressed. It will create a clone of *TankShell1* and the **wait** block will prevent the user from holding the key to keep firing.	define checkIsFiringShell if key q pressed? then play sound Tank Firing-SoundBible.com-998264747 create clone of TankShell1 wait until not key q pressed?
18	Click the *TankShell1* sprite. Place the following script blocks. This will make the tank shell hide when the game starts. It will also handle when it starts as a clone (fired from *Tank1*). It will start at *Tank1* and move in *Tank1*'s direction until it hits the edge, the wall zone, or the enemy tank.	when I receive startGame hide when I start as a clone go to Tank1 point in direction direction of Tank1 show repeat until touching edge ? or touching color ? or touching Tank2 ? move shellSpeed steps delete this clone

19	Click *Tank2* sprite.
	Repeat the steps 13-17 but changing the following:
	
	• **goToStartPosition** definition block
	   o change the **go to x y** coord to x:200 and y:130
	   o change the **point in direction** to 180
	• **defineMovement** definition block
	   o change the **key pressed?** values from WASD to the arrow Up, Down, Left, and Right keys
	• **checkIsHitByEnemy** definition block
	   o change the **touching?** value to *TankShell1*
	   o change the **change** score **by** value to TankOneScore
	• **checkIsFiringShell** definition block
	   o change the **key pressed?** values to L
	   o change the **create clone of** to *TankShell2*

20	Click *TankShell2* sprite.
	Repeat step 18 but changing the following:
	
	• **when I start as a clone**
	   o change the **go to** block value to *Tank2*
	   o change the **point in direction** value to direction of *Tank2*
	   o change the last **touching** value to *Tank2*

21	That's it! You now have a 2-player tank fighting game! But what now? Well why don't you try improving it by: • Creating an introduction screen before the game starts and a screen explaining the controls • Allowing the players to change to different tank costumes • Have the tanks start randomly after they explode • Create a game over screen when a score is reached • More levels that change every time a tank is exploded • Or anything else you can think of!