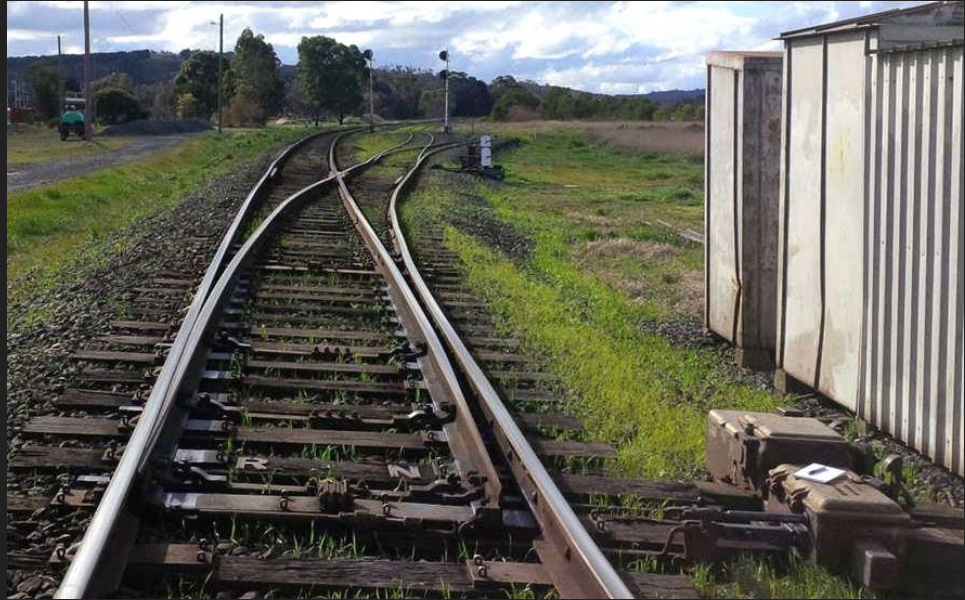# Variables & Conditions

Conditions

Conditions are like branches in a train track. They direct the flow of a program.

Action is made based on the truthfulness or falseness of the condition.


→ 🎮 Player | On collision with ➖ **Laser** | 🎮 Player | Spawn 🧨 **SmallExplosion** on layer **0** (image point *0*, create hierarchy: *False*)
| | | | + Add action

Variables

1. Variables store information for scripts to access and change
2. Have a name, eg: health
3. Have a type; number, boolean or string
4. Store only that type of data
5. Change variables in Actions

3

health

→ 🛩 Player	On collision with 🔸 **Laser**	🛩 Player	Subtract *1* from **health**
		🛩 Player	Spawn 🔥 **SmallExplosion** on layer **0** (image point *0*, create hierarchy: *False*)
		+ Add action	
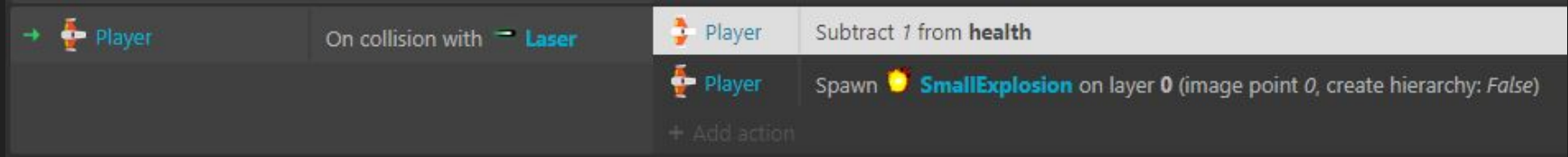
Variables & Conditions

Conditions can check variables for truthfulness and falseness too.

|			
🛩 Player	**health** ≤ 0	🛩 Player	Spawn ● **Explosion** on layer **0** (image point *0*, create hierarchy: *False*)
		🛩 Player	Destroy
		+ Add action	

Conditions can be chained together to check multiple conditions.

🛩 Player	Is **thrustOn**	+ Add action	
🔊 Audio	✕ Tag **"thruster"** is playing	🔊 Audio	Play **engine-thrust** looping at volume 0 dB (stereo pan *0*, tag *"thruster"*)
		🎇 Afterburner	Set visibility **Visible**
		+ Add action	

Challenge - Small Explosions on Enemy

When the Enemy is hit with a laser but not destroyed, make a smaller explosion and sound happen.

1. Find a smaller explosion effect and sound online and add them to the game.
2. Make them happen when a laser hits the enemy.

Challenge - Shield

Add a shield to the Player so that it can bump into enemies with a shield and explode when the shield is depleted:

1. Find a shield sprite online and add it to your game.
2. Make the Player bounce off enemies when the shield is equipped by adding an **Action->Player Reverse 8 Direction** to your code.
3. Give the shield health so that it depletes when it hits an enemy.
   Hint: See the **Enemy->On Collision with Laser** and **Enemy->health <= 0** blocks.