Collisions

This sheet explains how to detect whether sprites are touching one another, and do something as a result. For example, to destroy a sprite when a bullet hits it.

There are two main ways to check for collisions.

- use `isTouching()` for one-to-one and many-to-one checks. For example, if you have many enemies, and want to check if an enemy is touching the player.
- use `getTouching()` for many-to-many checks. For example, if you have many enemies and many bullets, and want to check if an enemy is touching any of the bullets.

Many-to-one

We can use `isTouching()` to check for a collision with exactly one other sprite. It returns a Boolean value (`true` or `false`), which we can use inside an `if` condition.

For example, we can check every frame if an enemy is touching a player, by adding a `forever` loop for the enemy.

```
enemy.forever(() => {
    if (enemy.isTouching(player)) {
        // do something
    }
})
```

Many-to-many

We can use `getTouching()` to get a list of all the other sprites which are overlapping with this one.

For example, we can check every frame if an enemy is touching any of the bullets, by adding a `forever` loop for the enemy.

Because it gives us all of the sprites touching this one, we need to check that they're the sort of sprites that we're interested in. Only bullets should kill enemies, for example; if the player touches the enemy that shouldn't also destroy the enemy.

```
enemy.forever(() => {
    for (let other of enemy.getTouching()) {
        if (other.costume == '💣') { // check it's a bullet
            // do something
        }
    }
})
```

Faster collisions

If your game is getting laggy, and you have a lot of sprites, you might find that it gets a lot faster by replacing `isTouching` and `getTouching` with their "fast" versions.

- `isTouching` → `isTouchingFast`
- `getTouching` → `getTouchingFast`

Normally we use Scratch-like pixel-perfect collision detection, which compares the images of the two sprites pixel-by-pixel. This is more accurate, but can be much slower. Using the "fast" versions is worth it if you can get away with it, particularly on older phones or slow computers.