



Image Animations

Upon completion of this module, a student will be able to

- use animated gifs in apps
- create an animated drawable object
- work with simple vectors in inkscape
- create animated vector drawables
- use animated vector drawables in apps



Assignment

- Task
 - For this project, you'll be creating a 3 page app. This can be done with 3 separate activities or a 3 tabbed window with fragments. In each tab, you'll display a different animated image.
- Repo
 - <https://github.com/LambdaSchool/AndroidAnimatedImages>
- Submission
 - Create a pull request on the main project repo and your PM will review it

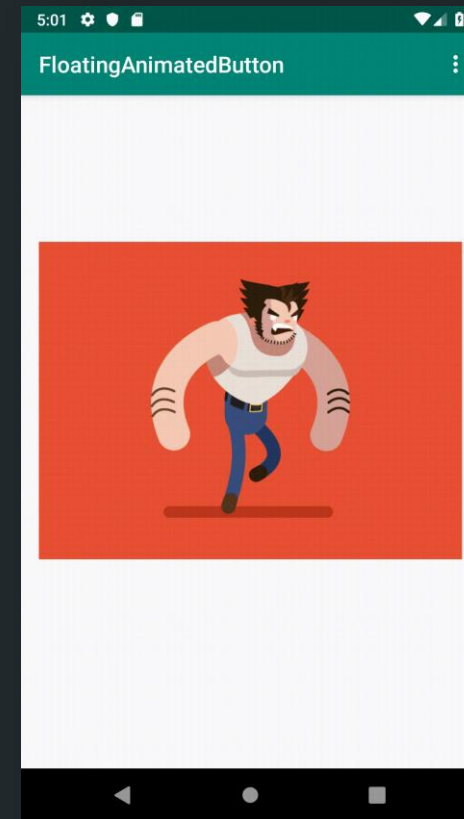




A Student Can
use animated gifs in apps

Animated Gifs

- Not supported natively before API 28 (Pie)
- AnimatedImageDrawable





A Student Can
create an animated drawable object

Animation Drawable

```
<?xml version="1.0" encoding="utf-8"?>
<animation-list xmlns:android="http://schemas.android.com/apk/res/android"
    android:oneshot="false">
    <item android:drawable="@drawable/wolverine000" android:duration="35" />
    ...
    <item android:drawable="@drawable/wolverine031" android:duration="35" />
</animation-list>
```

- AnimationDrawable
- Series of individual frames



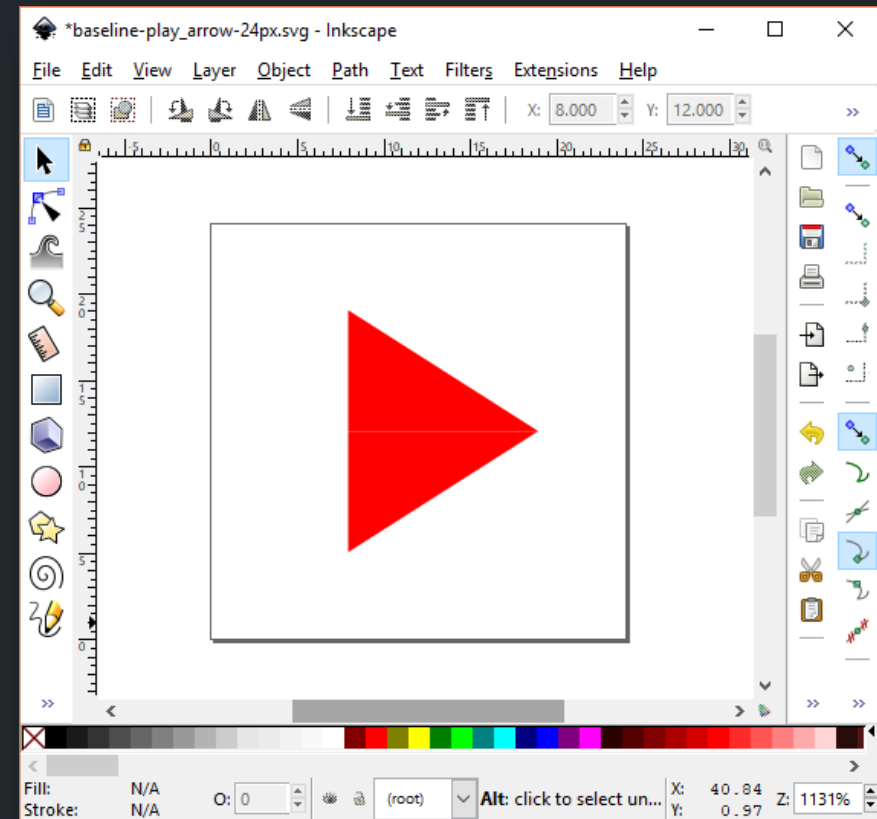


A Student Can

work with simple vectors in inkscape

Inkscape

- <https://romannurik.github.io/AndroidAssetStudio/index.html>
- Open source Vector Editing Tool



Path Manipulation

- Break Apart
 - Separate bodies grouped together
 - Path -> Break Apart
- Path Intersection
 - Create a new path which covered desired portion
 - Path -> Intersection

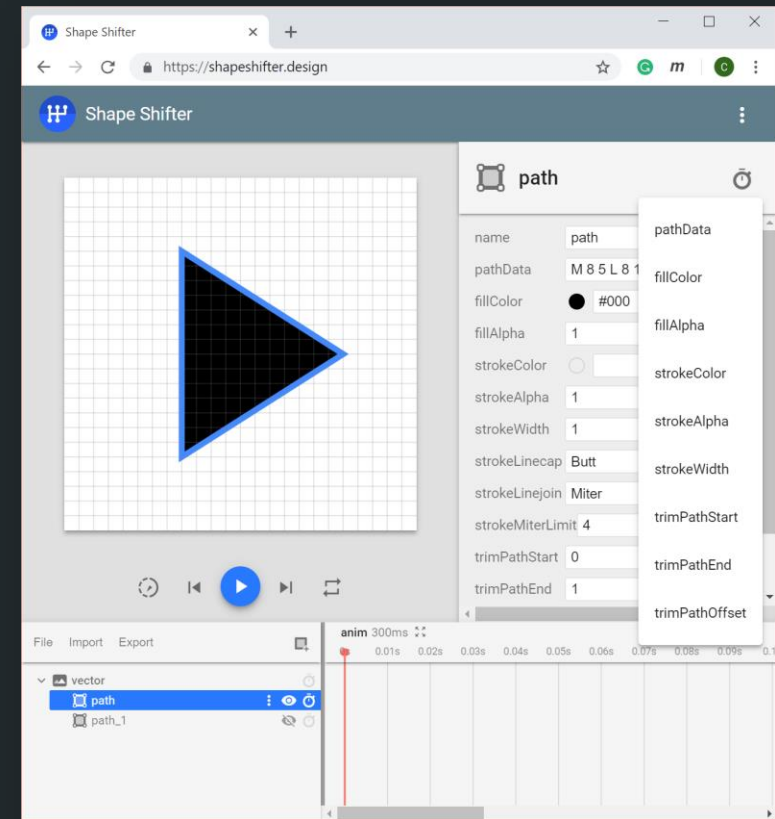




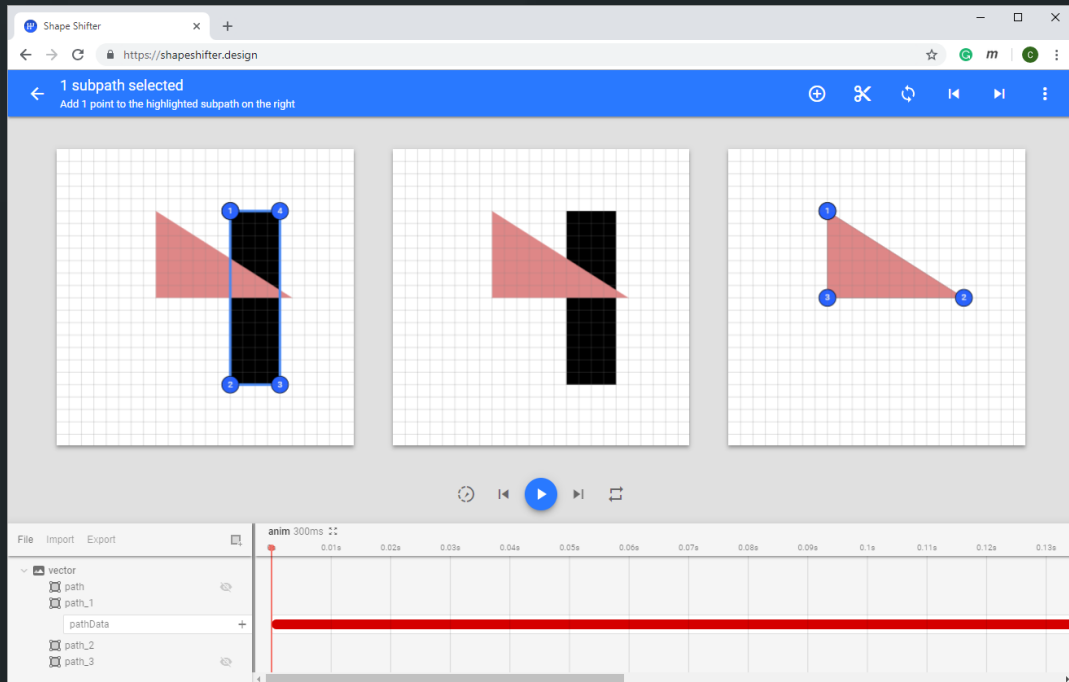
A Student Can
create animated vector drawables

Shape Shifter

- Use Existing Vector Images
- Edit and Animate
- Download as Animated Vector Drawable



Path Transform



- Transform PathData
- Past 2nd shape path into end path
- Track which points you want to go where and update them manually





A Student Can
use animated vector drawables in apps

In Code

```
imageButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        if(!playing) {  
            playing = true;  
            imageButton.setImageDrawable(getDrawable(R.drawable.avd_anim_play_pause));  
        } else {  
            playing = false;  
            imageButton.setImageDrawable(getDrawable(R.drawable.avd_anim_pause_play));  
        }  
  
        Animatable animatable = (Animatable) imageButton.getDrawable();  
        animatable.start();  
    }  
});
```

- Assign as normal drawable
- Animatable

