Sketch Nation Create Lesson Ideas:
*Teaching Computer Graphics*

Sketch Nation is a great way to teach the basics of computer graphics -- the way computers and mobile devices display images. In this tutorial, we explain some key aspects of game-related graphics, including pixels, palletization and animation, and suggest ways to engage students with these topics.

**Pixels**
Every image displayed by a computing device is made up of tiny squares called pixels. Today’s devices have such high resolutions that it’s often difficult to see the squares, but they are there! Old computers and gaming consoles had much lower resolutions and thus graphics with more visible blocks, usually referred to today as 8-bit graphics (because of how much memory they used).
8-bit graphics were usually 16 pixels wide by 16 pixels high because of memory limitations: 
$16 \times 16 = 256 = 2$ to the power of 8, meaning 8 bits of memory could be used to index a maximum of 256 pixels. Since each memory block was 8 bits wide, developers tended to use the entire 8 bits. If they created a sprite (the gaming term for an image composed of pixels), with fewer than 256 pixels, they would be wasting memory. If they used more than 256 pixels, they would need to use another memory block.

Today, devices have much higher resolution, allowing more detailed drawings. In fact, the resolution on new devices is so high that it can be almost impossible to see the individual pixels. Apple’s “retina display” technology actually refers to resolutions where the pixels are too small for the retina to discern.

![Mario](http://sketchnation.com)

**Palletization**
The 8-bit memory blocks also limited how many colors could be shown by each sprite. The display on a modern computer or mobile device can show more than 16.7 million colors, but in the past, due to memory limitations, engineers had to limit themselves to far fewer colors. Engineers developed different techniques to minimize the memory used by colors in applications, and one of these techniques was called “palletization”.

To store 16.7 million colors, each pixel would need 24 bits of memory ($2^{24}=16,777,216$) to tell the computer what color it is. That takes a lot of memory! With palletization, an artist, or the computer, picks a small subset of colors and stores them in a palette. Instead of each pixel referencing a color, it points to an index in the palette. By limiting the palette to 256 colors, each pixel now needs to use only 8 bits of memory instead of 24, a huge savings.
Below is a very simple example of how palletization is used:

Animation
Computers and mobile devices animate sprites by quickly switching the frames, or drawings, of the sprite. As an example, below are the "walking" frames for the Mario sprite in the original Super Mario Bros:

A game artist creates sprite animations by individually drawing each frame. The more frames an animation has, the smoother it appears. To keep things simple, Sketch Nation supports up to 10 frames in an animation.
1. Lesson Idea - Pixels

In Sketch Nation Create, have students create a new game -- any genre will work. At the bottom of the mode selection screen, they should slide the “Graphics style” slider all the way to the left and select 8-bit. This will set the resolution of each image to 16x16 pixels and will give them the “blocky” look associated with games like the original Super Mario Bros.

Now the students will be drawing and erasing the individual pixels of a sprite. As a follow-on, have students create a new game with a more modern, higher-resolution look by changing the “Graphics style” slider to “Low-res” (for low-resolution) or “HD” (for high-definition). This changes the number of pixels the user can draw for each object to 64x64 and 128x128 respectively.
2. Lesson Idea - Palletization

In Sketch Nation, we also limit the number of colors you can use in your drawings through palletization. While drawing, if you go to the color selection screen, you'll see that while there are many colors to choose from, there are far fewer than the 16.7 million your screen can display! This is the palette for drawing. We do this so that when you play and share games -- for example, sending a Sketch Nation game to a friend -- the game transfers, loads and plays as fast as possible.
While students are drawing on Sketch Nation, make them aware that they have to choose from a limited number of colors. You can tie this to an explanation of how this limited number of colors both helps Sketch Nation keep the size of games low, and how older, 8-bit style Nintendo games had to limit the number of colors in order to fit lots of sprites into their games.

Furthermore, when the user imports a photo, the palette will change to display the most common colors in the photo. As with drawings, this technique allows us to store the photo in a way that takes very little space and makes it easier to share games that include these photos with others. You can have the students draw their drawings on a piece of paper, then import them using the camera directly or after saving them to the photo library. Afterwards, ask students to look at the palette for their photo by going to the color selection screen, and describe what is different and why.

<table>
<thead>
<tr>
<th>Photo imported using the photo library</th>
<th>The palette for this photo</th>
</tr>
</thead>
</table>

3. Lesson Idea - Animation
Sketch Nation is a great way to learn how two-dimensional computer animation works. In Sketch Nation, the user can draw multiple frames for each object. The app will then automatically switch between the frames, creating an animation.

Here is a bird-flapping animation created with Sketch Nation:
And here is the combined animation:
Have your students experiment by creating animations for their players and enemies. They can start by creating a simple 2-frame animation, then add more frames to achieve a smoother animation.

For more advanced users, have them animate obstacles. This way they can see how their drawings affect the player. For example below we removed one frame of a platform, turning it into a more difficult obstacle that drops the player when the player stays on it too long (the absence of the obstacle makes the player fall).