This is a non-comprehensive collection of sample projects, mostly used by myself for testing, debugging and demonstrating Snap. The good ones have been sent to me by despairing users and did not work at the time because Snap was either broken or buggy, hence this list suffers from extreme and terrible selection bias, I’m sure that there are much better examples out there in the wild.

Awesome Projects from Users

The projects in this category define Snap’s “high end” and demonstrate its computational “no-ceiling” aspect. They were created by a BJC teacher and by amazing UC Berkeley CS10 students.

Space Invaders
(from a teacher in Colorado)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Space%20Invaders&editMode&noRun

PathFinder
(created by Aleks Kamko and Ben Purcell; UC Berkeley CS10, Fall 2012)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=PathFinder

Chess
(.created by UC Berkeley CS10 students)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Chess
Garden-Variety Projects

The projects in this category are a wild variety that don’t fit into a particular theme (below). Some are simple test projects for a certain feature, some have been transcribed from existing Scratch projects, and some have been shared by the BJC curriculum developers and from Brian and Dan.

**Snake**  
(playable game, stamping, using a list as queue)  
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Snake&editMode&noRun

**Bug on a Plate**  
(playable game by Bernatp from Hungary, pen-trails collision detection)  
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=BugOnAPlate&editMode&noRun

**Balloon Pop**  
(playable game by Cheddargirl from Scratch)  
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Balloon%20Pop

**Star Wars Game**  
(hilarious playable game from Scratch)  
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Star%20Wars%20Game
Funny Face
(by Hideki from Japan, off-center rotation point play, good starter project)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=funny%20face

Slideshow
(originally by Bernatp from Hungary - changing backgrounds, good starter project)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Slideshow

Constraint
(by Kazuhiro Abe from Japan)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=constraint

Portrait
(by Goch from Austria - sprite collision detection and stamping)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Portrait

Einmaleins
(in German: “AI-enhanced” multiplication trainer)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Einmaleins&editMode&noRun

Kaleidoscope
(awesome starter project from the BJC curriculum)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=kaleidoscope&editMode
3D Cube
(by Dan, wireframe animation)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=3D%20cube

Slices
(parallelization demo)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Slices&editMode&noRun

Jurassic
(simple stop motion animation by Luis, Jens’ son, example of a starter project)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Jurassic

Dog Walk
(keyboard-controlled sprite-animation with changing backgrounds, starter project)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=dogwalk&editMode

Path Follower
(simulates a very simple line-following robot using sprite-nesting)
http://snap.berkeley.edu/run#cloud:Username=jens&ProjectName=PathFollower
Themes

I have categorized the following projects into an arbitrary and by no means exhaustive set of “themes”, some of which used to be unique to Snap (recursion), or still are (HoFs, Nesting, SICP-style OOP).

Recursion

Double Jurassic Park
(fractal)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Double%20Jurassic%20Park

Hilbert Curve
(fractals)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Hilbert2

Tree Animation
(animated fractal)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=tree%20animation

Mystery Fractal
(by Dan, animation)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=2013Sp%20Final%20What%20Does%20This%20Do%3F&editMode
Count Change
(by Brian, SICP)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=count%20change&editMode

Tree
(/playable project by Brian, interactive fractal tree drawer)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Tree

Higher Order Functions

Vee
(Snap’s flagship project, idea by Paul, this design by Brian)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=vee&editMode&noRun

Roman Numerals
(MAP, REDUCE, DETECT)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Roman&editMode

Y Combinator
(by Brian back in the old BYOB3 days)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=y%20combinator&editMode
SICP-Style Dispatch-Procedure Objects

Joke
(modified Scratch example, Smalltalk-inspired ReadStream class with “self” dispatch)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=joke_self&editMode&noRun

Dolmetscher
(German - English - Swabian Interpreter, example scripts)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Dolmetscher&editMode

Music

Free Jacques
(round, coded explicitly)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Frere%20Jacques&editMode

Kanon
(playing a round from a voice recording)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Kanon&editMode&noRun

Menuet
(treating melodies/voices as lists)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Menuet&editMode&noRun
Round
(using clones to play a round)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Round&editMode&noRun

Sine Wave Blues Band
(AI-improvising)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Sine%20Wave%20Blues%20Band&editMode&noRun

Tiny Music
(atonal harmonic auto-composer)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Tiny%20Music&editMode&noRun

Clones

Sprudel
(“Sparkling Water” animation)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Sprudel&editMode&noRun

Growing Tree
(drawing a growing tree with clones)
snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Growing%20Tree&editMode
Exponential Growth
(a clone explosion)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Clones

Shooter
(playable game - use arrow keys to turn, space key to fire)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Shooter&editMode&noRun

Breakout (clones)
(self-playing game demo)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=breakout&editMode&noRun

Snow Globe
(Christmas card, flocking)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Snow%20Globe&editMode&noRun

Parable of the Polygons
after Vi Hart’s and Nicky Case’s famous blog post:
http://ncase.me/polygons/
This was a challenge from Alan Kay how we could facilitate authoring something like Vi and Nicky did in our respective programming environments. This was my first try in Snap:
(press green flag to set up the board, then drag the “unhappy” polygons around until they are happy. press space key to let the project auto-solve it.
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Parable%20(big)&editMode&noRun
Mesmerize
(Example for both clones and using JS block to draw custom costumes)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Mesmerize&editMode

Nested Sprites

Ferris Wheel
(nesting, with "dangling" connections)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Ferris%20wheel&editMode&noRun

Swimmer
(nesting, animation)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=swimmer&editMode

Copter
(an “Alan Kay” style driving car example, except it’s a quadcopter, and it doesn’t involve color collision detection, instead the rotors double as collision sensors)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=copter&editMode

Spirograph
(nesting, parts can draw individually):
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Spirograph
Planets
Eight Earth years are about equals to 13 Venus years, the two planets roughly trace out this pattern with 5-fold symmetry as they orbit around the Sun.

http://snap.berkeley.edu/snapsource/dev/snap.html#present:Username=jens&ProjectName=Earth%20Venus%20Orbits

Engine
(nesting, machine simulation):
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=engine&editMode&noRun

Zombies
(demonstrates how nested sprites can even be clones as whole)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Zombie%20%23ScratchMIT2014&editMode&noRun

Media Comp
Pixel manipulation, graphic effects. This experimental project came out of collaborating with Mark Guzdial on GP and other things, it’s also referenced in my description of the “table” feature:
http://snap.berkeley.edu/snapsource.TablesInSnap.pdf
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=MediaComp&editMode
(note: you can seriously speed up the examples in this project, especially for the high-res version of the photograph, by replacing “map” with “fastMap” as described in the pdf).

Pen-trails as Costumes
and then stamping them:
1. click green flag
2. press “s”
3. press “p”
etc…
these blocks are now an “official” library
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=save%20and%20restore%20pen%20trails&editMode&noRun
Fastest Fractal
(by Dan, uses the pen-trails saving and restoring blocks for linear-time fractals)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=save%20and%20restore%20pen%20trails%20FRACTAL&editMode

First Class Sprites

Towers of Hanoi
(demonstrates sprites in lists)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Towers%20of%20Hanoi

Population Simulation
(demonstrates collision detection among clones)
http://snap.berkeley.edu/run#cloud:Username=jens&ProjectName=population

Woodworm
(demonstrates enumerating many clones)
http://snap.berkeley.edu/snapsource/snap.html#cloud:Username=jens&ProjectName=Woodworm

Ferris Wheel 2016
(demonstrates “bootstrapping” a machine by drawing costumes for clones)
http://snap.berkeley.edu/run#present:Username=jens&ProjectName=Ferris%20Wheel%202016
Cartwheel
(demonstrates programmatic sprite-nesting)
http://snap.berkeley.edu/run#present:Username=jens&ProjectName=cartwheel

Ants and Eggs
(idea and prototype by Xavier Leroy)
http://snap.berkeley.edu/snapsource/
snap.html#cloud:Username=jens&ProjectName=Ants%20and%20Eggs

Geeky Features

Codification
(transpiling blocks to arbitrary text languages)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Codification&editMode

Continuations
(test-suite)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=continuations&editMode

Upvars
and variable scope
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=Upvars&editMode

JS-Functions
extending Snap with JavaScript
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=JSFunctions&editMode

Swapping Variable Values
(a challenge from our friends at the Citilab in Barcelona)
http://snap.berkeley.edu/snapsource/
snap.html#present:Username=jens&ProjectName=swap%20variables&editMode&noRun
Free Form Expressions
(type-in JS-functions instead of nesting reporter blocks)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=free%20form%20expressions&noRun

Tables
(test-cases, enable “table support” for these)
(see: http://snap.berkeley.edu/snapsource/TablesInSnap.pdf)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=table%20docu&editMode

Linked Lists
see:
http://snap.berkeley.edu/snapsource/LinkedListsInSnap.pdf
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=linked%20list%20identity&editMode&noRun

Tail-Call Elimination
(Proper Tail Recursion)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=tco&editMode&noRun

Mutable Formal Parameters
(Ethiopian Multiplication Example stub)
http://snap.berkeley.edu/snapsource/snap.html#present:Username=jens&ProjectName=Ethiopian%20Multiplication&editMode