



Topic	Git and Github	
Class Description	Students will learn how to use different git commands to set up a repository, clone a repository, commit changes etc. remotely through the command line.	
Class	C26	
Class time	45 mins	
Goal	<ul style="list-style-type: none"> • Install git and command line git bash. • Use git commands to manage, work and update a remote repository. 	
Resources Required	<ul style="list-style-type: none"> • Teacher Resources <ul style="list-style-type: none"> ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen • Student Resources <ul style="list-style-type: none"> ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen 	
Class structure	Warm Up Teacher-led Activity Student-led Activity Wrap up	5 mins 15 min 15 mins 5 min
<p style="text-align: center;"><u>CONTEXT</u></p> <ul style="list-style-type: none"> • Review the concepts from the previous class. • The pain of manually downloading, unzipping, updating a github code base. 		
Class Steps	Teacher Action	Student Action
Step 1: Warm Up (5 mins)	Welcome back to the class. Do you remember what we did in the last class?	ESR: Yes! - We added images and animation to all the game objects.

		<p>- We learned about class inheritance and how to create new classes which extend from a parent class and inherit all the properties and functions of the parent class.</p>
	<p>Before that I have an exciting quiz for you! Are you ready?</p> <p>Teacher click on the</p> <div data-bbox="467 758 690 842">  </div> <p>button on the bottom right corner of your screen to start the In-Class Quiz.</p> <p>A quiz will be visible to both you and the student.</p> <p>Encourage the student to answer the quiz question.</p> <p>The student may choose the wrong option, help the student to think correctly about the question and then answer again.</p> <p>After the student selects the correct option, the</p> <div data-bbox="613 1524 849 1587">  </div> <p>button will start appearing on your screen.</p> <p>Click the End quiz to close the quiz pop-up and continue the class.</p>	<p>ESR: Yes</p>

	<p>Good.</p> <p>Now, we must create a slingshot with which we can shoot our pigs. The current way of using mouse control for the bird is too easy.</p>	<i>Student listens.</i>
	<p>But before we start with creating a slingshot, don't you think we should do something about having to repeatedly go on the Github link, downloading the zipped files, unzipping them and then starting on our projects?</p> <p>This is a pain!</p> <p>Remember the golden saying - DO NOT REPEAT YOURSELF.</p> <p>Today, we will learn how to do all these things and more in an easy and fun way where we don't have to go through all this pain.</p> <p>Let's get started.</p>	-
Teacher Initiates Screen Share		
<p style="text-align: center;"><u>CHALLENGE</u></p> <ul style="list-style-type: none"> ● Install command line git and git bash. ● Clone the activity repository. ● Make some changes to the code and commit these changes to a remote repository. 		
Step 2: Teacher-led Activity (15 min)	<p>We have been using GitHub website to host our code online.</p>	

	<p>GitHub uses a tool called Git which is used to keep track of all the changes in the code. It is a very powerful tool for developers to keep track of the changes they are making to their code and also keeping the track of the changes made while collaborating with others.</p> <p>We will learn more about Git in today's class. But before that, we need to install Git into our system.</p> <p><i>Guide the student to install Git into their system. Make sure they check the checkbox to install the git bash shell for Windows.</i></p> <p><i>Also guide them to create a GitHub account if they didn't create it in the previous class. Get them to remember the GitHub credentials (username and password).</i></p>	<p><i>The student installs Git on their system.</i></p>
<p>For Mac (follow the instructions given in the PDF): https://whitehatjrcontent.s3.ap-south-1.amazonaws.com/curriculum/PRO+Asset/git_installation_MAC.pdf</p> <p>after installing Git to check the git version and set git id and password.</p> <p>note-(git version may differ)</p>		

. Open a terminal and verify the installation was successful by typing `git --version`:

```
$ git --version  
git version 2.9.2
```

. Configure your Git username and email using the following commands, replacing Emma's name with your own. These details will be associated with any commits that you create:

```
$ git config --global user.name "Emma Paris"  
$ git config --global user.email "eparis@atlassian.co
```

For Windows:

<https://gitforwindows.org/>

REMEMBER TO CHECK THE INSTALL GIT BASH CHECKBOX.

Git for Windows stand-alone installer

1. Download the latest [Git for Windows installer](#).
2. When you've successfully started the installer, you should see the **Git Setup** wizard screen. Follow the **Next** and **Finish** prompts to complete the installation. The default options are pretty sensible for most users.
3. Open a Command Prompt (or Git Bash if during installation you elected not to use Git from the Windows Command Prompt).
4. Run the following commands to configure your Git username and email using the following commands, replacing Emma's name with your own. These details will be associated with any commits that you create:

```
$ git config --global user.name "Emma Paris"
$ git config --global user.email "eparis@atlassia.com"
```

Open your git bash shell (for Windows) or a normal terminal (for Mac).

You know every computer uses a hard drive which is its memory. A hard drive is divided into sections and each section has a name. Each section can be divided into more sections called folders. Each folder may contain a file or another folder.

This is a very powerful window. You

	<p>can do a lot of things from here.</p> <p>Let's check which section of the computer's memory we are in right now.</p> <p>We do this by typing 'pwd'. pwd stands for present working directory.</p> <p><i>Teacher enters pwd on the bash shell and shows the output.</i></p>	<p><i>Student experiments with pwd in their own bash shell.</i></p>
<pre>\$ pwd /home/rajeev \$</pre>		
	<p>You can see all the files and folders that are present in this location right here in the window.</p> <p>You need to type the command 'ls' to list down all the files in this location.</p>	<p><i>Student experiments with 'ls' in their own bash shell.</i></p>

```
Welcome to fish, the friendly interactive shell
$ pwd
/home/rajeev
$ ls
1
aadhar.pdf
ad.png
Android/
android-studio/
AndroidStudioProjects/
angrybirds/
animation/
anyDeskAutomation
Backup/
Bhagwati/
bkscToSec.pdf
blue.wav
"Breakout - Lecture 2 - CS50's Introduction to Ga
"Breakout - Lecture 2 - CS50's Introduction to Ga
C-1
'Calibre Library'/
CA.pdf
chessboard.jpg
CompanyPresentation.pdf
conjecture_map
course2.wav
CourseraFinancialAid
course.wav
'CS50 2017 - Lecture 1 - C-3K4jWlpR4iY.f313.webm
'CS50 2017 - Lecture 1 - C-3K4jWlpR4iY.mp4'
C-Trial
curriculumDesigner2.png
curriculumDesigner.png
daaoa.pdf
datasciencecoursera/
DBR.pdf
Desktop/
Downloads/
dustbin_closed.png
dustbin_open.png
emailScript
Empsurvey.txt
expGround.png
falling.png
firebase-codelab/
flappy_assets/
flappyBall/
flappyBall2.tar
```

Amazing! isn't it?

You can in fact navigate to any other folder or directory and see the files/folders there.

We use 'cd' (change directory) command to change the terminal's location to any other directory location. You need to enter the complete address of the location.

Student experiments with the 'cd' and 'ls' commands in their own shell.

	<p>You need to type <code>cd</code> <code><SPACE><location of the new directory></code></p> <p>You can then use <code>'ls'</code> to list down the contents of that directory.</p>	
<pre>\$ cd ~ \$ cd /home/rajeev/ProjectAlgorithmsDesign/ \$ ls index.html p5.dom.min.js p5.js p5.play.js p5.sound.min.js sketch.js style.css \$</pre>		
	<p>We can also create a new directory/folder in any location using <code>'mkdir'</code> commands.</p> <p><code>'mkdir'</code> stands for make directory.</p> <p>You need to type <code>mkdir<SPACE><Name and location of the new directory></code></p> <p>If only the name of the directory is given, the directory is created in the default location.</p> <p>Use <code>'mkdir'</code> and then <code>'ls'</code> to show the directory created.</p> <p><i>Get the student to create a Projects Directory.</i></p>	<p><i>Student experiments with the 'mkdir' and 'ls' command in their shell.</i></p> <p><i>Student creates a directory called Projects inside his home directory.</i></p>
mkdir Projects/		
	<p>All this is cool!</p> <p>We can do much more but let's first learn to use git.</p>	

	<p>We had progressed to Angry Birds stage 2 in the last class.</p> <p>There is a GitHub link in your student Activity.</p>	<p><i>Student checks the Github Link.</i></p>
	<p>Earlier, we used to download the files in a zipped format, then unzipped and opened them in the visual code studio. Now we don't need to do that. We can simply clone the Github repository using the "git clone" command.</p> <p><i>Teacher shows how to git clone the repository. [Teacher Activity 1]</i></p> <p><i>Teacher uses 'ls' to show the cloned repository into the local system.</i></p>	<p><i>Student observes and learns.</i></p>

```

rocksvishu@Vishals-MacBook-Pro ~ % mkdir Project
rocksvishu@Vishals-MacBook-Pro ~ % cd Project
rocksvishu@Vishals-MacBook-Pro Project % git clone https://github.com/whitehatjr/angryBirdsStage2.git
Cloning into 'angryBirdsStage2'...
remote: Enumerating objects: 38, done.
remote: Total 38 (delta 0), reused 0 (delta 0), pack-reused 38
Unpacking objects: 100% (38/38), done.
rocksvishu@Vishals-MacBook-Pro Project % ls
angryBirdsStage2
rocksvishu@Vishals-MacBook-Pro Project % cd angryBirdsStage2
rocksvishu@Vishals-MacBook-Pro angryBirdsStage2 % ls
BaseClass.js    Ground.js      README.md      p5.dom.min.js  sketch.js
Bird.js         Log.js        index.html     p5.min.js      sprites
Box.js          Pig.js        matter.js      p5.sound.min.js style.css
rocksvishu@Vishals-MacBook-Pro angryBirdsStage2 %

```

```

$ mkdir Projects
$ cd Projects/
$ git clone https://github.com/whitehatjr/angryBirdsStage2
Cloning into 'angryBirdsStage2'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 38 (delta 10), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (38/38), done.
$ ls
angryBirdsStage2/
$ cd angryBirdsStage2/
$ ls
BaseClass.js  Box.js      index.html  matter.js   p5.min.js   Pig.js      sketch.js   style.css
Bird.js       Ground.js  Log.js      p5.dom.min.js  p5.sound.min.js  README.md  sprites/
$
  
```

Isn't this nice and amazing! Half of our pain is gone!

But we can do more. We can actually make some changes to our file.

Let's create a big platform for our slingshot. We will do this by creating a new Ground object using the Ground class in our sketch.js file.

Teacher writes code to create a platform for the slingshot.

Student observes.

```
1 sketch.js
const Engine = Matter.Engine;
const World= Matter.World;
const Bodies = Matter.Bodies;

var engine, world;
var box1, pig1;
var backgroundImg,platform;

function preload() {
  backgroundImg = loadImage("sprites/bg.png");
}

function setup(){
  var canvas = createCanvas(1200,400);
  engine = Engine.create();
  world = engine.world;

  ground = new Ground(600,height,1200,20);
  platform = new Ground(150, 305, 300, 170);

  box1 = new Box(700,320,70,70);
  box2 = new Box(920,320,70,70);
  pig1 = new Pig(810, 350);
  log1 = new Log(810,260,300, PI/2);

  box3 = new Box(700,240,70,70);
  box4 = new Box(920,240,70,70);
  pig3 = new Pig(810, 220);

  log3 = new Log(810,180,300, PI/2);

  box5 = new Box(810,160,70,70);
  log4 = new Log(760,120,150, PI/7);
  log5 = new Log(870,120,150, -PI/7);

  bird = new Bird(100,100);
}

function draw(){
  background(backgroundImg);
  Engine.update(engine);
  console.log(box2.body.position.x);
  console.log(box2.body.position.y);
}
```

```

platform = new Ground(150, 305, 300, 170);

box1 = new Box(700,320,70,70);
box2 = new Box(920,320,70,70);
pig1 = new Pig(810, 350);
log1 = new Log(810,260,300, PI/2);

box3 = new Box(700,240,70,70);
box4 = new Box(920,240,70,70);
pig3 = new Pig(810, 220);

log3 = new Log(810,190,300, PI/2);

box5 = new Box(810,160,70,70);
log4 = new Log(760,120,150, PI/7);
log5 = new Log(870,120,150, -PI/7);

bird = new Bird(100,100);
}

function draw(){
  background(backgroundImg);
  Engine.update(engine);
  console.log(box2.body.position.x);
  console.log(box2.body.position.y);
  console.log(box2.body.angle);
  box1.display();
  box2.display();
  ground.display();
  pig1.display();
  log1.display();

  box3.display();
  box4.display();
  pig3.display();
  log3.display();

  box5.display();
  log4.display();
  log5.display();

  bird.display();
  platform.display();
}

```

We have the code ready for the platform.

git knows that we have made some changes. You can use 'git status' to see that git knows that there have been some changes made.

Student observes and asks questions.

git status

One needs to get familiar with the following three areas while working with git:

- there is a working directory.
- there is a staging area.
- there is a repository (local and

Student observes and asks questions.

	<p>remote).</p> <p>Files are first modified in the working directory - just like we did right now when we modified the sketch.js file.</p> <p>Staging area hosts all the files which are changed before they are committed.</p> <p>A repository hosts all the different committed versions of the files.</p> <p>Every modified file travels from working directory to staging area to a repository.</p> <p>We need to add our sketch.js file to the staging area and then commit these changes (just like we did on Github after making changes to a file).</p> <p>Every commit is written with a commit message - so that later you can see the changes you made to the files.</p> <p><i>Teacher shows how to make a git add and git commit the files with a message.</i></p> <pre>git add <file_name> git commit -m "<message>"</pre> <p>(-m stands for message)</p>	
<pre>\$ git add sketch.js \$ git commit -m "Add Platform" [master 02d825a] Add Platform 1 file changed, 5 insertions(+), 3 deletions(-) \$</pre>		

	<p>Now that we have made some changes into our code, we can push this code directly to our remote github repository.</p> <p>For this, we have to create an empty remote Github Repository.</p> <p>Create an empty remote GitHub repository.</p> <p>Note: DO NOT INCLUDE A README.md file.</p>	<p><i>Student observes and asks questions.</i></p>
--	--	--

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner

 whitehatjr ▾

Repository name *

AngryBirdsStage2.5StudentActivit ✓

Great repository names are short and memorable. Need inspiration? How about **cautious-octo-fortnight?**

Description (optional)

Student activity for Angry Birds stage 2.5



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▾

Add a license: **None** ▾



whitehatjr / AngryBirdsStage2.5StudentActivity

Watch

0

Star

0

Fork

0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Security

Insights

Settings

Quick setup — if you've done this kind of thing before

or

HTTPS

SSH

https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

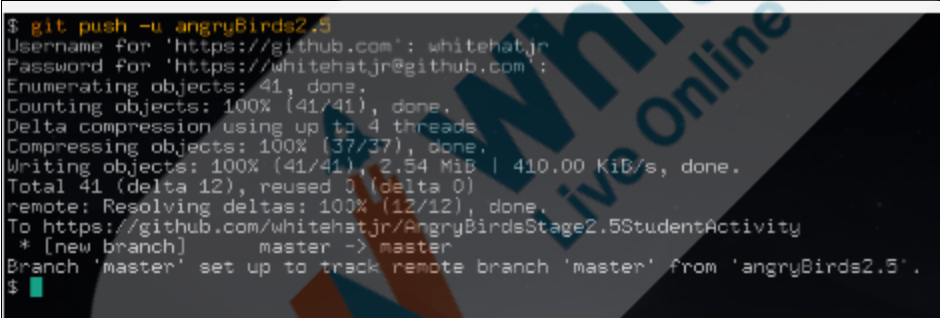
```

echo "# AngryBirdsStage2.5StudentActivity" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity.git
git push -u origin master

```

...or push an existing repository from the command line

```
$ git remote add test https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
```


	<p>Finally we have the committed changes stored in our local repository.</p> <p>We want to push these files and changes to our remote repository we just created.</p> <p><i>Teacher shows how to push files to a remote repository.</i></p> <p><code>git push -u <remote_name></code></p> <p>(-u stands for username which is promoted by this command)</p> <p>Note: A username is needed so that the codebase gets uploaded in your relevant account.</p>	<p><i>Student observes, learns and asks questions.</i></p>
	 <pre> \$ git push -u angryBirds2.5 Username for 'https://github.com': whitehatjr Password for 'https://whitehatjr@github.com': Enumerating objects: 41, done. Counting objects: 100% (41/41), done. Delta compression using up to 4 threads Compressing objects: 100% (37/37), done. Writing objects: 100% (41/41), 2.54 MiB 410.00 KiB/s, done. Total 41 (delta 12), reused 0 (delta 0) remote: Resolving deltas: 100% (12/12), done. To https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity * [new branch] master -> master Branch 'master' set up to track remote branch 'master' from 'angryBirds2.5'. \$ </pre>	
	<p>Check the remote repository url now by visiting github.com. Login and you will see the files have been updated there.</p> <p><i>Teacher shows the updated remote url.</i></p> <p>You can see all the commits ever made on this repository by typing the</p>	<p>-</p>

command "git log".

Teacher shows how to use the git log command.

In fact, you can go back to any of the previous versions of the code anytime - but we will see that later.

Student activity for Angry Birds stage 2.5 Edit

Manage topics

7 commits 1 branch 0 releases 2 contributors

Branch: master New pull request Create new file Upload files Find File Clone or download

rajeev-artha Add Platform Latest commit 62d825a 7 minutes ago

File	Action	Time
sprites	Delete test	2 days ago
BaseClass.js	Add files via upload	2 days ago
Bird.js	Add files via upload	2 days ago
Box.js	Add files via upload	2 days ago
Ground.js	Add files via upload	2 days ago
Log.js	Add files via upload	2 days ago
Pig.js	Add files via upload	2 days ago
README.md	Initial commit	2 days ago
index.html	Add files via upload	2 days ago

```
$ git log
commit a2e7acbe0692ceed25264239ce2c19fa98139067 (HEAD -> master, angryBirds2.5/master)
Author: rajeev.artha <rajeev.artha@gmail.com>
Date: Sun Jul 14 03:47:18 2019 +0530

    Changed README

commit 02d825a57be6d990130f85693763c50116baa1e9
Author: rajeev.artha <rajeev.artha@gmail.com>
Date: Sun Jul 14 03:18:29 2019 +0530

    Add Platform

commit 7510a4e70f02d82653bbf1a3b36ba33acf1e705e (origin/master, origin/HEAD)
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Fri Jul 12 00:15:23 2019 +0530

    Update sketch.js

commit 8e6a9e2765d781619dead3a302de654065884858
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Fri Jul 12 00:10:49 2019 +0530

    Delete test

commit ebfc4fd674a96f3786287eea16a857752a02350b
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Thu Jul 11 23:59:54 2019 +0530

    Add files via upload

commit e3e87fc8e423b55a535ee5932023ecacc1dd5cc4
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Thu Jul 11 23:59:04 2019 +0530

    Create test mod+Enter
    for focused mod+Shift+e

commit dad39f8e944eb5e923b20557f0a414461d185243
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Thu Jul 11 23:58:32 2019 +0530

    mod+mod+mod+
    Add files via upload
    send to workspace 1-8 - mod+Ctrl+1-8

commit ab53b027c3934524c550e9a3d704ce662e9b8d7d5
Author: whitehatjr <52084703+whitehatjr@users.noreply.github.com>
Date: Thu Jul 11 23:57:11 2019 +0530

    mod+mod+shift+
    Initial commit
```

There is some learning involved to do this but isn't this interesting?

Why don't you do this on your own now. I can assist you.

ESR:
Yes!

Teacher Stops Screen Share

Now it's your turn. Please share your screen with me.

- Ask Student to press ESC key to come back to panel
- Guide Student to start Screen Share
- Teacher gets into Fullscreen

ACTIVITY

- Clone a remote repository.
- Add the code to create a platform for the bird.
- Check the status of the new repository.
- Create a remote repository.
- Commit the change to a remote repository.

Step 3: Student-Led Activity (15 mins)

Guide the student to clone the code from the remote repository ([Student Activity 1](#)).

Student uses git clone to clone the code from the remote repo to the local working directory.

```
$ git clone https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
```

Guide the student to navigate to this directory.

Student uses "cd" to move to this directory.

```
$ git clone https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
Cloning into 'AngryBirdsStage2.5StudentActivity'...
remote: Enumerating objects: 44, done.
remote: Counting objects: 100% (44/44), done.
remote: Compressing objects: 100% (27/27), done.
remote: Total 44 (delta 13), reused 44 (delta 13), pack-reused 0
Unpacking objects: 100% (44/44), done.
$ cd AngryBirdsStage2.5StudentActivity/
$
```

Guide the student to modify the Readme.md file to add more details.

Student makes some changes to the Readme file.

```
1 README.md
2 angryBirdsStage2.5
3 -Angry Birds stage 2.5 with Class Inheritance and Images
4
5
6
```

Guide the student to add the files to the staging area.

The student uses 'git add' to add the files to the staging area.




<pre>\$ git add README.md \$ git status On branch master Your branch is up to date with 'origin/master'. Changes to be committed: (use "git reset HEAD <file>..." to unstage) modified: README.md Changes not staged for commit: (use "git add <file>..." to update what will be committed) (use "git checkout -- <file>..." to discard changes in working directory) modified: Ground.js</pre>		
	<p><i>Guide the student to add it to the local repository with a commit message.</i></p>	<p><i>Student writes a commit message and adds the file to the local repository.</i></p>
<pre>\$ git commit -m "Changed Readme" [master 90b2e56] Changed Readme 1 file changed, 1 insertion(+), 1 deletion(-)</pre>		
	<p><i>Guide the student to create a remote Repository on Github without a ReadMe file.</i></p>	<p><i>Student creates a remote repository on github and copies the link.</i></p>
	<p><i>Guide the student to add the remote repository to the working directory with a remote name and URL.</i></p>	<p><i>Student adds the remote repository to the local repository.</i></p>
<p><Use the repository link here></p> <pre>\$ git commit -m "Changed Readme" [master 90b2e56] Changed Readme 1 file changed, 1 insertion(+), 1 deletion(-) \$ git remote add angryBirds2.5 https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity</pre>		
	<p><i>Guide the student to push the changes to the remote repository.</i></p> <p><i>Students can see git status to check the status of the staging area.</i></p> <p><i>Students can see git log to check the log of commits to the repository.</i></p>	<p><i>Student pushes the changes from the local repository to the remote repository.</i></p>

```
$ git push -u angryBirds2.5
```

Teacher Guides Student to Stop Screen Share

FEEDBACK

- Encourage the student to make reflection notes in markdown format.
- Complement the student for her/his effort in the class.
- Review the content of the lesson.

Step 4: Wrap-Up (5 min)	How was the class today? How are you feeling?	ESR: varied
	Can you try to summarize what we learned today?	ESR: <ul style="list-style-type: none"> - We learned the use of git commands to <ul style="list-style-type: none"> - clone a repository. - stage and commit changes in a file in a repository. - push the changes to a remote repository.
	<p>Git is exciting and very powerful.</p> <p>Developers use git to collaborate on their work. In fact, two developers can often work on the same code base using git.</p> <p>Git promotes collaboration.</p> <p>We will keep learning about the more powerful features of git as we progress through the lessons.</p>	<p><i>Make sure you have given at least 2 Hats Off during the class for:</i></p> <div>Creatively Solved Activities  +10</div> <div>Great Question  +10</div> <div>Strong Concentration  +10</div>

	<p>You get Hats Off for your excellent work!</p> <p>Next class we will start working on the slingshot.</p>	
<p><u>Project Name:</u> <u>Masterchef</u> <u>Junior</u></p>	<p>* This Project will take only 30 mins to complete. Motivate students to try and finish it immediately after the class.</p> <p>Goal of the Project:</p> <p>In this class you have learned how to use different git commands to set up a repository, clone a repository, commit changes remotely through the command line (terminal).</p> <p>In this project you will apply what you have learned by publishing your most favorite recipe online!</p> <p>Story:</p> <p>Joshua is a world famous chef! And he wants you to do the task of taking his most favorite recipe and publishing it online for the world to see.</p> <p>I am very excited to see your project solution and I know you will do really well.</p> <p>Bye Bye!</p>	<p>Note: You can assign the project to the student in class itself by clicking on the Assign Project button which is available under the projects tab.</p>

<div style="text-align: center;"> Teacher Clicks ✕ End Class </div>		
Additional Activities	<p><i>Encourage the student to write reflection notes in their reflection journal using markdown.</i></p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> • What happened today? <ul style="list-style-type: none"> - Describe what happened - Code I wrote • How did I feel after the class? • What have I learned about programming and developing games? • What aspects of the class helped me? What did I find difficult? 	<p><i>Student uses the markdown editor to write her/his reflection as a reflection journal.</i></p>

Activity	Activity Name	Links
Teacher Activity 1	Angry Birds Stage 2 Github link	https://github.com/whitehatjr/angryBirdsStage2
Student Activity 1	Angry Birds Stage 2.5 Github link	https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
Project Solution	Masterchef Junior	As it is an open-ended project, there is no specific solution