```
Lecture Notes for Week 2
========================

+ Git

1. What is a Version Control System and why is it useful?

Software that allows for numerous people to work on files at the same time and save
versions of the files.
The files exist on a server as a master repository, and allows
the keeping of file history and changes.


2. What do each of the following git commands do?

git clone <url>: Makes a copy of the online repository in your own computer
git pull: gets the possibly updated files from a given online repository and integrates it with
the local repository
git add: adds files to next commit (stage files)
git commit -m "Replace this with a message": saves all the added files in the local repository
as a "snapshot" of the current state of files
git push: sends the commits back to the original repository


+ Java

3. From last week's notes: what are the following: "constructor", "method", "class variable",
primitive types, reference types.

see last week's notes

(If you already answered some of these, just say "see last week's notes")

4. What are "autoboxing", "unboxing", "Wrapper" class?
autoboxing: taking a primitive variant of the class and automatically calls the constructor
using the primitive value
unboxing: taking values from inside of objects
Wrapper: A class that takes a primitive variable and "wraps" it into a object

5. When are the equals method and the == operator the same? When are they different?

For primitives, equals and == are the same as they both compare the data in the variable
For reference variables, == compare the memory addresses while equals is a method that can be
changed to compare attributes (or other things)

6. What do the accessibility modifiers mean?
+ public
Can be accessed from anywhere
+ private
Can only be accessed in the class
+ protected
Can be accessed anywhere in the package and in its subclasses
+ package protected
Can be accessed anywhere in the package


7. What is the difference between "static" and "instance"?

static variables are created independant of object instances
instance variables are made when objects and made and belong to that object

8. When do we use the keyword "this"?

When initializaing the instance variables in the constructor
```

9. What is the "default" constructor and when does a class get one?

If super is not called in the constructor, the constructor in the parent is called with no
arguments

10. What is the difference between overriding and overloading?

overloading: creating numerous versions of a method of the same name with different parameters
(arguments)
overriding: changing a method with the same name and parameters from its parent class

11. What is the Java lookup rule for:
instance methods: Start at the bottom of the class definition and searches upwards until theres
nothing more, ending up in an error
instance variables: Start by searching all expressions of the given type and then upwards until
nothing more, ending up in an error
static methods: Start at the bottom of the class definition and searches upwards until theres
nothing more, ending up in an error
static variables:

12. How does inheritance work in Java? How is it connected to "the super constructor"?

Classes can extend other classes to inherit their methods and data
Constructors typically start with super(args), calling the constructor in the parent class

13. When do the Java lookup rules apply?

When calling a method or instance variable.

14. What is casting? Why do we try not to use "()" to do casting?

Force Java to treat a variable as a different data type that what it was when defined, casting
may want to not be used as to keep the code as closed/encapsulated as possible

15. What does it mean for Class A to be dependent on Class B?

When class A and B have high coupling.