Joshua Bilsland

The Odin Project Notes

A collection of notes

Contents

[Foundations – Introduction 2](#_Toc109910412)

[How This Course Will Work 2](#_Toc109910413)

[Introduction to Web Development 2](#_Toc109910414)

[Motivation and Mindset 3](#_Toc109910415)

[Asking For Help 3](#_Toc109910416)

[Join the Odin Community 4](#_Toc109910417)

[Foundations – Prerequisites 4](#_Toc109910418)

[Computer Basics 4](#_Toc109910419)

[How Does the Web Work? 5](#_Toc109910420)

[Installation Overview 6](#_Toc109910421)

[Installations 6](#_Toc109910422)

[Text Editors 6](#_Toc109910423)

[Command Line Basics 6](#_Toc109910424)

[Setting Up Git 7](#_Toc109910425)

[Foundations – Introduction to Git 7](#_Toc109910426)

# Foundations – Introduction

## How This Course Will Work

* “The Odin Project is an open-source community dedicated to providing the best information sources to take you from zero to a full-stack developer”
* Will go over the basics of internet, Git, GitHub, HTML, CSS, JS, back-end technologies, etc
* Lessons will contain questions that should be answered before moving on
* After the foundations course, the Full Stack JavaScript and Full Stack Rails paths can be taken
* A collection of the best sources that could be found by the community
* Some things made by the community themselves

## Introduction to Web Development

* Web development is project-focused and involves collaborating with a team that helps to co-ordinate the client’s needs into the end product.
* Front end:
  + What the website visitors see
  + The presentation of content and user interface elements
  + Use of HTML, CSS, and JavaScript
* Back end:
  + “The guts of the application”
  + Stores and serves data to ensure the front end has what it needs
  + Uses languages such as Java, Python, and Ruby
* Full Stack:
  + Developers that are comfortable working with both the front and back ends
  + The Odin Project focuses on teaching full-stack development
* Can work for large tech companies, startups, as a freelancer, or as a consultant
* <https://web.archive.org/web/20160925155912/http://www.happybearsoftware.com/how-to-get-a-programmer-job.html>
* Can help add to the project (Will be good for CV)
* <https://github.com/kamranahmedse/developer-roadmap>

## Motivation and Mindset

* Someone with a ‘fixed mindset’ believes that if they don’t get something on their first attempt, they never will
* Someone with a ‘growth mindset’ believes they can get better at anything with effort and persistence
* Intelligence is not fixed and can be developed
* ‘Focus mode’ is when your mind is focused on learning, reading, watching videos, or working on a project
* ‘Diffuse mode’ is when you are not actively learning and is the state where your mind starts to connect what you have been learning (This is why when you are falling asleep you might have a breakthrough on a problem you were stuck on)
* This is why taking a break when you are really stuck on a problem is good as you can usually come up with a solution
* Teaching things to others is a great way to solidify what you have learned

## Asking For Help

* It is essential to ask well formed questions to make it easier for people to help
* Always provide code and the surrounding context
* Ask for help, not the solution itself
* <https://medium.com/@gordon_zhu/how-to-be-great-at-asking-questions-e37be04d0603>
* Text

  Description automatically generated<https://xyproblem.info/> - When people ask about their attempted solution rather than their actual problem.
* <https://stackoverflow.com/help/how-to-ask>
* https://slash7.com/2006/12/22/vampires/

Question Answers:

1. Context or code
2. Where someone asks for help with Y to assist their solution with X when they really should be using an alternative X. Asking about their attempted solution rather than their actual problem.
3. Doesn’t google things/read docs, doesn’t ask specific questions, asks common questions that have already been answered before

## Join the Odin Community

* Working and collaborating with other people is an important part of working as a web developer
* Use the #TheOdinProject
* Community discord
* Rubber duck debugging – Go through the code line by line
* Use backticks in discord to show code different from the rest of the text.
* Use three backticks above and below code for multiple lines of code
* Specify the language after the three backticks to add colour

# Foundations – Prerequisites

There are not many notes for this section as I already know lots about the topics taught in this section, thanks to GCSE’s and A-Level’s.

## Computer Basics

* <https://edu.gcfglobal.org/en/computerbasics/what-is-a-computer/1/>
* <https://edu.gcfglobal.org/en/computerbasics/understanding-operating-systems/1/>
* <https://edu.gcfglobal.org/en/computerbasics/understanding-applications/1/>
* <https://edu.gcfglobal.org/en/basic-computer-skills/open-source-vs-closed-source-software/1/>
* <https://edu.gcfglobal.org/en/techsavvy/taking-screenshots/1/>
* <https://edu.gcfglobal.org/en/techsavvy/password-tips/1/>

Question Answers:

1. Windows is an operating system
2. Open source software is software where the source code is publicly accessible and usable. Closed source software is the opposite.
3. Showing error messages to show IT support and showing evidence of code for exams.
4. Weak: password123 🡪 Strong: P4$$w0rD231

## How Does the Web Work?

* <https://www.youtube.com/watch?v=eHp1l73ztB8>
* <https://developer.mozilla.org/en-US/docs/Learn/Common_questions/How_does_the_Internet_work>
* <https://www.youtube.com/watch?v=7_LPdttKXPc&t=46s>
* <https://www.youtube.com/watch?v=BrXPcaRlBqo>
* <https://developer.mozilla.org/en-US/Learn/Getting_started_with_the_web/How_the_Web_works#Clients_and_servers>

Question Answers:

1. A network is two or more computers that are connected to share data.
2. A global network of networks
3. A unique identifier for a computer using the Internet Protocol
4. A router connects networks together. It allows devices to connect to the internet.
5. A client is a device that is requesting and receiving data from a server and using its services.
6. A server is a device that stores and provides data, services, or programs to client devices.
7. A web page is a document for the WWW, viewed in a web browser.
8. A web server is a server that stores websites and processes HTTP requests
9. A web browser is software that is used for accessing websites/the WWW
10. Software that searches the WWW using search queries
11. A DNS request is a request to a **Domain Name Server** asking it what IP address is associated with a given domain name
12. Google Chrome is the browser I use the most
13. You send a query (a set of keywords) to google who will then go through their database of webpages and find pages that are the most relevant to what you searched. You can then click on one of the pages returned to go through the process of loading the page (DNS request, request to web server, response from web server, website sent to client, web page loaded in client’s browser)

## Installation Overview

* Dual-booting is where you install two operating systems on your computer, which can give you the option to boot either OS when your computer first starts up.
* A virtual machine is an emulation of a computer that runs within your existing OS.

## Installations

* The Odin Project recommends using either a virtual machine or dual-boot to use Linux as it doesn’t support windows.

## Text Editors

* “A good text editor can help you write better code with real-time code checking, syntax highlighting, and automatic formatting.”
* Microsoft Word and Libre-Office Writer cannot be used as they store information about how to display the text on the screen which means interpreters unable to execute the file as code.
* Code editors are tools that can take a text file an provide features such as plugins, syntax highlighting, auto-closing of brackets and braces, and linting. Visual Studio Code (VSCode) is the most popular choice.

## Command Line Basics

* The command line interface (CLI) is where you can enter commands that your computer will run.
* $ is used to show that what follows is a command that should be put into the terminal.

Question Answers:

* A command line is an interface that lets the user enter commands which will be carried out by the computer
* You can open it by clicking the icon on your GUI OS or by pressing CTRL + ALT + T (on linux)
* You can navigate to a directory by entering $ cd /file/path
* cd on its own will take you to your home directory
* $ cd .. will take you back up a directory
* $ pwd will display which directory you are in and the path to get there
* $ ls is used to display the contents of the directory you are in
* $ mkdir directoryName is used to create a new directory
* $ touch fileName is used to create a new file
* $ rm fileName will delete the named file 🡪 rmdir directoryName will delete the named directory
* $ mv oldfile.txt newfile.txt will rename a file 🡪 $ mv oldDirectory newDirectory

## Setting Up Git

* Git is a popular version control system
* GitHub allows you to upload code/files using Git and manage your code using a web interface.

# Foundations – Git Basics

## Introduction to Git

* Git is a version control system – “Git is like a really epic save button for your files and directories”
* A save in Git records differences in the files and folders and keeps a historical record of each save
* Enables you to review how your project grows and restore file states from the past
* GitHub is a remote storage facility on the web for all your projects
* <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>
  + Local Version Control Systems – A local database which stores every file change as a patch. The patches are then put together to re-create a file at a given point in time.
  + Centralised Version Control Systems – All the files and their versions are stored on a central VCS server. Multiple people can then save to and retrieve from the server.
  + Distributed Version Control Systems – A clone of the repository is stored locally on each developer’s computer so that if the server goes down, the repository can be recovered from the mirrored copies stored by all the contributors.
* <https://www.youtube.com/watch?v=8oRjP8yj2Wo>
* <https://www.youtube.com/watch?v=1h9_cB9mPT8&t=13s>
* <https://github.com/TheOdinProject/curriculum>

Question Answers:

1. Git is a version control system (VCS)
2. A text editor saves all the words in a single file. There are no other records of the file recorded so you would need to make multiple copies of the file to keep track of how it has changed. Git records differences in the files and folders and keeps a record of these ‘saves’.
3. Git works at a local level.
4. GitHub works at a remote level.
5. An individual developer can use it to show off their projects to employers as well as make it easier to work on files from different computers. It can make bug fixing easier as you can go back to previous versions to see if they contain the same bug. Branches can be used to work on new files/features without affecting the main branch.
6. Branches allow developers to make changes without stepping on each other’s code. Branches can also be tested before merging them to the main branch to find bugs and problems first. Git will alert developers when there is a merge conflict and tell them where it/they occurred.

## Git Basics

* ‘git clone’ is used to clone the contents of a repository to the current directory.
* ‘git remote’ gives you the URL of the repository you are using.
* ‘git status’ displays the state of the working directory and the staging area.
* ‘git add fileName’ will add a file to the staging area. The staging area is part of the two-step process for making a commit in Git.
* ‘git commit’ makes a ‘save point’ in the repository.
* ‘git log’ shows a log of all previous commits, who made them, message given to the commit, etc.
* ‘git push’ updates a remote branch with local commits.

Questions Answers:

1. A new repository can be made on GitHub by clicking the ‘new’ button
2. The ‘git clone’ command can be used to copy a repository to your local machine
3. ‘Origin’ is the default name of your remote connection
4. origin refers to the remote
5. main refers to the branch you are pushing to
6. You add files to the staging area and the use commit to commit the changes
7. git status
8. ‘git add -A’ or ‘git add fileName’
9. git commit -m “message”
10. git push
11. git log