CS 197U: Introduction to Unix

Lecture 8: Shell Scripting, Development Tools

Instructor: Arun Dunna

Lectures: Monday/Wednesday, 4pm - 5:15pm, LGRC A301

Lab 4 / Quiz 4

Out 2/18

Due Sunday, 2/24, at 11:59pm

Let me know if you have any questions about them.

Shell Scripting

Recall the Unix philosophy of modularity.

In order to execute this effectively in the terminal, we can utilize pipes, redirects, etc.

However, sometimes you may have a series of commands where you want to retain a variable between them such as a file name, or where there are too many commands so one line becomes messy.

For that, we can utilize shell scripting.

- Basically a list of commands executed in the specified order
- Can also use conditional tests (such as >, <, or =), comments, loops, variables, functions, and other logic mechanisms.

Shell Scripting

Shell scripts should end with the `.sh` extension for readability.

Here is the basic format (starting with the shebang):

```
#!/bin/bash
command1
command2
...
```

For example:

```
#!/bin/bash
cd ~
df -h
du -h -d 1
```

Shell Scripting

Running

To run a shell script, we need to give it the execute permission:

```
chmod +x script.sh
```

And then invoke it:

`./script.sh` (or in another dir, `/some/dir/./script.sh`)

Shell Script as a Program

We can also put the shell script into `/usr/local/bin`, for example as `/usr/local/bin/someprog`, and not follow with the `.sh` ext.

This means that we can then run `someprog` from anywhere just like any other program, such as `ls` or `rm`.

Shell Scripting - Variables

The usage of variables lets us do a lot of things, but we'll cover two main uses.

Using As Input

```
NAME="Arun"
SCHOOL="UMass"
echo "Hello $NAME! You go to $SCHOOL!"
```

Getting User Input

```
echo "What is your name?"
read NAME
echo "What is your number?"
read NUM
NUMSQ=$(($NUM * $NUM))
echo "$NAME's number squared is $NUMSQ"
```

Shell Scripting - Comments

Often times, our code can be hard to understand, or we want to make reminders to do changes, etc.

For this reason, programming languages have support for comments. These are lines of code that aren't executed.

For shell scripting comments, we use the `#`:

```
# Get user's first name
read -p "Your first name: " FIRST
# Get user's last name
read -p "Your last name: " LAST
# Output full name
echo "Your full name: $FIRST $LAST"
```

Shell Scripting - Example

`~/calculator.sh`:

```
#!/bin/bash
read -p "Enter your arithmetic phrase: " ARITH
FIN=$(echo $ARITH | bc)
echo "Evaluation: $FIN"
```

We can move this to `/usr/local/bin/calc` and execute with:

`calc`

Anywhere on the system.

Development Tools

Development Tools

I'm gonna kind of build a pyramid of development tools, with the base being your OS and the top being high-level software.

If you have questions about anything or want recommendations on software/tools, I've used and broken a lot of things so feel free to ask.

Development Tools - OS

Operating System

Some people find this to be a super important question. Some don't because lots of software is cross-compatible or a web app.

Primary three are Linux, Mac, and Windows.

But best of all...

You can run any combination of OSes you want in a multi-boot configuration, or in virtual machines.

ex: maintain Windows for games, and Linux for development.

Development Tools - Linux

Pros

- You can choose flavor to minimize tweaking (ex. Ubuntu)
- Support for lots of developer applications
- Has a strong CLI
- Tweakable software and easy to modify hardware
- Minimal resource consumption (can run on a toaster)

Cons

- Is a hassle to fix once you break it
- Will take time to maintain (moreso than Mac/Windows)
- Lack of app support for large apps like Photoshop or games

Deveopment Tools - Mac

Pros

- "It Just Works" - Works well with minimal time to maintain
- Easy to use and pretty lightweight/attractive
- Usually has long battery life
- Unix-based so everything Linux mostly works with Mac

Cons

- Very expensive and pretty locked down
- Repairs are expensive and difficult if do-able at all
- Very specific hardware so cannot pick to add GPUs, etc.
- Worse app support than Windows (but improving), ex. games

Development Tools - Windows

Pros

- Easy to use and setup; decent interface
- Supports the most applications
- Has massive game support compared to other OSes
- Can install on lots of different configurations unlike Mac

Cons

- Typically not great with battery life or resources
- Very different from Unix, so hard to make apps cross-plaform
- Modern Windows 10 installs a lot of bloatware
- Probably spend more time crashing than on Linux maintaining

Development Tools - Window Manager

(If your OS is Linux)

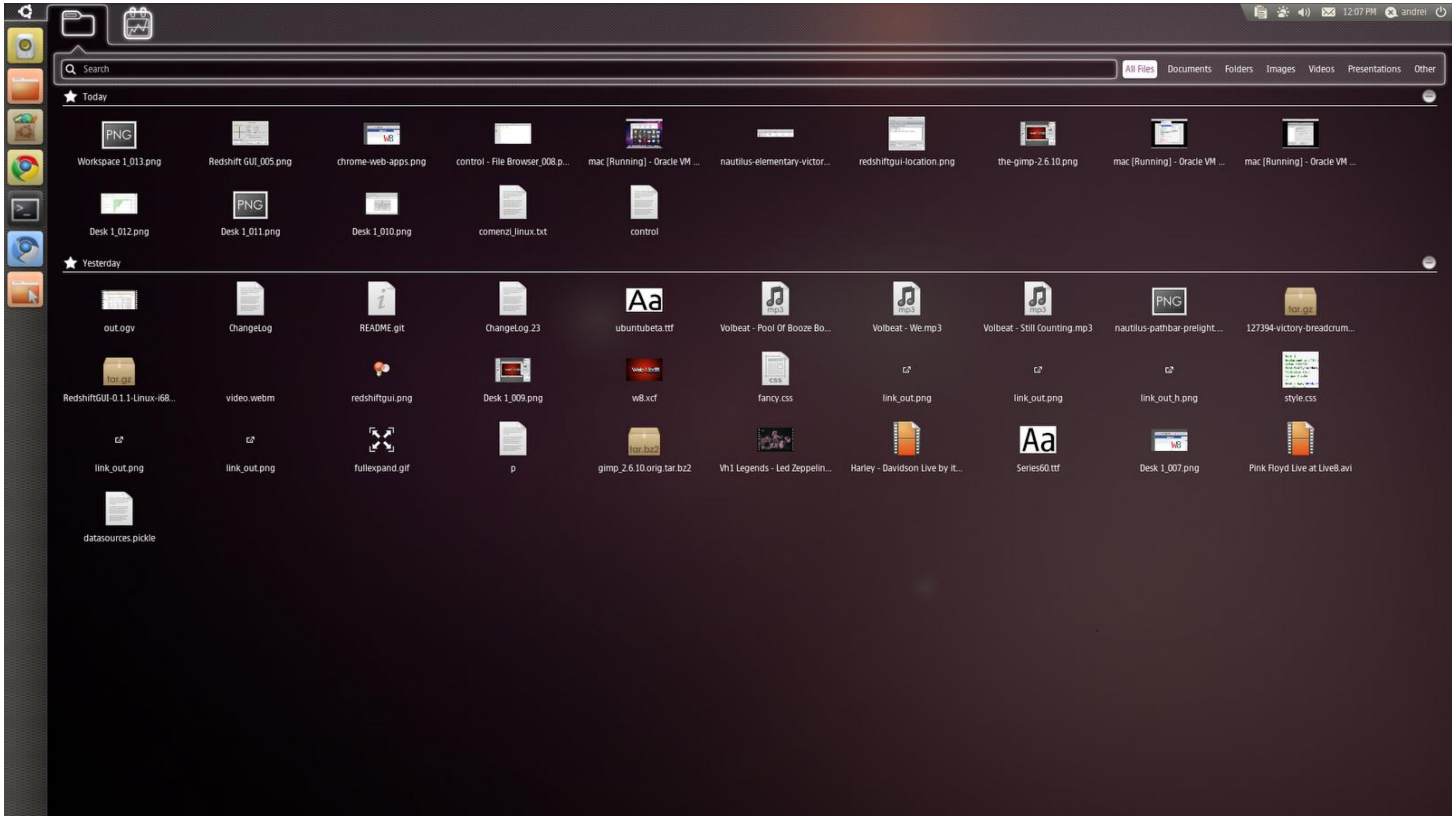
Your window manager is like your user interface.

It defines how windows interact.

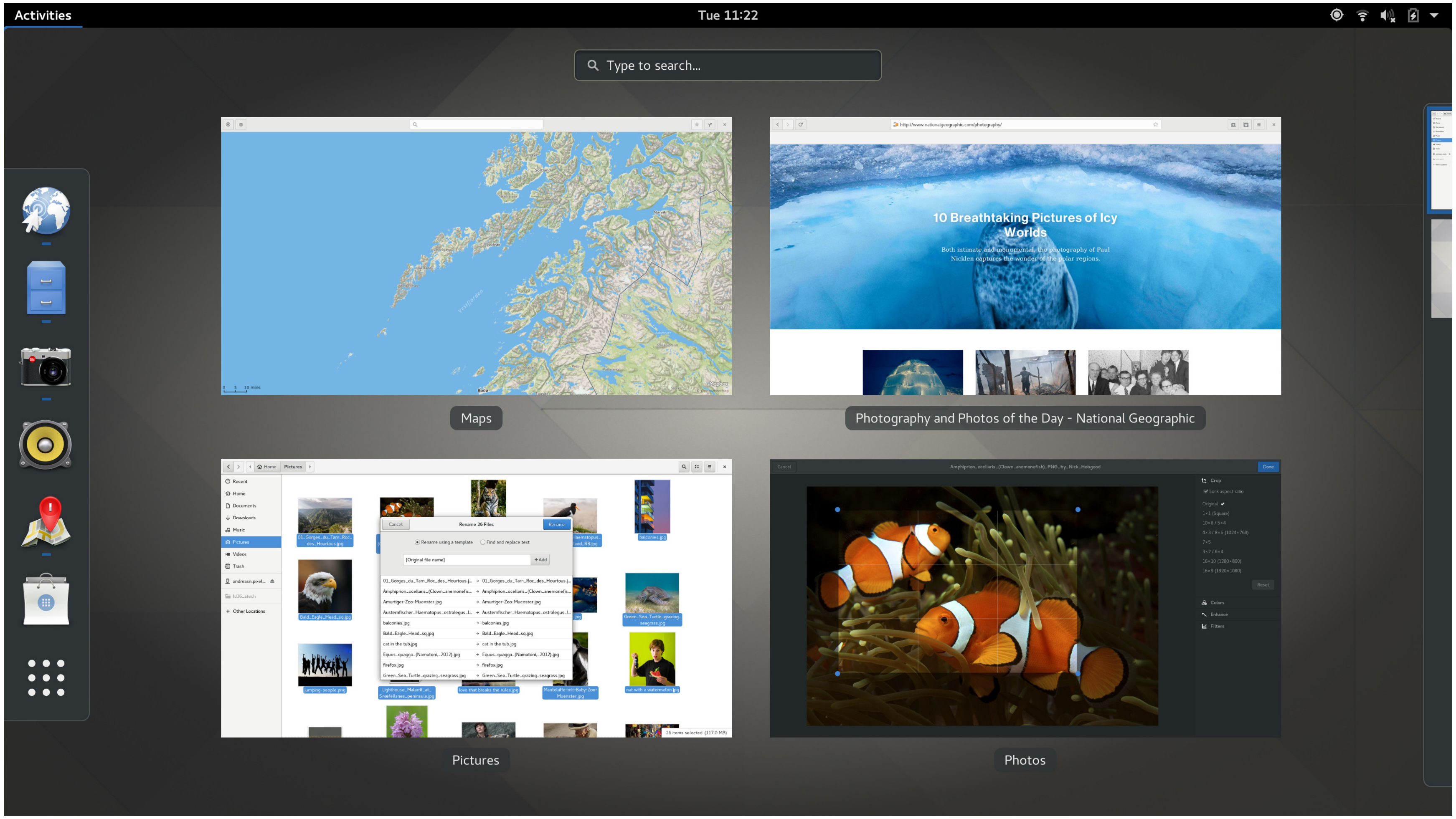
Mac/Windows have their own proprietary ones.

Linux has a bunch of different ones you can use.

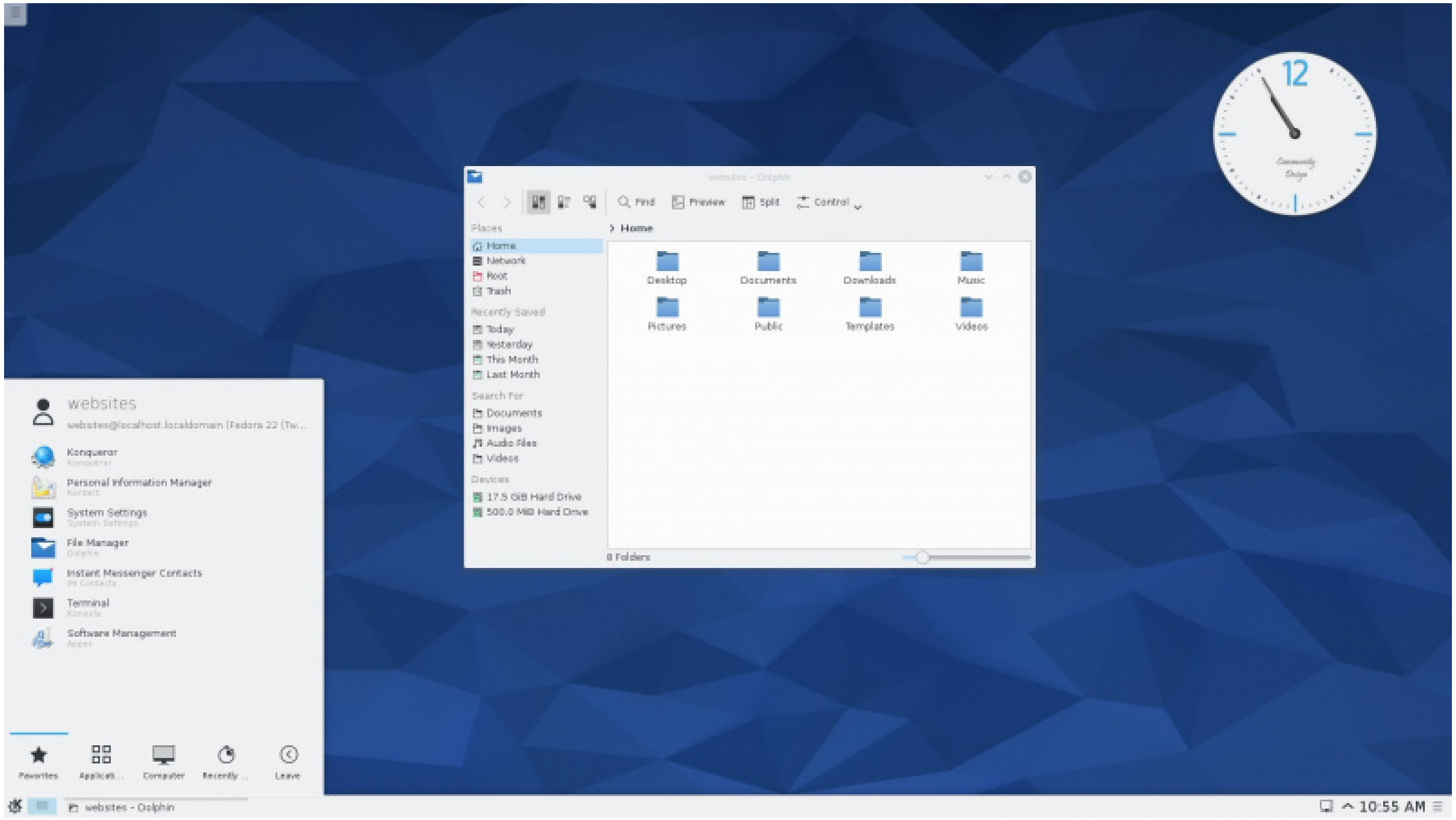
Development Tools - Window Manager



Development Tools - Window Manager



Development Tools - Window Manager



Development Tools - TWM

Tiled Window Manager (TWM)

Is a different type of window manager. I use this type (i3-gaps).

- Puts everything in tiles for moving easily with key shortcuts
- Still can use mouse like normal, but windows are auto sized
    - Can resize/reposition with key shortcuts
- Different modes like tabs/tiles; different WMs like i3 or xmonad

Allows for higher productivity since you don't have to move things around, and can easily shift between workspaces.

Feel free to ask me how to set this up, or for a demonstration.

Development Tools - Terminal

Remember, the terminal is not the shell.

On Linux/Mac, and to some extent Windows, you can pick your terminal emulator.

- Some have different features
- Some look nice or have different aesthetics (colors/animations)
- Some are more/less resource intensive

Some examples are Gnome Terminal, Terminus, xterm, and Terminator.

I use Terminator, but you're free to experiment!

Development Tools - Shell

Runs inside the terminal.

Default is **bash**.

My go-to is **zsh** or **fish**. I use them on different systems.

If you use one, I recommend getting **oh-my-zsh** or **oh-my-fish**, which are pretty good extension packages.

Development Tools - Applications

Now that you have your system, let's talk about software.

- Text Editors
- Code Editors
- Integrated Development Environments (IDEs)
- Organization Software
- Security Tools
- Productivity Applications
- Other Development Tools
- Free Student Stuff

Development Tools - Text Editors

Basic ones are **vim** and **emacs**, you've learned about **vim**.

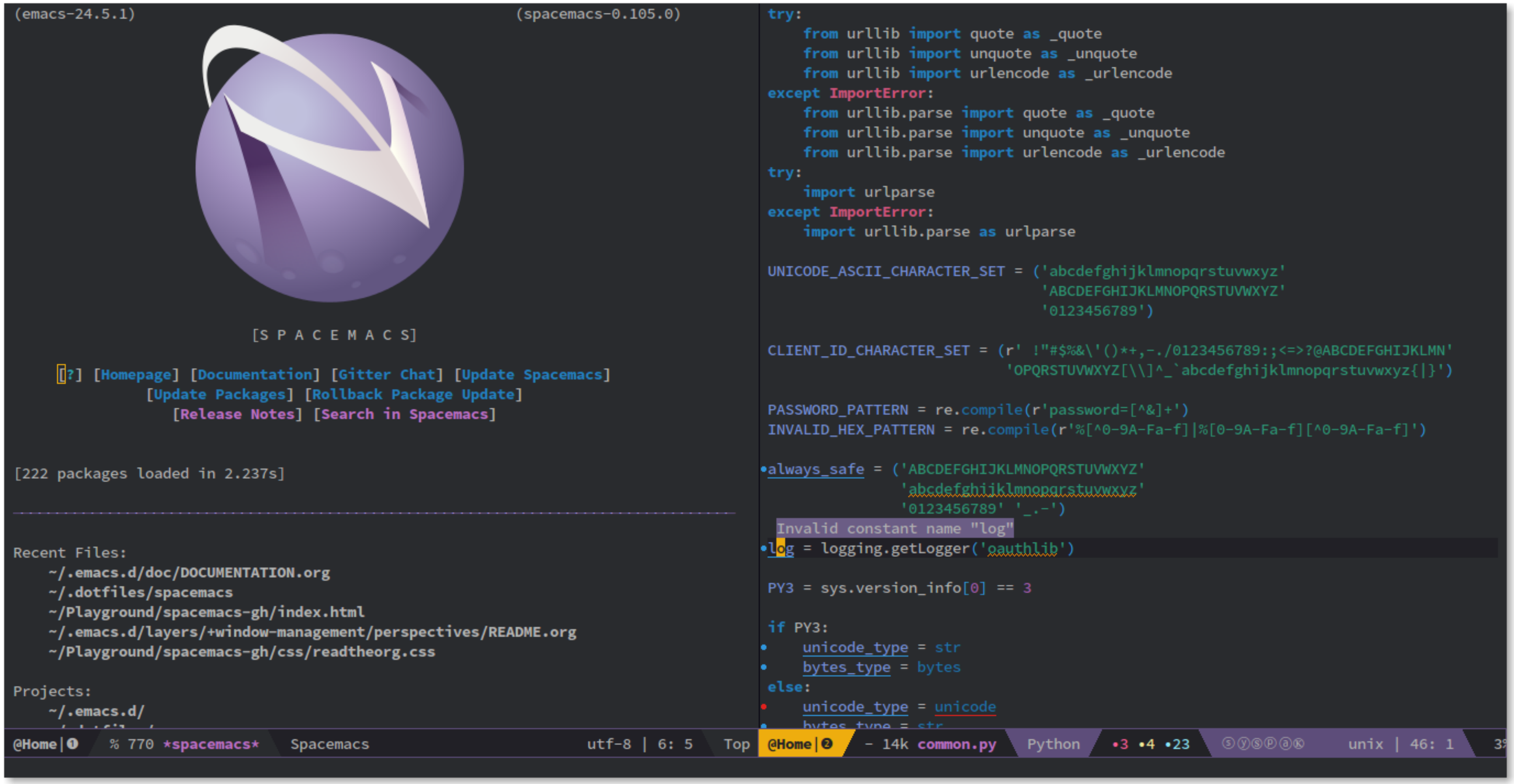
emacs

- Much more complex but depending on how good you get with it, you can basically live inside of it
    - It's like an operating system inside your terminal
    - Can get extensions for web browser, file browser, music player, etc.

If you want to look at emacs, I recommend getting **spacemacs**, which is a bundle of configurations for it. It takes out a lot of hassle.

Note you still have to learn the key commands... and there are a lot. So until you do, expect to have a cheat sheet. :)

Development Tools - Text Editors



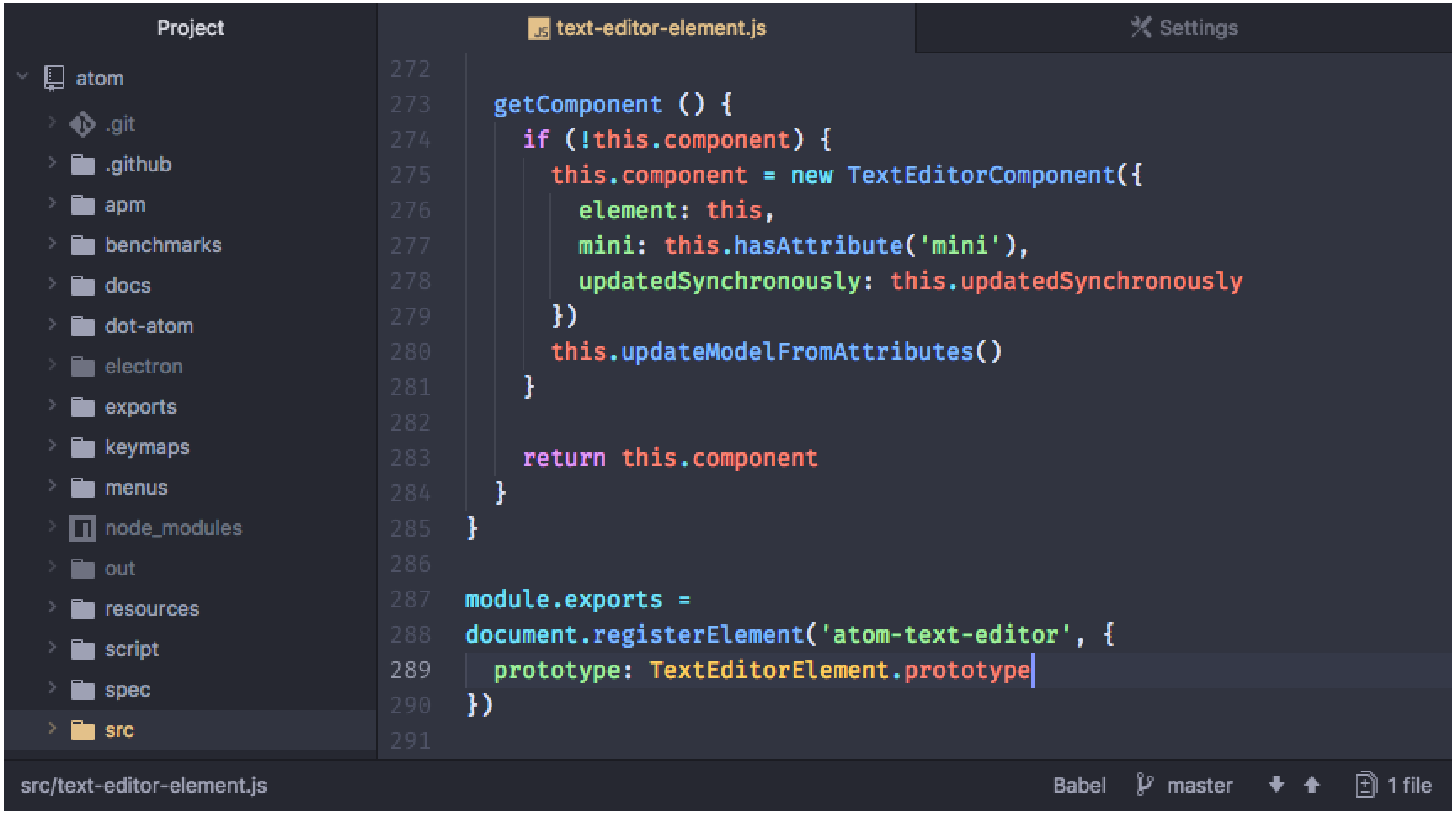
```
(emacs-24.5.1)                                        (spacemacs-0.105.0)

                          [S P A C E M A C S]

    [?] [Homepage] [Documentation] [Gitter Chat] [Update Spacemacs]
            [Update Packages] [Rollback Package Update]
              [Release Notes] [Search in Spacemacs]

[222 packages loaded in 2.237s]

Recent Files:
    ~/.emacs.d/doc/DOCUMENTATION.org
    ~/.dotfiles/spacemacs
    ~/Playground/spacemacs-gh/index.html
    ~/.emacs.d/layers/+window-management/perspectives/README.org
    ~/Playground/spacemacs-gh/css/readtheorg.css

Projects:
    ~/.emacs.d/

@Home|❶  % 770 *spacemacs*  Spacemacs      utf-8 | 6: 5  Top
```

```
try:
    from urllib import quote as _quote
    from urllib import unquote as _unquote
    from urllib import urlencode as _urlencode
except ImportError:
    from urllib.parse import quote as _quote
    from urllib.parse import unquote as _unquote
    from urllib.parse import urlencode as _urlencode
try:
    import urlparse
except ImportError:
    import urllib.parse as urlparse

UNICODE_ASCII_CHARACTER_SET = ('abcdefghijklmnopqrstuvwxyz'
                               'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
                               '0123456789')

CLIENT_ID_CHARACTER_SET = (r' !"#$%&\'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN'
                           'OPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}')

PASSWORD_PATTERN = re.compile(r'password=[^&]+')
INVALID_HEX_PATTERN = re.compile(r'%[^0-9A-Fa-f]|%[0-9A-Fa-f][^0-9A-Fa-f]')

always_safe = ('ABCDEFGHIJKLMNOPQRSTUVWXYZ'
               'abcdefghijklmnopqrstuvwxyz'
               '0123456789' '_.-')
 Invalid constant name "log"
log = logging.getLogger('oauthlib')

PY3 = sys.version_info[0] == 3

if PY3:
    unicode_type = str
    bytes_type = bytes
else:
    unicode_type = unicode
    bytes_type = str

@Home|❷  - 14k common.py  Python  •3 •4 •23  ⓢⓨⓈⓅⓐⓀ  unix | 46: 1  3%
```

Development Tools - Code Editors

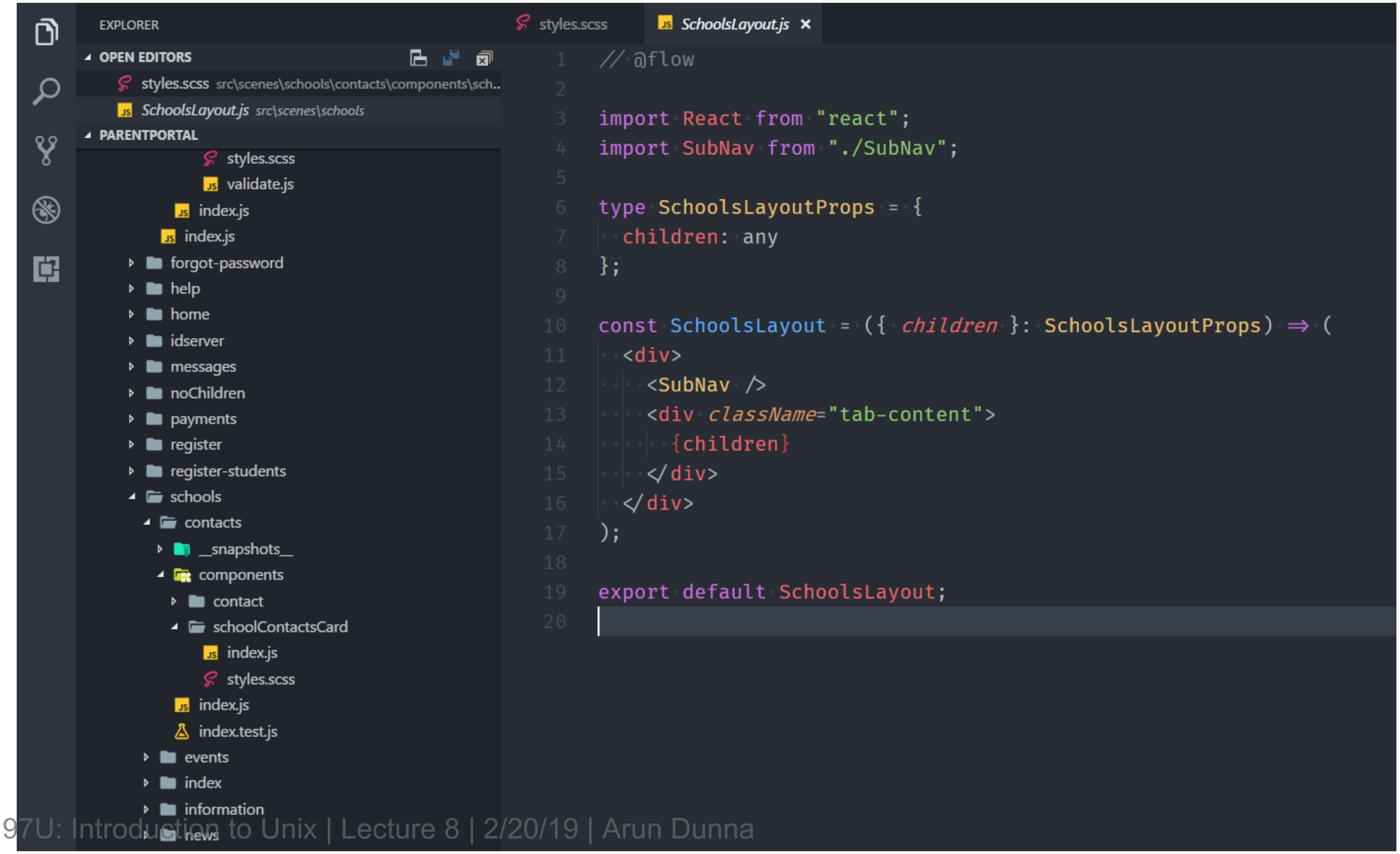
- Like text editors, but with more advanced features
- Doesn't have built-in compilation/interpreting
    - Some have packages that can run compilers or interpreters inside, but mostly it's just for editing code, and you run code outside of it
- Super code-centric, has autocompleting, syntax highlighting, some have project management... ton of developer features
- Popular ones are Atom (mine and very customizable), Visual Studio Code, emacs, vim, and Sublime Text

Development Tools - Code Editors



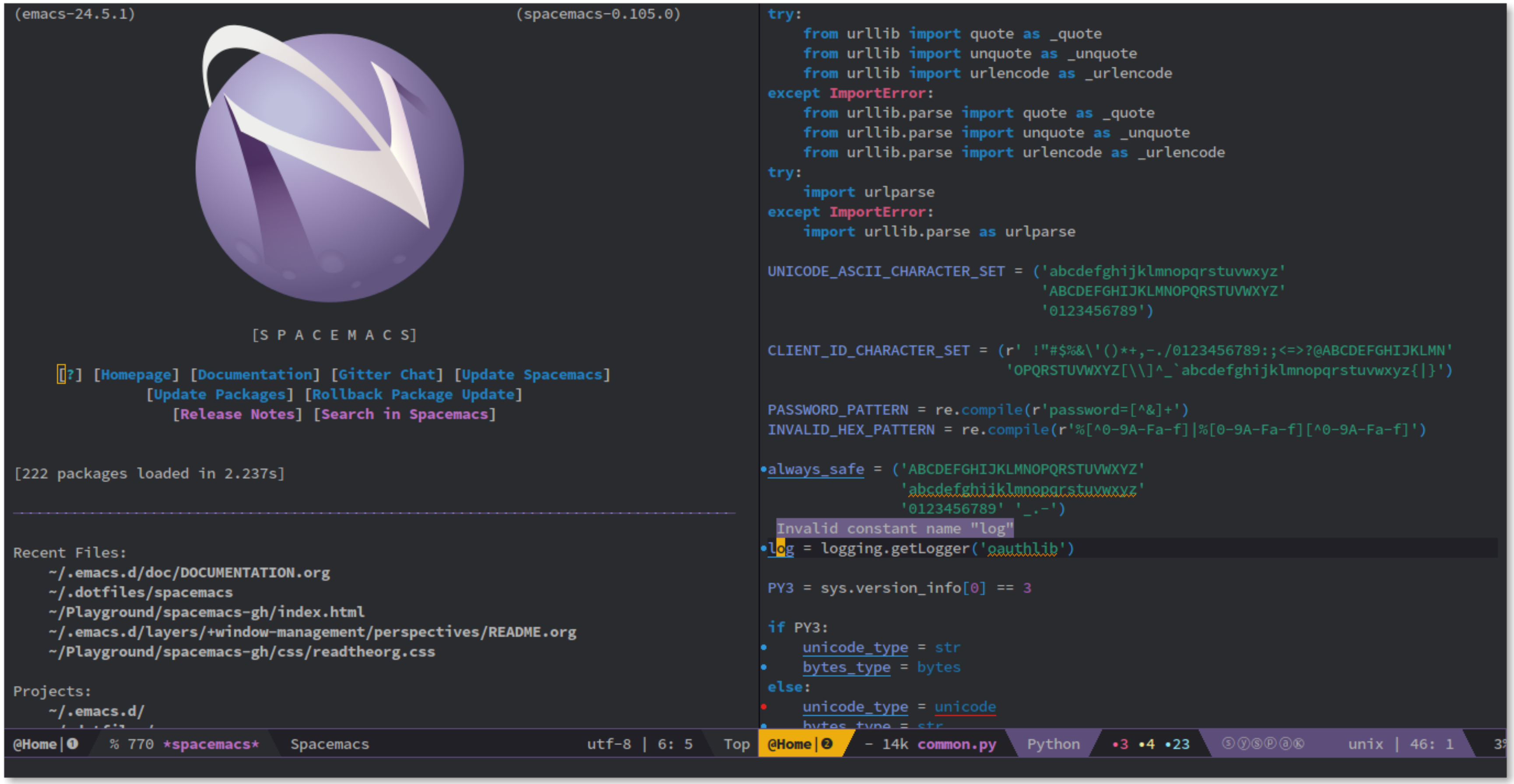
```

  getComponent () {
    if (!this.component) {
      this.component = new TextEditorComponent({
        element: this,
        mini: this.hasAttribute('mini'),
        updatedSynchronously: this.updatedSynchronously
      })
      this.updateModelFromAttributes()
    }

    return this.component
  }
}

module.exports =
document.registerElement('atom-text-editor', {
  prototype: TextEditorElement.prototype
})

```

Development Tools - Code Editors



```
// @flow

import React from "react";
import SubNav from "./SubNav";

type SchoolsLayoutProps = {
  children: any
};

const SchoolsLayout = ({ children }: SchoolsLayoutProps) => (
  <div>
    <SubNav />
    <div className="tab-content">
      {children}
    </div>
  </div>
);

export default SchoolsLayout;
```

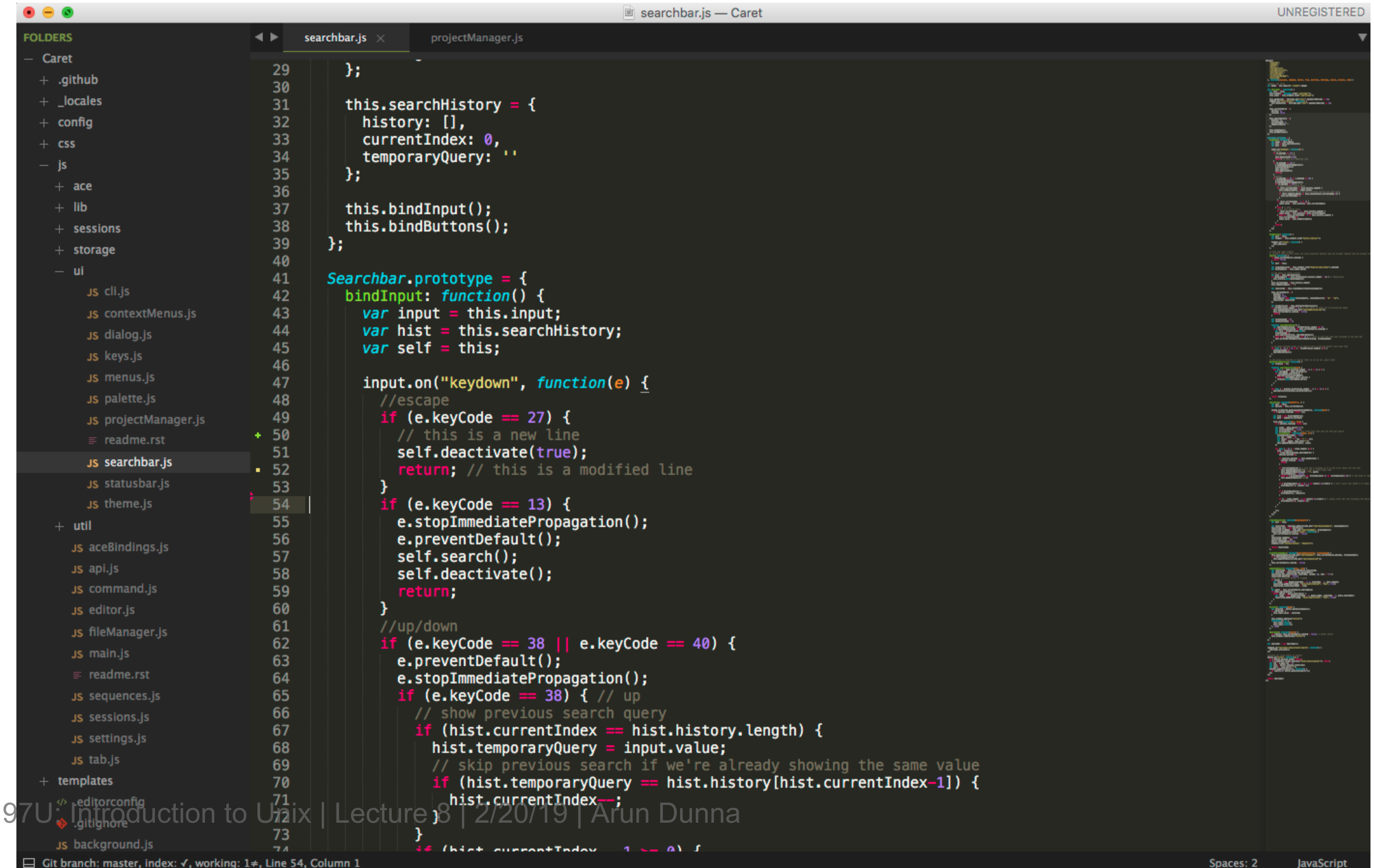
Development Tools - Code Editors



```
(emacs-24.5.1)                                          (spacemacs-0.105.0)

                              [S P A C E M A C S]

    [?] [Homepage] [Documentation] [Gitter Chat] [Update Spacemacs]
            [Update Packages] [Rollback Package Update]
               [Release Notes] [Search in Spacemacs]


[222 packages loaded in 2.237s]

---------------------------------------------------------------------

Recent Files:
    ~/.emacs.d/doc/DOCUMENTATION.org
    ~/.dotfiles/spacemacs
    ~/Playground/spacemacs-gh/index.html
    ~/.emacs.d/layers/+window-management/perspectives/README.org
    ~/Playground/spacemacs-gh/css/readtheorg.css

Projects:
    ~/.emacs.d/
@Home|1   % 770 *spacemacs*   Spacemacs          utf-8 | 6: 5   Top
```

```python
try:
    from urllib import quote as _quote
    from urllib import unquote as _unquote
    from urllib import urlencode as _urlencode
except ImportError:
    from urllib.parse import quote as _quote
    from urllib.parse import unquote as _unquote
    from urllib.parse import urlencode as _urlencode
try:
    import urlparse
except ImportError:
    import urllib.parse as urlparse

UNICODE_ASCII_CHARACTER_SET = ('abcdefghijklmnopqrstuvwxyz'
                               'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
                               '0123456789')

CLIENT_ID_CHARACTER_SET = (r' !"#$%&\'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN'
                           'OPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}')

PASSWORD_PATTERN = re.compile(r'password=[^&]+')
INVALID_HEX_PATTERN = re.compile(r'%[^0-9A-Fa-f]|%[0-9A-Fa-f][^0-9A-Fa-f]')

always_safe = ('ABCDEFGHIJKLMNOPQRSTUVWXYZ'
               'abcdefghijklmnopqrstuvwxyz'
               '0123456789' '_.-')
 Invalid constant name "log"
log = logging.getLogger('oauthlib')

PY3 = sys.version_info[0] == 3

if PY3:
    unicode_type = str
    bytes_type = bytes
else:
    unicode_type = unicode
@Home|2  - 14k common.py   Python   •3 •4 •23   ⓢⓨⓢⓟⓐⓚ   unix | 46: 1   3
```

Development Tools - Code Editors



searchbar.js — Caret

UNREGISTERED

FOLDERS

```
– Caret
  + .github
  + _locales
  + config
  + css
  – js
    + ace
    + lib
    + sessions
    + storage
    – ui
      cli.js
      contextMenus.js
      dialog.js
      keys.js
      menus.js
      palette.js
      projectManager.js
      readme.rst
      searchbar.js
      statusbar.js
      theme.js
    + util
      aceBindings.js
      api.js
      command.js
      editor.js
      fileManager.js
      main.js
      readme.rst
      sequences.js
      sessions.js
      settings.js
      tab.js
  + templates
    .editorconfig
    .gitignore
    background.js
```

searchbar.js | projectManager.js

```
  };

  this.searchHistory = {
    history: [],
    currentIndex: 0,
    temporaryQuery: ''
  };

  this.bindInput();
  this.bindButtons();
};

Searchbar.prototype = {
  bindInput: function() {
    var input = this.input;
    var hist = this.searchHistory;
    var self = this;

    input.on("keydown", function(e) {
      //escape
      if (e.keyCode == 27) {
        // this is a new line
        self.deactivate(true);
        return; // this is a modified line
      }
      if (e.keyCode == 13) {
        e.stopImmediatePropagation();
        e.preventDefault();
        self.search();
        self.deactivate();
        return;
      }
      //up/down
      if (e.keyCode == 38 || e.keyCode == 40) {
        e.preventDefault();
        e.stopImmediatePropagation();
        if (e.keyCode == 38) { // up
          // show previous search query
          if (hist.currentIndex == hist.history.length) {
            hist.temporaryQuery = input.value;
            // skip previous search if we're already showing the same value
            if (hist.temporaryQuery == hist.history[hist.currentIndex-1]) {
              hist.currentIndex--;
            }
          }
          if (hist.currentIndex - 1 >= 0) {
```

Git branch: master, index: ✓, working: 1≠, Line 54, Column 1 | Spaces: 2 | JavaScript

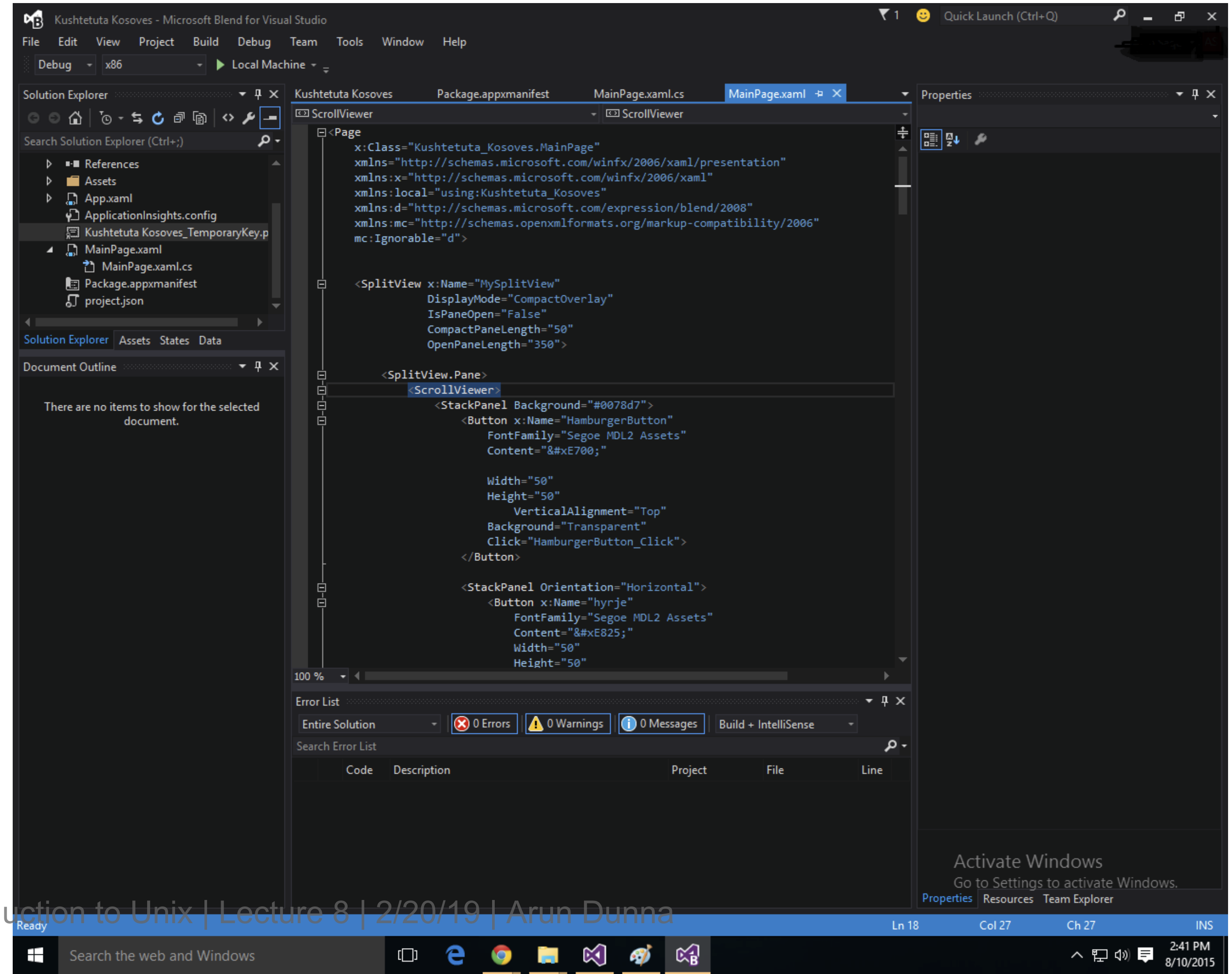
Development Tools - IDEs

Are code editors with even more features, and mean to contain the whole software development process: writing code, managing the project, building, testing, debugging, etc.

Usually language-specific.

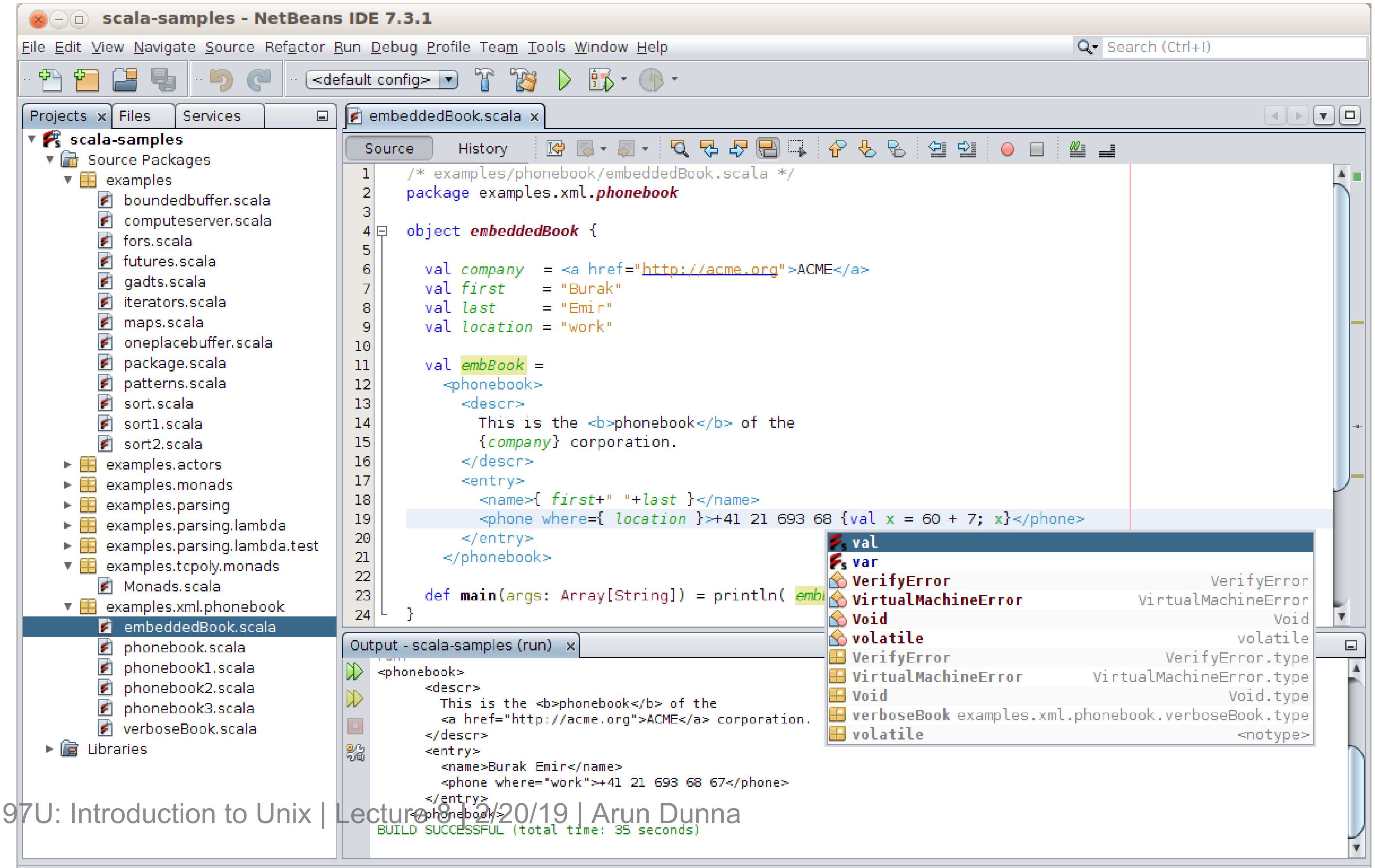
- Visual Studio: JavaScript, BASIC, C#, C++, others
- NetBeans: C, C++, Fortran, HTML, PHP, Java, others
- PyCharm: Python, Node.js, JavaScript, HTML, others
- Intellij IDEA: Java, JavaScript, PHP, Python, others
- Eclipse: Java, C, C++, PHP, Python, Ruby, others
- XCode: Java, C, C++, AppleScript

Development Tools - IDEs


Kushtetuta Kosoves - Microsoft Blend for Visual Studio
File Edit View Project Build Debug Team Tools Window Help
Debug x86 Local Machine
Quick Launch (Ctrl+Q)
[redacted]
Solution Explorer
Search Solution Explorer (Ctrl+;)
References
Assets
App.xaml
ApplicationInsights.config
Kushtetuta Kosoves_TemporaryKey.p
MainPage.xaml
MainPage.xaml.cs
Package.appxmanifest
project.json
Solution Explorer Assets States Data
Document Outline
There are no items to show for the selected document.
Kushtetuta Kosoves Package.appxmanifest MainPage.xaml.cs MainPage.xaml
ScrollViewer ScrollViewer
Properties
100 %
Error List
Entire Solution 0 Errors 0 Warnings 0 Messages Build + IntelliSense
Search Error List
Code Description Project File Line
Activate Windows
Go to Settings to activate Windows.
Properties Resources Team Explorer
Ready Ln 18 Col 27 Ch 27 INS
Search the web and Windows
2:41 PM
8/10/2015

```
<Page
    x:Class="Kushtetuta_Kosoves.MainPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="using:Kushtetuta_Kosoves"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d">


    <SplitView x:Name="MySplitView"
               DisplayMode="CompactOverlay"
               IsPaneOpen="False"
               CompactPaneLength="50"
               OpenPaneLength="350">

        <SplitView.Pane>
            <ScrollViewer>
                <StackPanel Background="#0078d7">
                    <Button x:Name="HamburgerButton"
                        FontFamily="Segoe MDL2 Assets"
                        Content="&#xE700;"

                        Width="50"
                        Height="50"
                            VerticalAlignment="Top"
                        Background="Transparent"
                        Click="HamburgerButton_Click">
                    </Button>

                    <StackPanel Orientation="Horizontal">
                        <Button x:Name="hyrje"
                            FontFamily="Segoe MDL2 Assets"
                            Content="&#xE825;"
                            Width="50"
                            Height="50"
```

Development Tools - IDEs

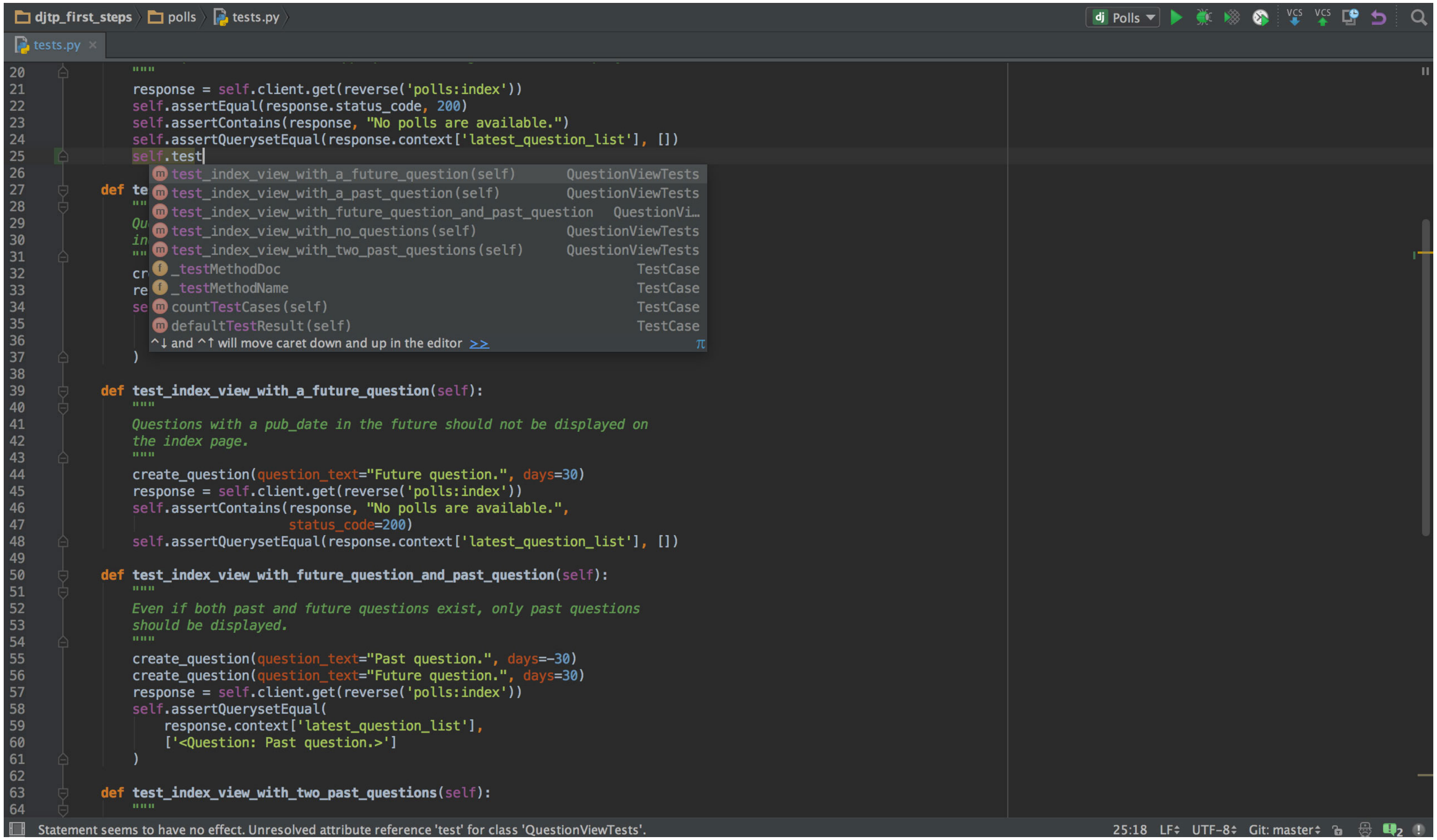


```
/* examples/phonebook/embeddedBook.scala */
package examples.xml.phonebook

object embeddedBook {

  val company  = <a href="http://acme.org">ACME</a>
  val first    = "Burak"
  val last     = "Emir"
  val location = "work"

  val embBook =
    <phonebook>
      <descr>
        This is the <b>phonebook</b> of the
        {company} corporation.
      </descr>
      <entry>
        <name>{ first+" "+last }</name>
        <phone where={ location }>+41 21 693 68 {val x = 60 + 7; x}</phone>
      </entry>
    </phonebook>

  def main(args: Array[String]) = println( embl
}
```

```
val
var
VerifyError                                VerifyError
VirtualMachineError                VirtualMachineError
Void                                              Void
volatile                                      volatile
VerifyError                           VerifyError.type
VirtualMachineError           VirtualMachineError.type
Void                                         Void.type
verboseBook examples.xml.phonebook.verboseBook.type
volatile                                      <notype>
```

```
<phonebook>
      <descr>
        This is the <b>phonebook</b> of the
        <a href="http://acme.org">ACME</a> corporation.
      </descr>
      <entry>
        <name>Burak Emir</name>
        <phone where="work">+41 21 693 68 67</phone>
      </entry>
    </phonebook>
BUILD SUCCESSFUL (total time: 35 seconds)
```

Development Tools - IDEs

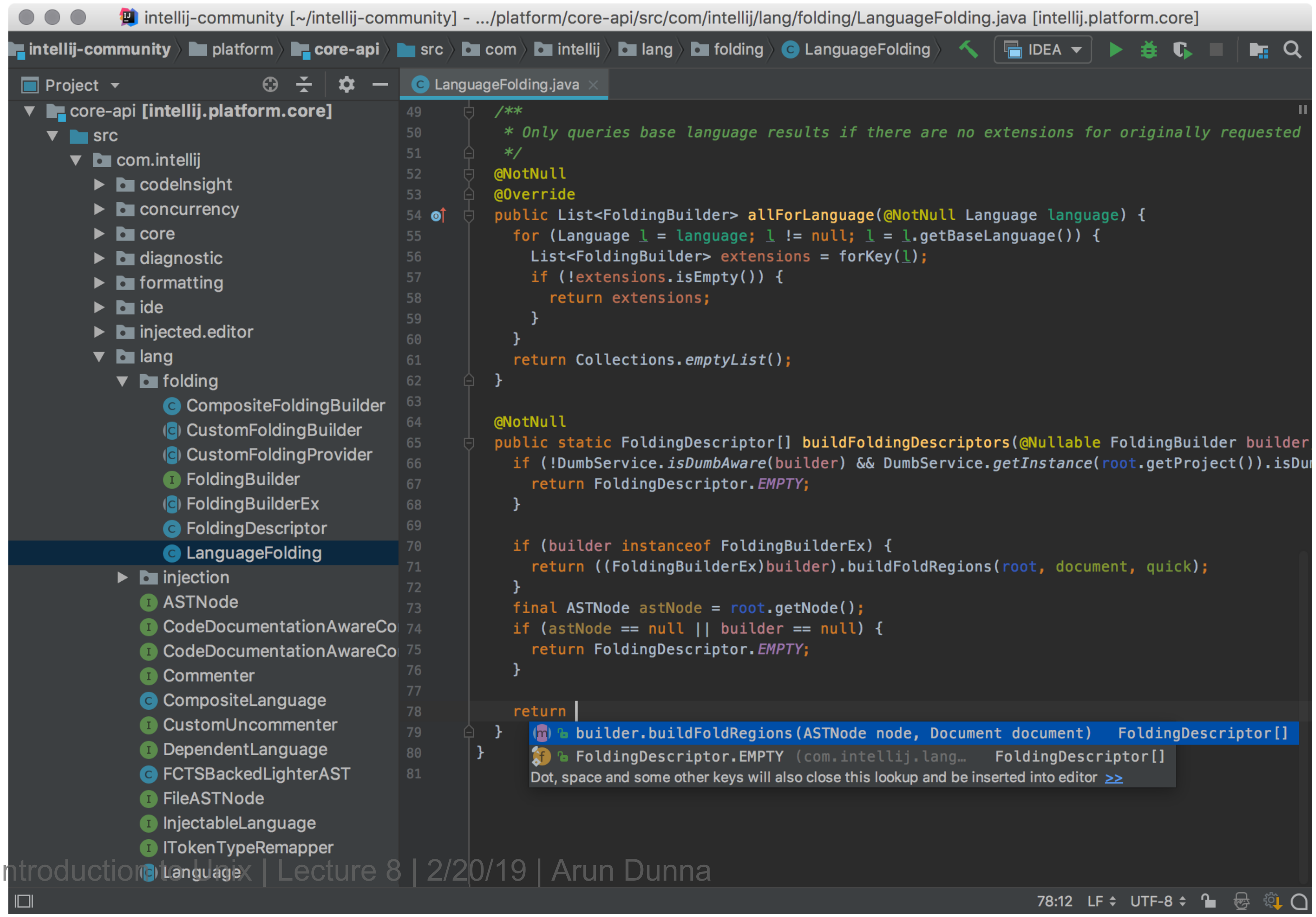

djtp_first_steps
polls
tests.py
Polls
VCS
VCS
tests.py
test_index_view_with_a_future_question(self) QuestionViewTests
test_index_view_with_a_past_question(self) QuestionViewTests
test_index_view_with_future_question_and_past_question QuestionVi…
test_index_view_with_no_questions(self) QuestionViewTests
test_index_view_with_two_past_questions(self) QuestionViewTests
_testMethodDoc TestCase
_testMethodName TestCase
countTestCases(self) TestCase
defaultTestResult(self) TestCase
^↓ and ^↑ will move caret down and up in the editor >>
π

```
        """
        response = self.client.get(reverse('polls:index'))
        self.assertEqual(response.status_code, 200)
        self.assertContains(response, "No polls are available.")
        self.assertQuerysetEqual(response.context['latest_question_list'], [])
        self.test

    def te
        """
        Qu
        in
        """
        cr
        re
        se


        )

    def test_index_view_with_a_future_question(self):
        """
        Questions with a pub_date in the future should not be displayed on
        the index page.
        """
        create_question(question_text="Future question.", days=30)
        response = self.client.get(reverse('polls:index'))
        self.assertContains(response, "No polls are available.",
                            status_code=200)
        self.assertQuerysetEqual(response.context['latest_question_list'], [])

    def test_index_view_with_future_question_and_past_question(self):
        """
        Even if both past and future questions exist, only past questions
        should be displayed.
        """
        create_question(question_text="Past question.", days=-30)
        create_question(question_text="Future question.", days=30)
        response = self.client.get(reverse('polls:index'))
        self.assertQuerysetEqual(
            response.context['latest_question_list'],
            ['<Question: Past question.>']
        )

    def test_index_view_with_two_past_questions(self):
        """
```

Statement seems to have no effect. Unresolved attribute reference 'test' for class 'QuestionViewTests'.

25:18 LF÷ UTF-8÷ Git: master÷

Development Tools - IDEs

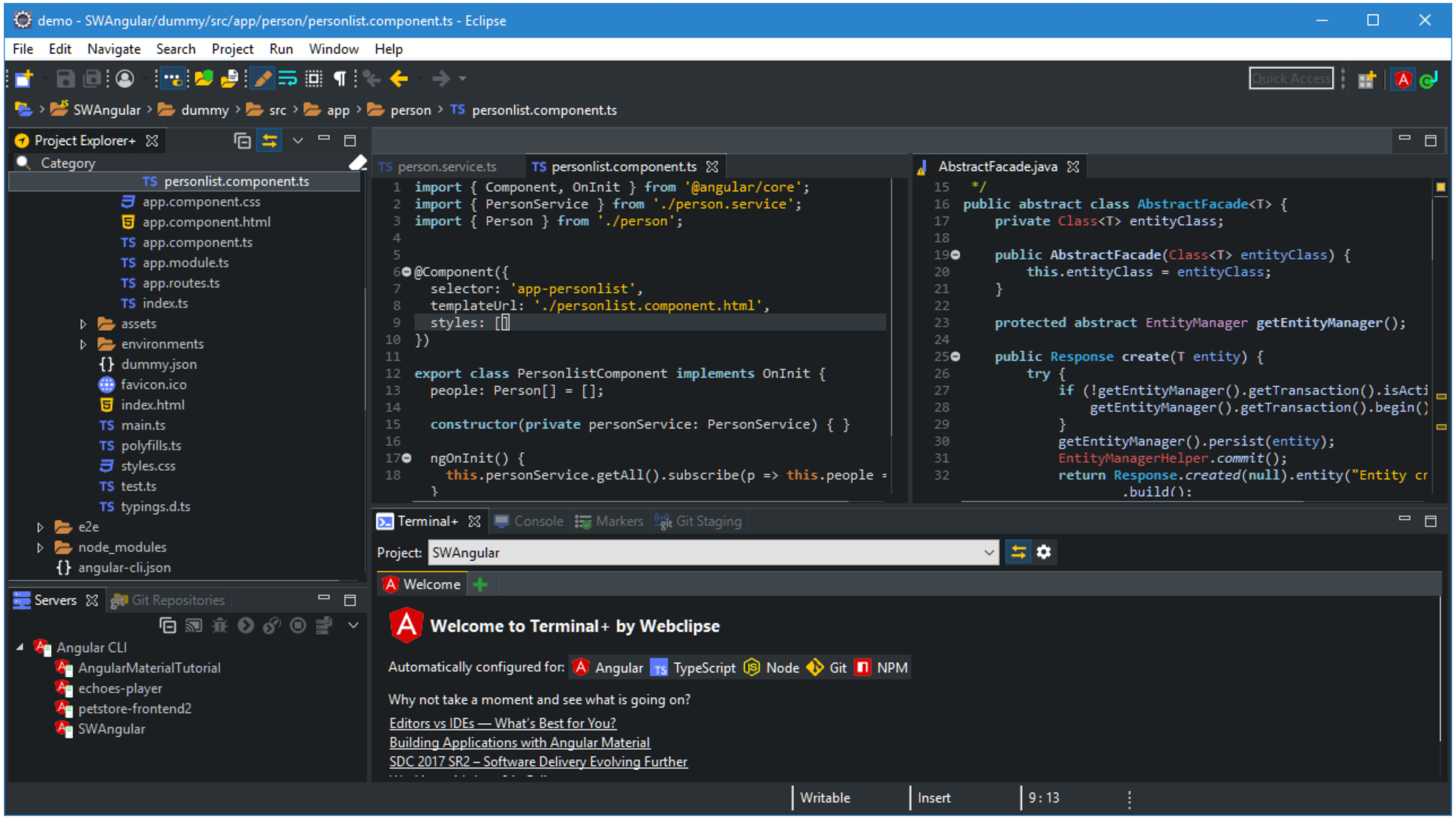


intellij-community [~/intellij-community] - .../platform/core-api/src/com/intellij/lang/folding/LanguageFolding.java [intellij.platform.core]

```
/**
 * Only queries base language results if there are no extensions for originally requested
 */
@NotNull
@Override
public List<FoldingBuilder> allForLanguage(@NotNull Language language) {
  for (Language l = language; l != null; l = l.getBaseLanguage()) {
    List<FoldingBuilder> extensions = forKey(l);
    if (!extensions.isEmpty()) {
      return extensions;
    }
  }
  return Collections.emptyList();
}

@NotNull
public static FoldingDescriptor[] buildFoldingDescriptors(@Nullable FoldingBuilder builder
  if (!DumbService.isDumbAware(builder) && DumbService.getInstance(root.getProject()).isDu
    return FoldingDescriptor.EMPTY;
  }

  if (builder instanceof FoldingBuilderEx) {
    return ((FoldingBuilderEx)builder).buildFoldRegions(root, document, quick);
  }
  final ASTNode astNode = root.getNode();
  if (astNode == null || builder == null) {
    return FoldingDescriptor.EMPTY;
  }

  return |
}
}
```

```
builder.buildFoldRegions(ASTNode node, Document document)   FoldingDescriptor[]
FoldingDescriptor.EMPTY (com.intellij.lang…     FoldingDescriptor[]
Dot, space and some other keys will also close this lookup and be inserted into editor >>
```

Development Tools - IDEs



demo - SWAngular/dummy/src/app/person/personlist.component.ts - Eclipse

File Edit Navigate Search Project Run Window Help

SWAngular › dummy › src › app › person › personlist.component.ts

Project Explorer+

- personlist.component.ts
- app.component.css
- app.component.html
- app.component.ts
- app.module.ts
- app.routes.ts
- index.ts
- assets
- environments
- dummy.json
- favicon.ico
- index.html
- main.ts
- polyfills.ts
- styles.css
- test.ts
- typings.d.ts
- e2e
- node_modules
- angular-cli.json

person.service.ts | personlist.component.ts

```
import { Component, OnInit } from '@angular/core';
import { PersonService } from './person.service';
import { Person } from './person';


@Component({
  selector: 'app-personlist',
  templateUrl: './personlist.component.html',
  styles: []
})

export class PersonlistComponent implements OnInit {
  people: Person[] = [];

  constructor(private personService: PersonService) { }

  ngOnInit() {
    this.personService.getAll().subscribe(p => this.people =
  }
```

AbstractFacade.java

```
 */
public abstract class AbstractFacade<T> {
    private Class<T> entityClass;

    public AbstractFacade(Class<T> entityClass) {
        this.entityClass = entityClass;
    }

    protected abstract EntityManager getEntityManager();

    public Response create(T entity) {
        try {
            if (!getEntityManager().getTransaction().isActi
                getEntityManager().getTransaction().begin()
            }
            getEntityManager().persist(entity);
            EntityManagerHelper.commit();
            return Response.created(null).entity("Entity cr
                    .build();
```

Servers | Git Repositories

- Angular CLI
  - AngularMaterialTutorial
  - echoes-player
  - petstore-frontend2
  - SWAngular

Terminal+ | Console | Markers | Git Staging

Project: SWAngular

Welcome

Welcome to Terminal+ by Webclipse

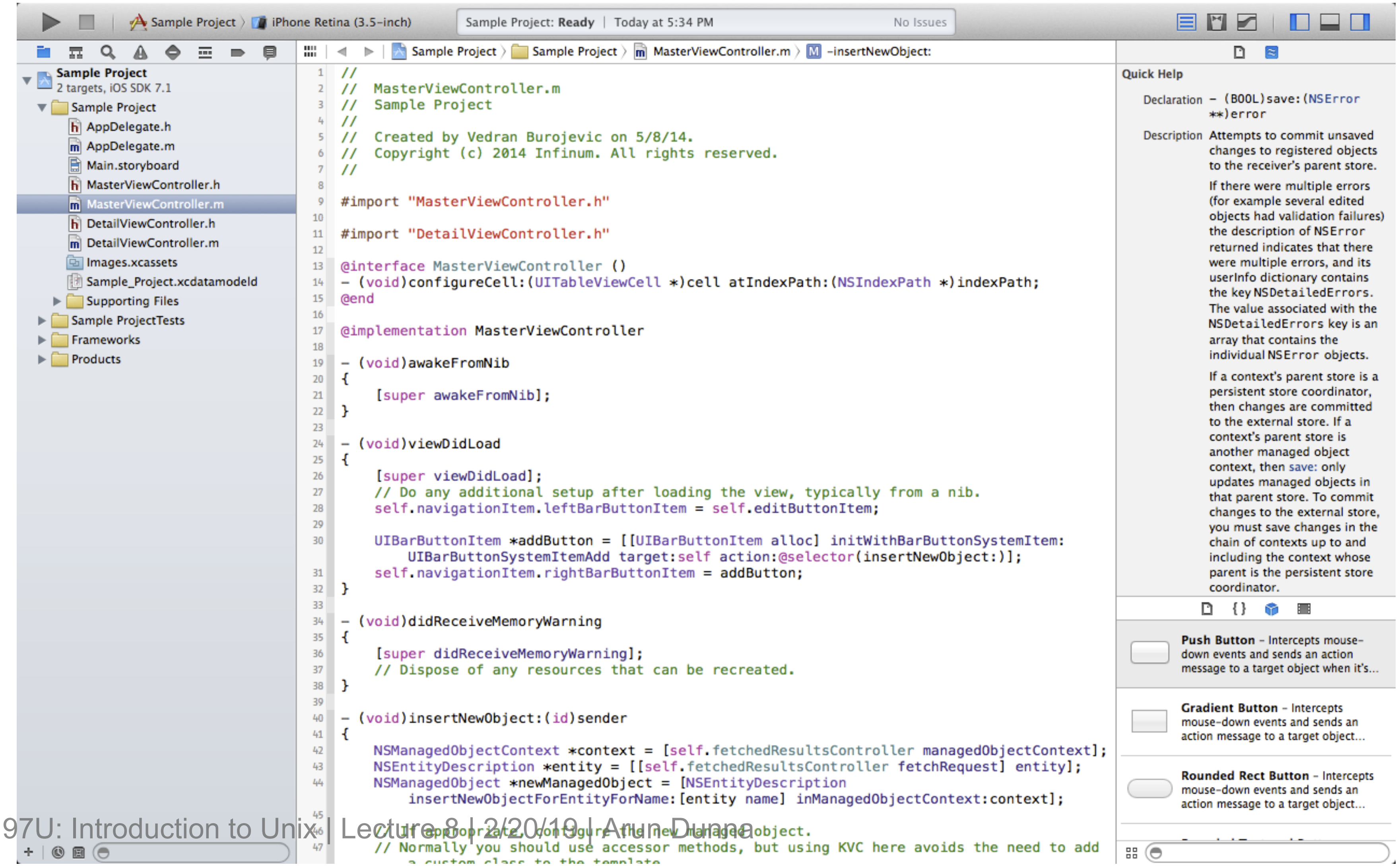
Automatically configured for: Angular TypeScript Node Git NPM

Why not take a moment and see what is going on?

Editors vs IDEs — What's Best for You?
Building Applications with Angular Material
SDC 2017 SR2 – Software Delivery Evolving Further

Writable | Insert | 9 : 13

Development Tools - IDEs



Sample Project › iPhone Retina (3.5-inch) | Sample Project: **Ready** | Today at 5:34 PM | No Issues

Sample Project › Sample Project › MasterViewController.m › -insertNewObject:

```
//
//  MasterViewController.m
//  Sample Project
//
//  Created by Vedran Burojevic on 5/8/14.
//  Copyright (c) 2014 Infinum. All rights reserved.
//

#import "MasterViewController.h"

#import "DetailViewController.h"

@interface MasterViewController ()
- (void)configureCell:(UITableViewCell *)cell atIndexPath:(NSIndexPath *)indexPath;
@end

@implementation MasterViewController

- (void)awakeFromNib
{
    [super awakeFromNib];
}

- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view, typically from a nib.
    self.navigationItem.leftBarButtonItem = self.editButtonItem;

    UIBarButtonItem *addButton = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
        UIBarButtonSystemItemAdd target:self action:@selector(insertNewObject:)];
    self.navigationItem.rightBarButtonItem = addButton;
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

- (void)insertNewObject:(id)sender
{
    NSManagedObjectContext *context = [self.fetchedResultsController managedObjectContext];
    NSEntityDescription *entity = [[self.fetchedResultsController fetchRequest] entity];
    NSManagedObject *newManagedObject = [NSEntityDescription
        insertNewObjectForEntityForName:[entity name] inManagedObjectContext:context];

    // If appropriate, configure the new managed object.
    // Normally you should use accessor methods, but using KVC here avoids the need to add
```

Quick Help

Declaration - (BOOL)save:(NSError **)error

Description Attempts to commit unsaved changes to registered objects to the receiver's parent store.

If there were multiple errors (for example several edited objects had validation failures) the description of NSError returned indicates that there were multiple errors, and its userInfo dictionary contains the key NSDetailedErrors. The value associated with the NSDetailedErrors key is an array that contains the individual NSError objects.

If a context's parent store is a persistent store coordinator, then changes are committed to the external store. If a context's parent store is another managed object context, then save: only updates managed objects in that parent store. To commit changes to the external store, you must save changes in the chain of contexts up to and including the context whose parent is the persistent store coordinator.

Push Button - Intercepts mouse-down events and sends an action message to a target object when it's...

Gradient Button - Intercepts mouse-down events and sends an action message to a target object...

Rounded Rect Button - Intercepts mouse-down events and sends an action message to a target object...

Development Tools - Organization

Task Manager (ex. Todoist, any.do, Google Tasks)

- Useful for cross-platform syncing of todo lists with reminders
- Keep track of assignments, tasks, etc.

Calendar Application (ex. Google Calendar, Apple iCal)

- Needed to keep track of classes, events, and meetings
- Coordinate meetings between people

Paper Manager (ex. Mendeley, Zotero)

- Sync papers, annotations, citations, notes, etc.

Chat Applications

- Slack, Hangouts, SMS, Email, and some IM app

Development Tools - Security Tools

Password Manager (ex. Lastpass, KeePassX, Dashlane)

- Generate and remember random passwords for you securely

Browser and Extensions (ex. Firefox, Chrome)

- Browsers: Firefox or Chrome, not Edge ugh
  - Use Firefox for more privacy, Chrome if you want all the features and sync stuff, and browser apps
- Extensions: Privacy Badger, uBlock Origin, HTTPS Everywhere (I have more but these cover 99% of cases)

Development Tools - Productivity Apps

Office Apps

- Microsoft Office Online (I use with OneNote on my Surface for taking notes) - you get this free as a student
- Google Suite (Docs, Slides, Sheets, Drive) - great for collaboration
- Libre Office for offline Linux
- Microsoft Office for offline Windows and Mac

Development Tools - Other

Cloud Providers

- Not as relevant now but will be later as you progress
    - Amazon Web Services (AWS), Google Compute, Microsoft Azure are main ones

Version Control & Collaboration

- GitHub (most widely used, free), GitLab, or BitBucket

Static Website Generator

- Easy to generate/update websites (mine/course site are ex.)

LaTeX

- Used to create most academic documents (like our lab documents, research papers, etc.) - who wants more on this?

Development Tools - Free Stuff

- GitHub Education Develoepr Pack
  - https://education.github.com/pack
  - Gives you a bunch of free or discounted subscriptions/credits for services, like free domains, AWS credits, etc.
- Spotify for $5/mo as a student
  - Not just a music application... it's a way of life. Podcasts, music, syncing cross-platform, add your own stuff, etc.

Any other questions for development tools? Want input for specific tools or types of tools?

Wrap Up

That's the end of the core Intro to Unix course content.

Lab/Quiz 4 due **Sunday 2/24** at **11:59pm**.

Next Lectures:

- Lecture 9: CS Topics & Courses, Window Managers, Software Lifecycle
- **Lecture 10: Guest Lecture and Final Review**
- Lecture 11: Internships, Grad School, Resumes, Coding Interviews, and LaTeX

I am still counting attendance for these. There will be no Lab 5. Quiz 5 will be an extra credit Quiz that will replace your second lowest Quiz score. Your lowest Quiz score will still be dropped.