# Noteable™

## A cloud computational notebook service

**Guy Stevenson, Sales Manager, EDINA**

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# EDINA

**EDINA at The University of Edinburgh, renowned trusted online service provider with 20+ years experience delivering online services at scale**

- **Deliver portfolio of online services to over 5000 customers**
- **Best practice ITIL service delivery**
- **Regulatory best practice - GDPR and Digital Accessibility**
- **Help Desk Mon-Fri 9-5**
- **99.99% service uptime**
- **Team of software developers and service delivery staff**

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# What is Noteable?

- Noteable is a managed, cloud-based computational notebook service developed by EDINA at the University of Edinburgh
- Designed with input from learning technologists and lecturers to support the teaching of coding
- Provides access to several different computational notebook
- Guaranteed consistent coding environment for all users

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# Noteable features

- Seven notebook types
  - *Python, R with stan, R Studio, SageMath, Geospatial, Chemistry, Machine Learning*
- Pre installed, managed and maintained libraries within each notebook
- Full VLE integration, accessed via courses defined within your VLE
- Features to assist teaching: collaborative coding, multiple markers, assignment setting, autograding
- 10GB permanent storage and 4GB of RAM for every user

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# Python notebook

Standard Notebook is the default python 3 notebook, with a wide selection of libraries includes.

- Based on the jupyter/minimal-notebook image.
- Includes nbgrader and +GitRepo tools for document sharing.
- Provide rubberband, exercise, and exercise-2 extensions. **Note** these are not related to nbgrader in any way.
- Data Access libraries: beautifulsoup4, dill, gitdb2, libxml2, lxml, mysql-connector-python, openpyxl, protobuf, pyxlsb, sqlalchemy, sqlite, textblob, unixodbc, xlrd, xlwt, xlsxwriter, and xmlschema
- Data Science libraries: dask, pandas, and pointpats
- Data Visualization tools: altair, bokeh, demesdraw, graphviz, hvplot, ipywidgets, ipympl, k3d, matplotlib-base, mobilechelonian, mpld3, palettable, plotly, pykrige, pymc3, seaborn, terminaltables, widgetsnbextension, and wordcloud
- Image Processing: imagemagick, opencv, scikit-image, and spectral
- Machine Learning libraries: dask-ml and scikit-learn
- Stats & Data Modelling libraries: h5py, hdf5, numexpr, numpy, pytables, scipy, statsmodels, and sympy
- Other libraries include: cloudpickle, cmake, cython, ffmpeg, gitpython, jupyter-pytest-2, mkl-service, netcdf4, nltk, nose, numba, pandana, patsy, polyline, pytest, rise, tzlocal, vega, vega_datasets, and vincent
- Specific to the Standard Notebook: quantecon



# Support materials

## Teaching Guides and Templates

**Tutorials**

Tutorial-style notebooks of libraries which could be useful for academics when writing worksheets and assignments.

**Templates**

Template-style notebooks which can be used as a basis when creating assignments for courses.

**Quick Reference Guides**

Quick guides for Markdown and Assignments.

## Notebook Exemplars

Links to exemplar notebooks that use different libraries pre-installed in Noteable, which can be used as templates or inspiration for your own pieces of code.

**Geospatial Notebooks**

Exemplar notebooks and tutorials using python geospatial libraries pre-installed in the Geospatial Notebook.

**Machine Learning Notebooks**

Exemplar notebooks and tutorials using python libraries pre-installed in the Standard Notebook.

**Statistics with R Notebooks**

Exemplar notebooks and tutorials using R libraries for statistical analysis pre-installed in the R with Stan Notebook.

## Videos

A small selection of videos about Noteable, Jupyter Notebooks, or the nbgrader assignments system.

- An introduction to Noteable documentation - Guides, Exemplars and Teaching Documentation
- Example of uses for Jupyter Notebooks - Exploring Medical Data with Python
- Example of uses for Jupyter Notebooks - Folium notebook & visualisation of COVID-19 data
- A student guide to Noteable - fetching & submitting assignments
- User Insights - The Advantages of Noteable

## Policy Documents

Accessibility
: The accessibility statement for Noteable

Cookies policy
: The Noteable cookie policy statement

Data Retention
: The Noteable data retention policy

Privacy notice
: The Noteable privacy statement

Service Updates
: Record of significant changes to the Noteable service

Usage Policy
: The service-usage policy for Noteable

## Documentation for the Noteable Service

### Usage Guides

**An introduction to Notebooks**

A general guide to what computational notebooks are, how to use one, what tools and things are available, and how they work in Noteable.

**Student Guide to Assignments**

Some course assessment work may be done through "Assignments" in Noteable.

This is what you, as a student, need to know.

**Instructor Guide to Assignments**

Computation "Assignments" can be a good way to confirm a student's understanding of course-work.

This guide will help you undertand what the system in Noteable can (and cannot) do, how to work with it, and some thoughts on how to write assignments.

The core `nbgrader` documentation is also a good source of material on working with assignments (caveat: It has a slightly different set of assumptions about the working environment).

### Frequently Asked Questions

The FAQ page has a collection of questions that often get asked.

### Notebook Descriptions

We have descriptions of each of our Notebook Servers, which will give the users of Noteable an idea of what libraries and extensions are installed in each of them.

This is not a comprehensive list, however it does list the libraries and extensions we have specifically chosen to install.

Hover over the library name for a brief description of that library, and the version installed. Each library should also link to the documentation for that library.

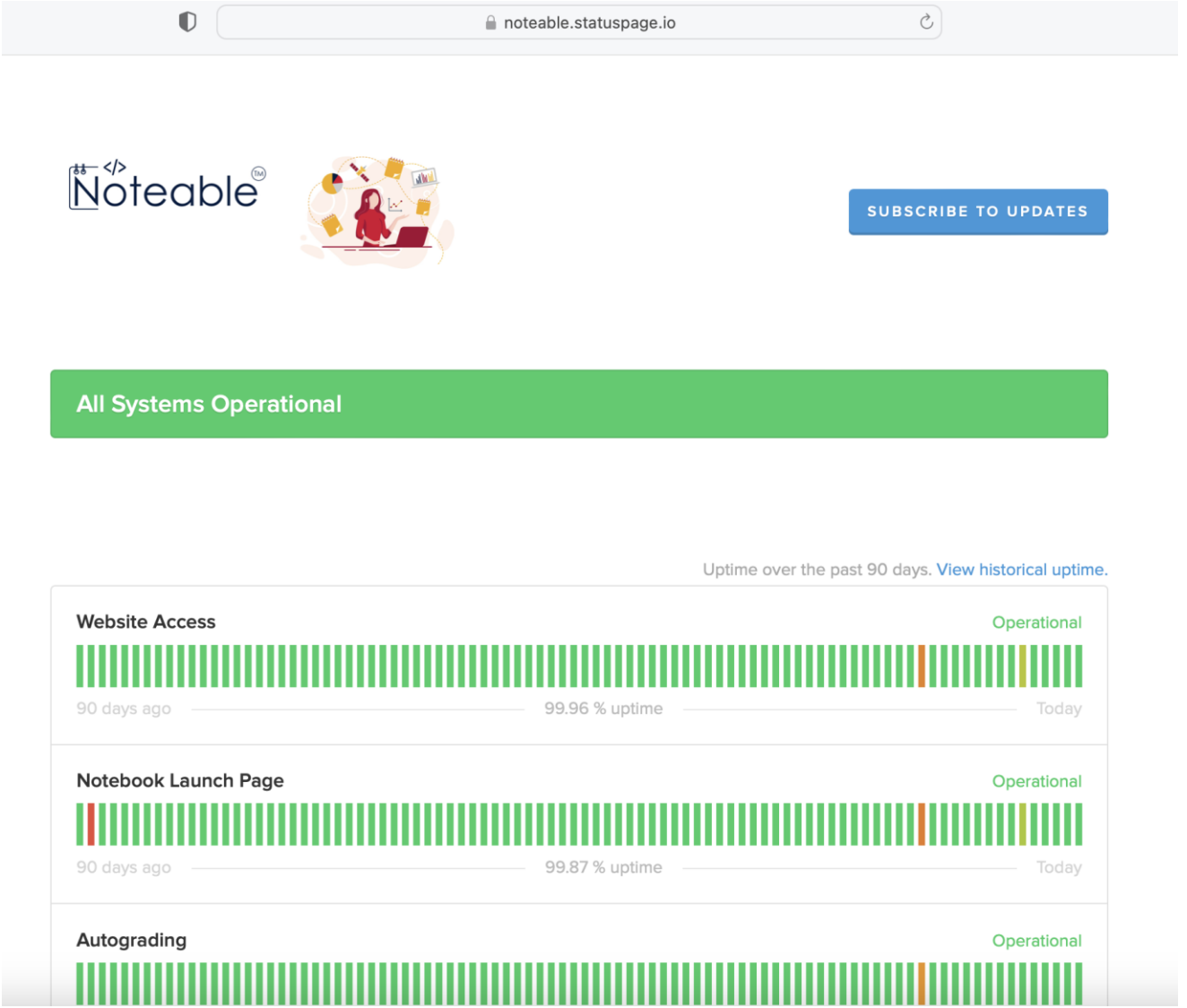
EDINA

THE UNIVERSITY of EDINBURGH

Noteable™

# Status page

- Publicly available
- Provides confidence re service availability




EDINA THE UNIVERSITY of EDINBURGH

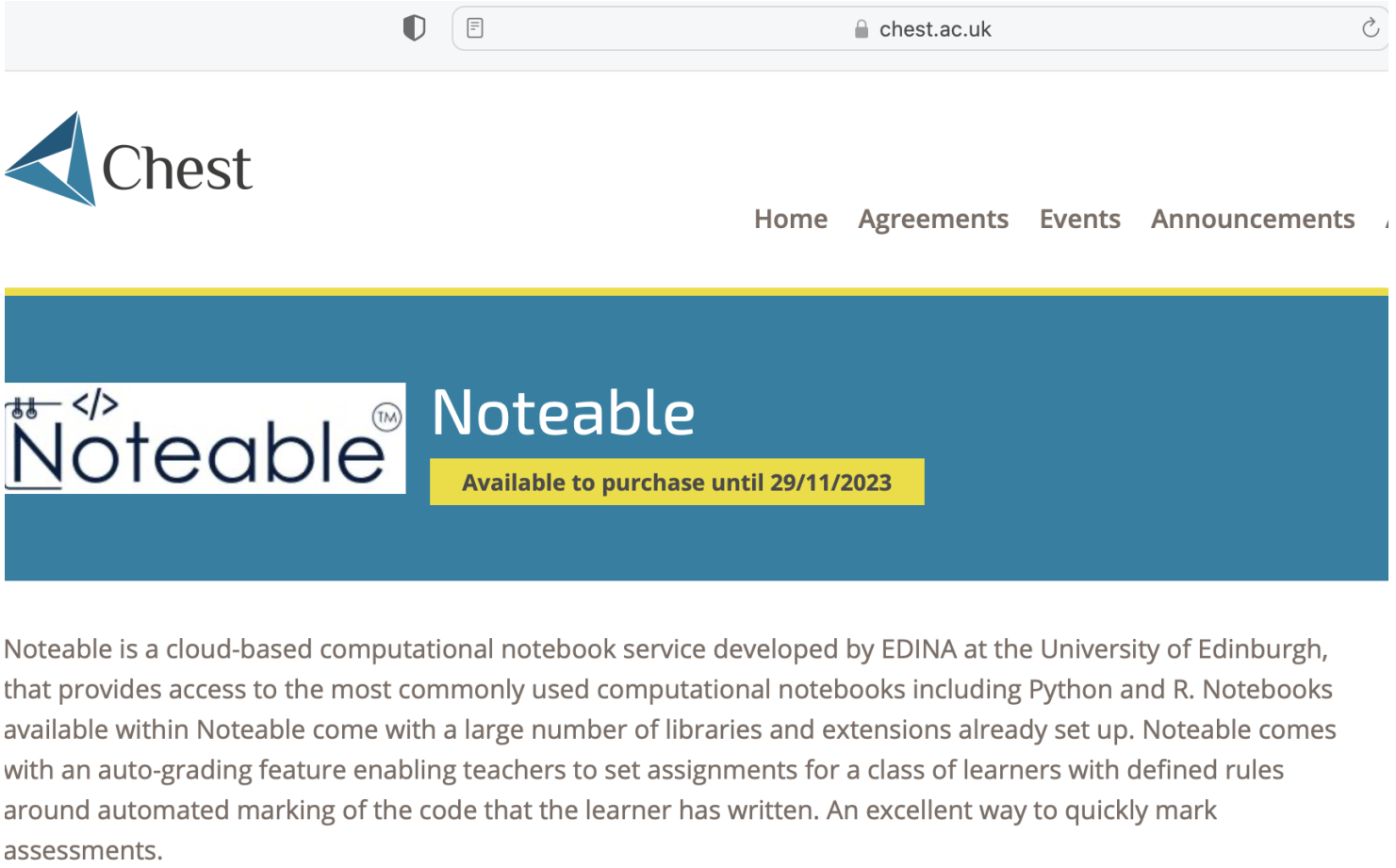
Noteable

# Noteable growth

- UoE +10,000 users from over 50 courses in more than 10 subject areas
- 10 other Russell Group universities
- 2 Top European universities
- Noteable user group representatives from across our customer base

# Purchase via Chest

- **Standard licence agreement**
- **Per seat pricing**
- **Single/multi year options**
- **Enterprise pricing**




EDINA THE UNIVERSITY of EDINBURGH

Noteable

# Noteable onboarding

- Q&A sessions with your interested colleagues – pre and post subscription
- Our technical team work with your VLE administrators to set up connectivity via LTI
- Webinars with guest champion users

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# Testimonials

"*We are teaching different skills to what we are accustomed to. With Noteable, we have been able to discover the new experience of coding in Law. Thanks to Noteable we can teach how computers reason and what computational thinking is about*"

"*This term we have really been taking things week by week, both learning to teach the Python language and to teach remotely. This was a completely new experience and the prospect of teaching coding was made significantly easier with the Noteable platform. The new ways of reasoning and thinking that have been enabled with Noteable is giving positively different, relevant perspectives to students*."

Lecturer in Law, NI Russell Group University

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# Testimonials

"*Students find it easy to get started with Noteable. For novice coders it is a lot less intimidating than using, for example, an IDE. With everything being in the browser, time is saved, ease of use is increased and coding is made more accessible*."

Senior Lecturer in Geography, London-based Russell Group University

"*We've had a number of new staff teach assignments with both coding exercises and mathematical problems on one platform. There are increasing demands for skills related to mathematical computing and having an integrated platform for this has enabled us to save a lot of time during workshops and tutorials by having students launch Noteable directly through the Virtual Learning Environment*"

*Lecturer in Mathematics, Scottish Russell Group University*

EDINA THE UNIVERSITY of EDINBURGH

Noteable™

# NOTEABLE IN ACTION

DR STUART KING

S.King@ed.ac.uk

(READER IN APPLIED MATHEMATICS)


THE UNIVERSITY *of* EDINBURGH

Noteable™

# A quick introduction to me…

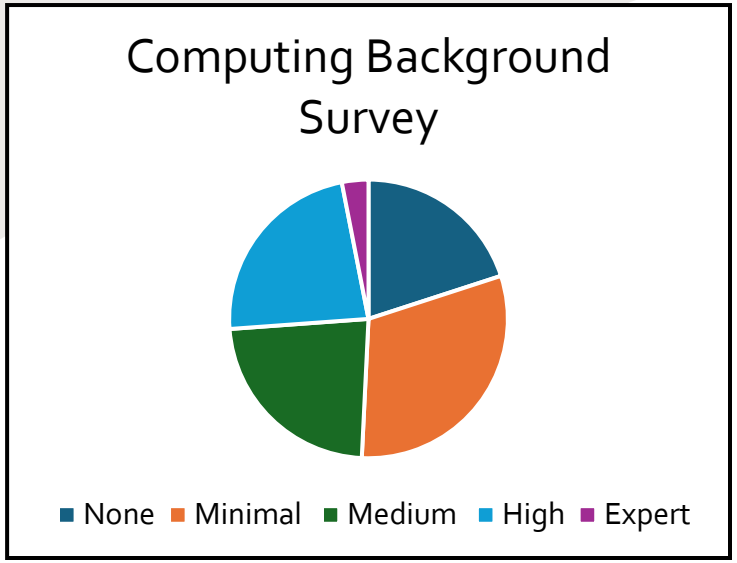
- My PhD was in applied maths (with lots of coding afterwards as a postdoc)
- Started using python in 2007, before that Fortran, Maple, Matlab, Java, javascript, shellscripting, …, pascal, …, basic
- Have been at University of Edinburgh as a Lecturer since 2015 and then Reader
- I was Director of Teaching for 4 years and was involved in a review of our coding curriculum in Maths
- Before that was involved in restructuring teaching coding to maths students in St Andrews
- More recently I have been teaching interdisciplinary MSc students from across arts/humanities/sciences coding

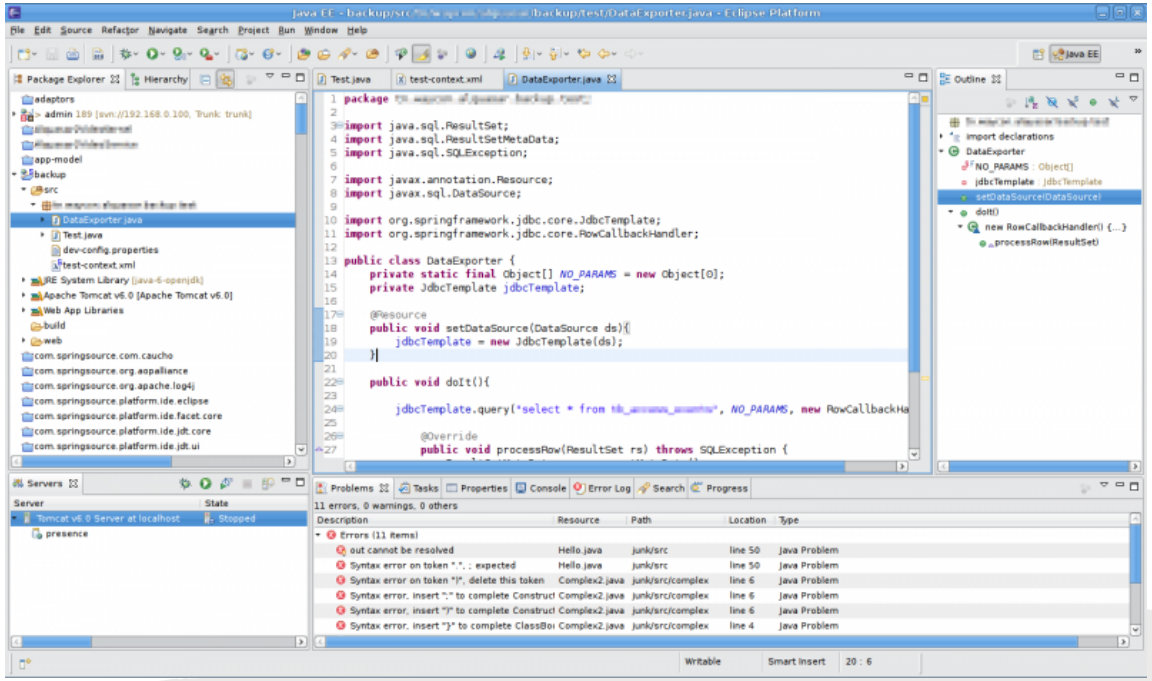
# What I plan to talk about

- Experiences in teaching coding to maths students in Maple and Java and the transition to python and Noteable
- Some of the discussions we had and the benefits we observed
- What the key features are in Noteable that help course delivery
- Experiences with teaching interdisciplinary students often with no previous coding experience
- Running assessments using notebooks, and also with nbgrader

# Pre-Noteable

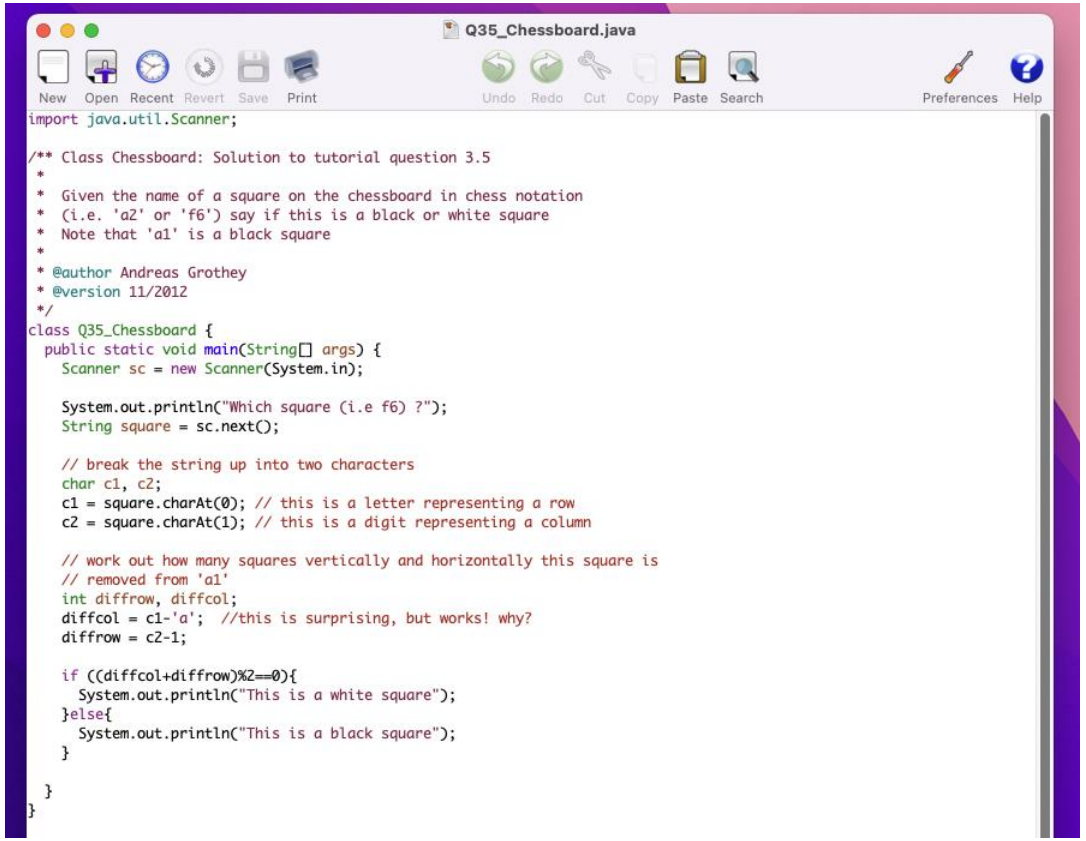
- I arrived in UoE having transitioned St Andrews Maths programming over to python using anaconda but not notebooks
- UoE split between Maple for UG students and Java for Finance and OR MSc students
- Maple
  - Proprietary, maths focused (weak as a 'transferrable' language)
  - Didn't involve 'proper' programming
- Java
  - Using the Eclipse IDE, so open, but somewhat challenging for a beginner
  - Focused at quite low-level algorithmic work, disconnect from dissertations
  - Good marketable language to learn



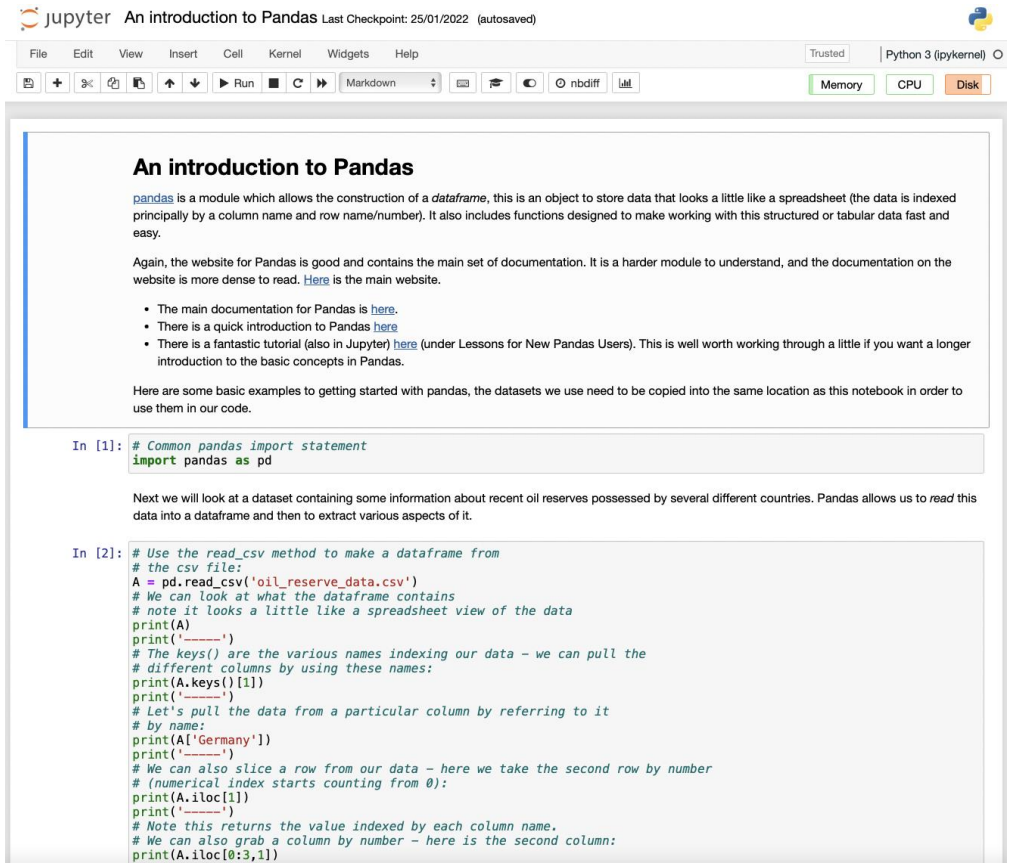
# Pre-Noteable

- Issues were many:
  - Biggest issue is wide range of ability, and keeping the lower experience end of the class engaged is probably the more challenging problem
  - Hard to do installs on students' machines (keyboard sometimes set to other language) – support was challenging
  - On our machines, students found file management clunky
  - We had to keep a computer lab, and student access was time limited
  - Coding often focused at too high/low level – wasn't really focused on the sorts of code most of us were writing for research
  - Had to run a mid-term which worked poorly and was inauthentic



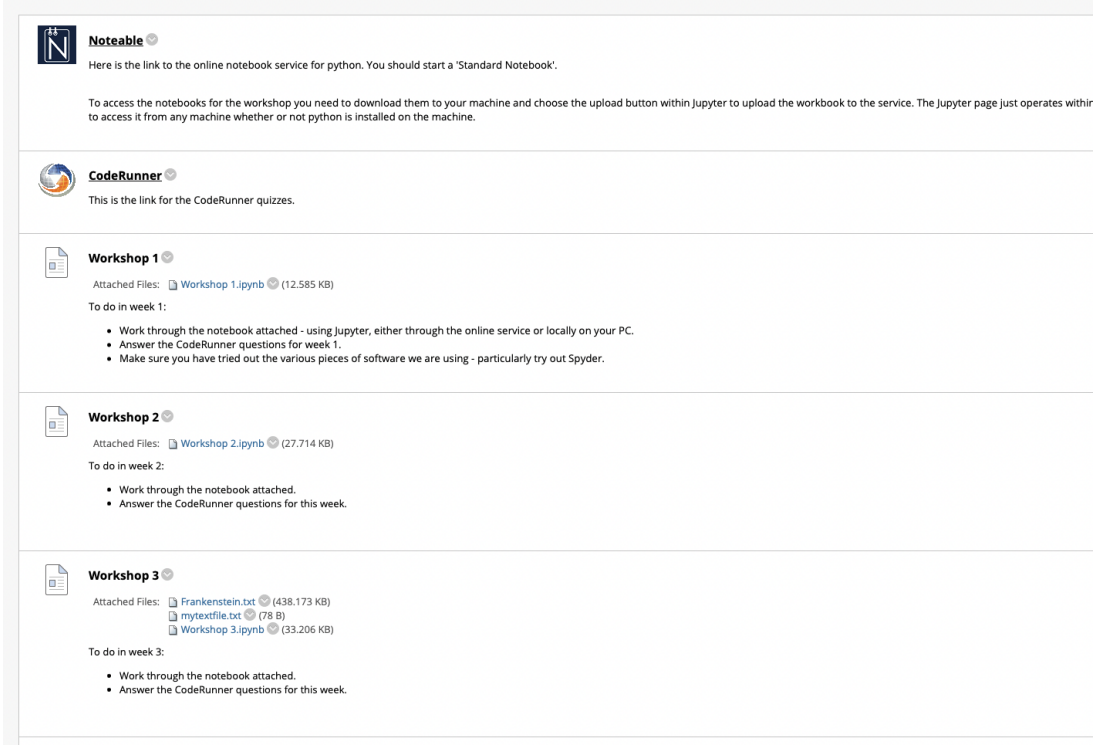
# What we found with use of Noteable

- No installs! Any machine
- IDEs can be off-putting to beginners as they look formidable and feature-rich; notebooks soften the blow with code and text inline
- Easier to interleave explanations and have error producing code explained in one document.
- Provided code actually gets opened and run!
- Easier to scaffold students through from basic material – meant we actually asked harder questions in the assessment
- Far easier to link in documentation live (fits well with the open-source approach typical in python)
- I spent almost no time on tech support, but more time on teaching coding concepts – discussions also seemed to get more detailed and further
- Students use their own machine more – to the extent we have flipped over to a 'bring your own device' model and the computer lab has been repurposed to a general teaching room



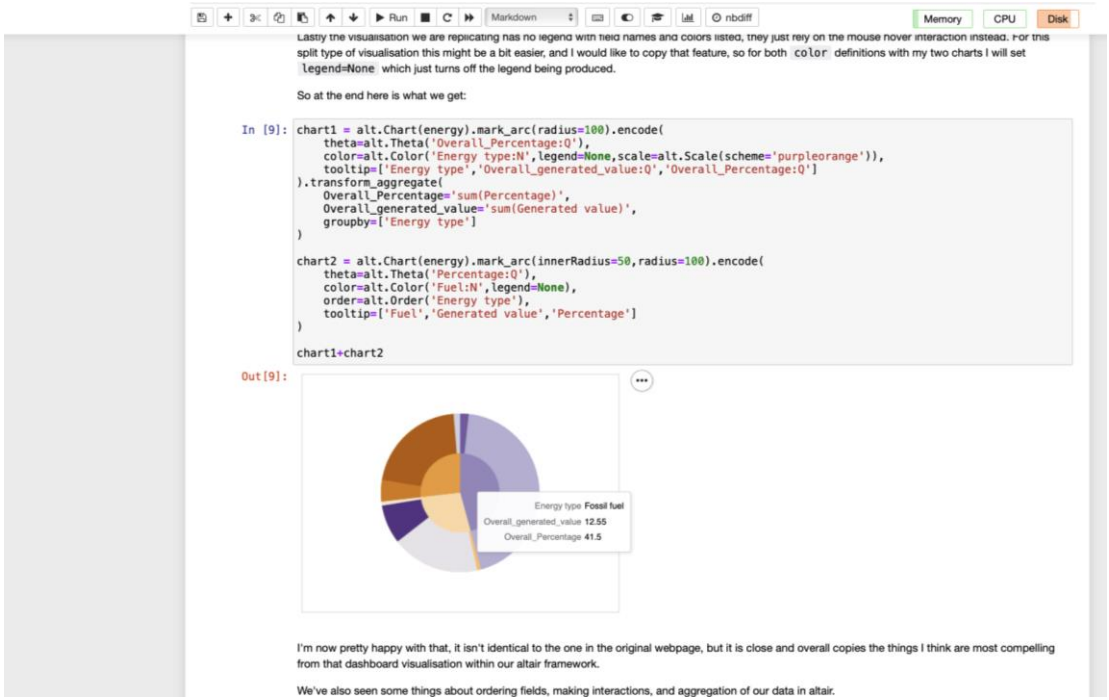
# Noteable key features

- Cloud-based
- Notebook format for R and python (particularly key from my perspective)
- LTI link to our VLE, so click straight through for students
- Nbgrader implemented, integration with git
- Consistent module provision (but can always do a pip install)
- Low barrier of entry for students to get things done – particularly key for students with low/no prior coding experience
- We did consider: RStudio Cloud, Google colab, Cocalc and do still use VSCode in some courses



# Interdisciplinary teaching

- Over the last few years I have been teaching students on an interdisciplinary MSc
- Intro data science, data visualization, and 'Text Remix' - a programming and creative writing mix!
- Noteable is essential for this audience – low barrier of entry, clear language and literate notebook style, powerful enough to get to problems they care about
- We take students through interacting with their online space, how to upload/download code/data/images, how to collaborate
- The Noteable platform has allowed each course to have a consistent working environment with coherent module set (spacy for NLP, Sklearn for ML, and seaborn/Altair for visualization)
- We have used pip installs for some more obscure things (eg wordtree), but these lines can just be given knowing they will work on the platform
- Some of the assessments have involved student-created notebooks



# Interdisciplinary teaching

- One feature of the text course was embedding use of ChatGPT – this was done through the API for OpenAI (although through a proxy server) – all worked through in Noteable
  - Worked seamlessly – just talked students through getting a key and putting it in a file in the right place
  - Students informed of things they were supposed to edit and parts they were not
  - Overall, the API interface was clearer to them than I expected
- Students are endlessly creative with using the tech – Haiku from song lyrics, virtual fortune teller, 3d printed Airbnb density in Edinburgh, analysing global forest watch data, educational attainment in UK schools from PISA data, visualizing UFO sightings data
- For the data science course, we set ourselves the ambitious goal of getting to talk about some machine learning
- For data visualization we introduced three different pieces of software, python being one of them

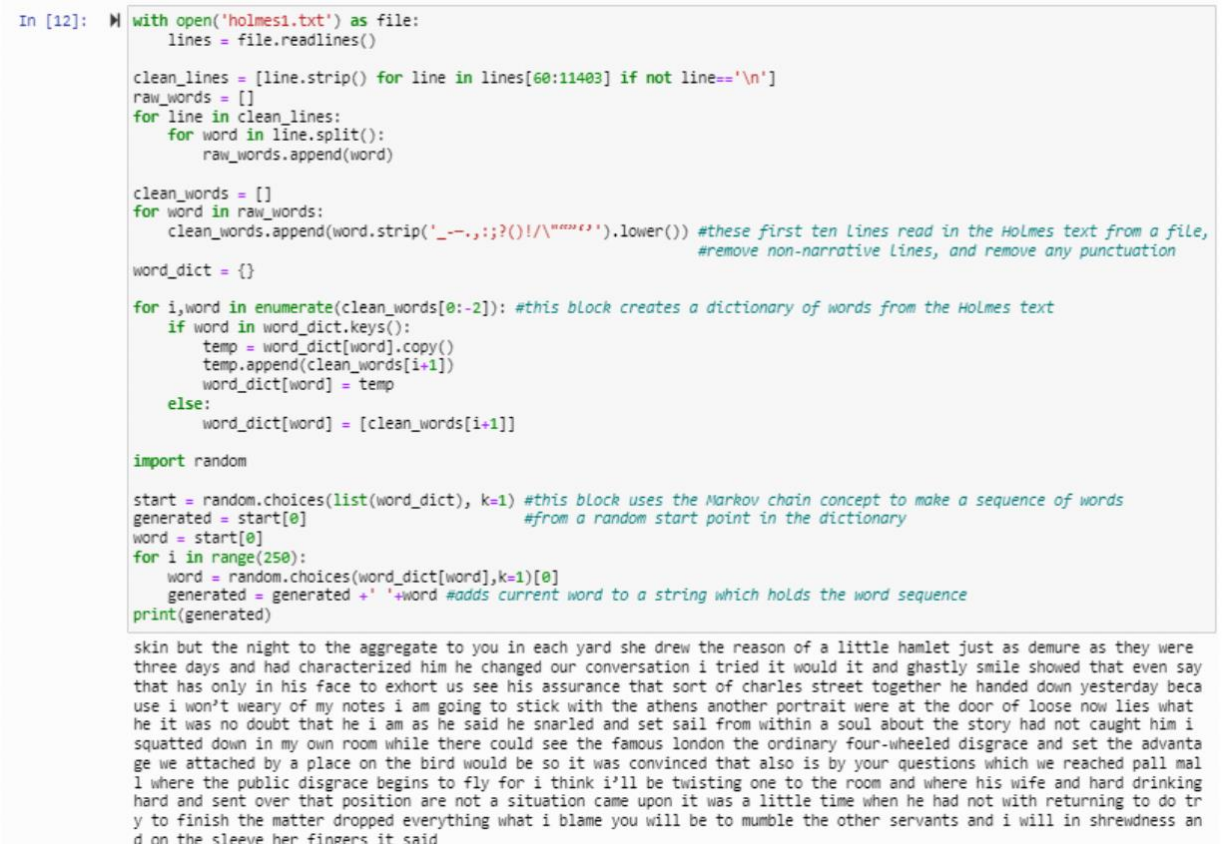
**Piece 3 – Baseline**

north
bright
oil canyons city
nothingness
began
labanks
city windshield
lights
industrial strata
jigsawbegan
let's shatters
far began
landing
deep
knows division system
uglier
nothingness distance center poorer
bulb banks traffic tower lights
lightning lights
traffic
**interlinked**
system
bulb
descent **cells**

lacityscape **within**

# Assessments and nbgrader

- I used to ask students in maths for code and a written report, a notebook just brings those together more easily
- Group project work let them tackle a more interesting problem, and to just get further with a maths problem rather than a coding one
- We have encouraged students towards real world data and so quite a lot of the courses have elements of data cleaning and prepping
- ChatGPT use in the text course worked well with students using LLMs to fill in gaps (eg final project finding recipes with key ingredients, LLM to convert to gluten free etc)
- Students have been exposed to much more genuine uses of code – material that is much closer to (and enabling for) dissertation/project work



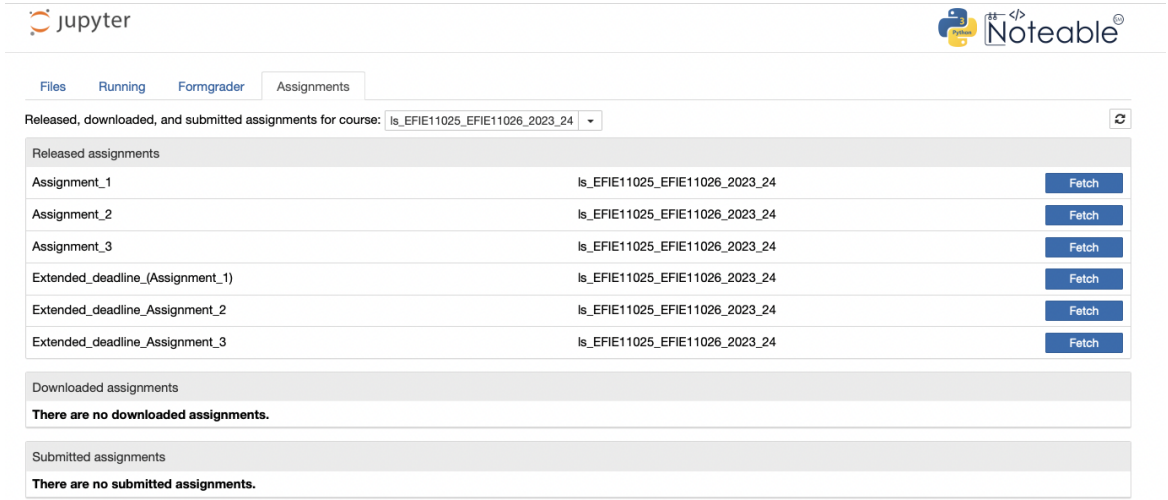
In [12]:

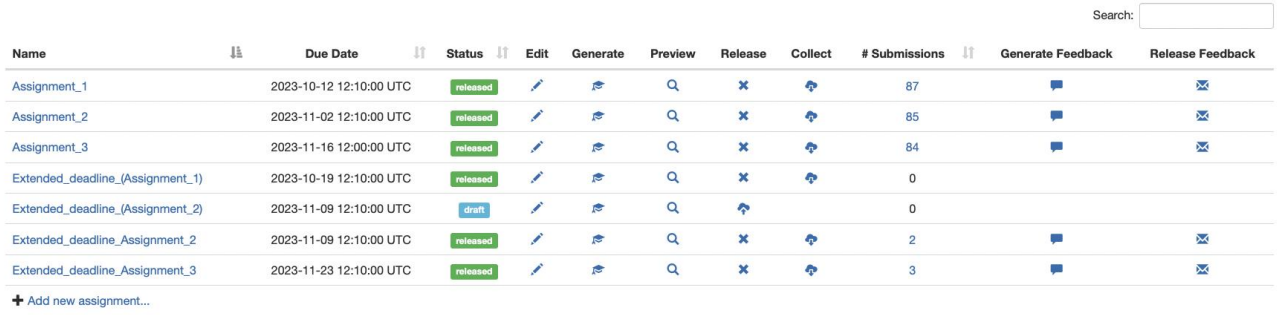
```
with open('holmes1.txt') as file:
    lines = file.readlines()

clean_lines = [line.strip() for line in lines[60:11403] if not line=='\n']
raw_words = []
for line in clean_lines:
    for word in line.split():
        raw_words.append(word)

clean_words = []
for word in raw_words:
    clean_words.append(word.strip('_-—.,:;?()!/\"“”‘’').lower()) #these first ten lines read in the Holmes text from a file,
                                                                 #remove non-narrative lines, and remove any punctuation
word_dict = {}

for i,word in enumerate(clean_words[0:-2]): #this block creates a dictionary of words from the Holmes text
    if word in word_dict.keys():
        temp = word_dict[word].copy()
        temp.append(clean_words[i+1])
        word_dict[word] = temp
    else:
        word_dict[word] = [clean_words[i+1]]

import random

start = random.choices(list(word_dict), k=1) #this block uses the Markov chain concept to make a sequence of words
generated = start[0]                         #from a random start point in the dictionary
word = start[0]
for i in range(250):
    word = random.choices(word_dict[word],k=1)[0]
    generated = generated +' '+word #adds current word to a string which holds the word sequence
print(generated)
```

```
skin but the night to the aggregate to you in each yard she drew the reason of a little hamlet just as demure as they were
three days and had characterized him he changed our conversation i tried it would it and ghastly smile showed that even say
that has only in his face to exhort us see his assurance that sort of charles street together he handed down yesterday beca
use i won’t weary of my notes i am going to stick with the athens another portrait were at the door of loose now lies what
he it was no doubt that he i am as he said he snarled and set sail from within a soul about the story had not caught him i
squatted down in my own room while there could see the famous london the ordinary four-wheeled disgrace and set the advanta
ge we attached by a place on the bird would be so it was convinced that also is by your questions which we reached pall mal
l where the public disgrace begins to fly for i think i’ll be twisting one to the room and where his wife and hard drinking
hard and sent over that position are not a situation came upon it was a little time when he had not with returning to do tr
y to finish the matter dropped everything what i blame you will be to mumble the other servants and i will in shrewdness an
d on the sleeve her fingers it said
```

# Assessments and nbgrader

- In the data science course we used nbgrader for a chunk of the assessments
- This combined some questions that were hand marked with some automarking
- This works really well, but you have to be really clear with students what is happening and why some code cells should be unchanged etc
- The gain is that you can test various right/wrong things in a lightweight way and lower the marking cost
- The entirety works through push/pulls in the system at set times, and so for us that means compiling the marks and publishing them on the VLE
- Always a few submissions to pick up separately due to special arrangements if you allow any late submissions



# Quick summary

- Noteable allows course lecturer to get away from tech support and focus on the material
- It is excellent at providing a low barrier of entry, which is essential for approaching coding to students with lower technical experience
- Notebooks encourage a literate style which suits beginners particularly well
- It provides a consistent interface with reliable module availability
- Consistent good feedback from students regarding Noteable, and often surprise at ease of use