# Topic Exploration Pack

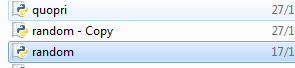
# Applications Generation

# Activity 1: hacking Python

One of the unique features of Python is that it can be extended on the fly by its users. It takes the concept of open source to the point where most files that make up Python can be opened in a text editor (e.g. IDLE or Notepad) and edited.

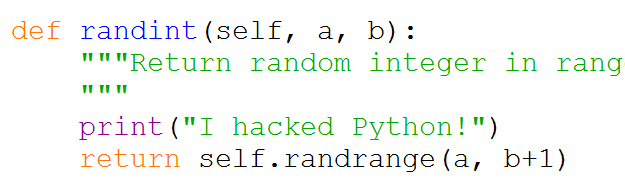
Screenshot of Python installation directory


Task A: Go to the python installation directory (e.g. C:\Python33\Lib), find a Python module called ‘random’ and make a copy of it (you can copy the file and paste into the same folder with a different name).

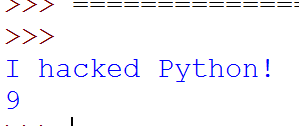
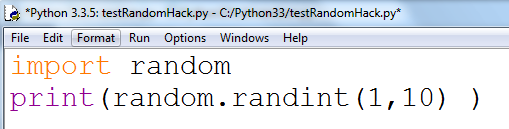


Now, open the module and find the function randint; feel free to admire how professional Python code is written and the amount of comments the author put in there.

Add the line about hacking Python just above the return statement:



Save the file and execute the program given below.



Assuming you worked with Python what things would you change about it to suit your working style? (No, you can’t have multi-coloured text in IDLE console.)

Would your programs work if you sent them to another person?

Delete your modified random.py file and replace it with the copy you had made prior to making the changes.

Task B: Open a few more Python module files from this folder, and based on the comments inside, describe their functionality (e.g. the web browser module), and then create a Python program that will import and use each of these modules.

# Activity 2: finding out about software licenses

Consult the GNU site (<https://www.gnu.org/licenses/license-list.html>) and the Free Software Foundation site (<http://www.fsf.org>) for a concise list of different, free and open source licenses available.

Task A: Browse the list of free software on this site: <http://directory.fsf.org/wiki/Main_Page>. Do you recognise any of the software listed there? If not, why?

Task B: Create a list of software used by three other pupils, including mobile apps. Create a questionnaire similar to the one below, and for every application, note its license (the information can be obtained off the author’s website or Wikipedia).

Name \_\_\_\_\_\_\_\_\_\_\_\_\_

Main computer type/model/OS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Main mobile type/model/OS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| **Commercial applications/apps used** | **GNU applications/apps used** | **Other software licenses used** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Activity 3: compiled and interpreted languages popularity

Using <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html> or a similar programming language popularity site, find out for each language in the top 10 whether it is compiled or interpreted.   
For each language, write/research why it made sense for the language to go with this type.

|  |  |  |
| --- | --- | --- |
| **Language** | **Compiled or Interpreted** | **Most used for…** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Activity 4: organise a debate

Is Android operating system an open software or not? Resources to start:

<http://www.infoworld.com/article/2623095/android/proof-android-is-not-open-source----and-why-that-s-good.html>

<http://source.android.com/>

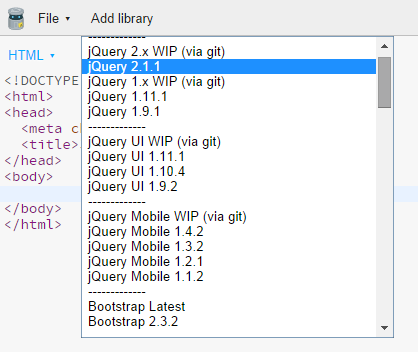
<http://arstechnica.com/gadgets/2013/10/googles-iron-grip-on-android-controlling-open-source-by-any-means-necessary/>

Note down the arguments used on either side in the table below:

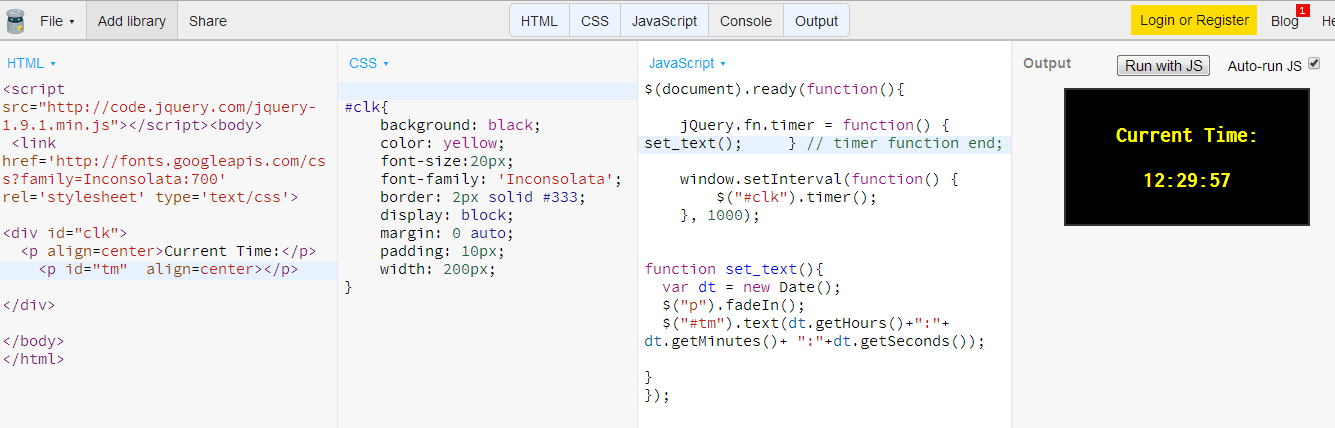
|  |  |
| --- | --- |
| **Android is open source** | **Android is not open source** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Activity 5: on JavaScript open source libraries

Using a WYSIWYG editor that supports open source libraries, such as jsbin.com, implement a   
program that uses at least two different JavaScript libraries, jQuery and Video.js.

*Topic Exploration Pack *

Task A: Implement a program with jQuery:



Task B: Implement any other library. Here we use Paper.js (<http://paperjs.org/examples/candy-crash/>) for an example.

