What Works
In Teaching Python
Toolkit Contents

- PRIMM
- Concept Before Code
- Worked Examples with Subgoal Labelling
- Project-based Learning
- Tell a Story with the Code
PRIMM is a structure that you can use to plan programming lessons. It is useful for programmers at different experience levels who are learning a new concept.

Learn More
PRIMM

Example:
- Replit
- Intro to Python
- Teaching Curriculum

Created by Andrew Colley (@mracolley)
Programming – Output – Predict & Run

5     #Task 1 - Add a comment on line 7 to predict what the code on line 8 will do.

6

7

8     print("Hello World!")

Add comments to the code to predict exactly what it will output.
Run the code to see if you were correct.

https://repl.it/@MrAColley/11-Output

Resources created by Andy Colley (@MrAColley)
Programming – Output – Investigate

```python
print("Hello World!")

# Task Investigate

# What would the output of the code `print("I love Computing")` be?

# What would happen if the code `print("I love Comping")` was run?

# What would happen if the code `print("I love Computing" was run?)
```

Add comments to the code to answer the questions.

https://repl.it/@MrAColley/11-Output
Modify – reuse the print statement to add your own single line message to the program.

Make – use the print statement to output a joke that appears on multiple lines (keep it clean!). Extra challenge – can you figure out how to add blank lines like in my joke?

What's orange and sounds like a parrot?

A carrot!

>
Concept Before Code

Concept Before Code is a strategy that involves exposing learners to a new programming concept through non-programming activities prior to using it in code for the first time.

Learn More
1. Define the concept.
2. Share an analogy related to learners’ background knowledge.
3. Engage learners with unplugged or digital non-programming activities.
4. Show the Python code that represents the concept. Explicitly relate it to the prior activity.

Example:

Code-IT

Supporting Algorithmic & Programmatic Thinking

Resources

Created by Phil Bagge (@baggiepr)
Everyday variables Visitor Car Parking Space

Visitor

Name

Value (Red car)

Monday
Everyday variables Visitor Car Parking Space

Visitor

Name

Value (Yellow car)

Tuesday
Everyday variables Visitor Car Parking Space

Tell your neighbour how the value of the variable changed every day
Monday Red car
Tuesday Yellow car
Wednesday Empty
Worked Examples with Subgoal Labeling

Worked examples demonstrate programmatic solutions. They usually consist of a problem statement or goal and an example program or code snippets that represents the solution. When subgoals are labelled, the programmatic solution is broken down into discrete steps or chunked by functionality and these steps or parts are explicitly labelled.
Worked Examples with Subgoal Labeling

Examples:
Improving Engagement in Program Construction Examples for Learning Python Programming


Subgoals Help Students Solve Parson’s Problems

Fig. 1 A Python programming worked example in the PCEX activity. The example includes the goal (A), interactive worked code (B), the subgoal label presented as a comment (C), the link to instructional explanations (question mark symbols) (D), explanations (E), a navigation link to the explanation for the previous/next line (F), additional details for the highlighted line (G), and a challenge navigation link (H).
<table>
<thead>
<tr>
<th>No labels</th>
<th>Given Labels</th>
<th>Generate Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>sum = 0</td>
<td>Initialize Variables&lt;br&gt;sum = 0&lt;br&gt;lcv = 1&lt;br&gt;Determine Loop Condition&lt;br&gt;WHILE lcv &lt;= 100 DO&lt;br&gt;sum = sum + lcv&lt;br&gt;lcv = lcv + 1&lt;br&gt;ENDWHILE</td>
<td>Label 1:&lt;br&gt;sum = 0&lt;br&gt;lcv = 1&lt;br&gt;Label 2:&lt;br&gt;WHILE lcv &lt;= 100 DO&lt;br&gt;Update Loop Var&lt;br&gt;lcv = lcv + 1&lt;br&gt;ENDWHILE</td>
</tr>
</tbody>
</table>

**Figure 1.** Partial worked example formatted with no labels, given labels, or placeholders for generated labels.
Choose Code that Makes Processes Visible

When there are multiple ways to meet a goal, professional software developers often choose the option that is the most efficient to run and/or to write, which usually utilizes abstraction. In education, efficiency isn’t the priority. Rather making abstract concepts explicit and visible is. Let the code “tell the story.”
Which print statement tells the clearest story?

```python
name = "Luna"
age = 10

print('My name is', name, 'and I am', age, 'years old.')
print(f'My name is {name} and I am {age} years old.')
print('My name is ' + name + ' and I am ' + str(age) + ' years old.')
print('My name is {} and I am {} years old.'.format(name, age))
print(' '.join('My name is', name, 'and I am', str(age), 'years old.'))
```
Which code snippet tells the clearest story about how the data in the table was obtained?

```python
import urllib2
response = urllib2.urlopen('https://www.codesters.com/api/SpriteImage/?format=json')
html = response.read()
```

Opening the Magic Box, Farfan, Ray, & Ricard (2016)
Project-Based Learning

Project-Based Learning (PBL), puts learning in the context of a project that is personally meaningful to learners. Project-building is iterative and students are assessed based on multiple aspects of the final version of their project. When projects have a tangible component to them, it is often even more effective.

Learn More - general PBL

Learn More - PBL for CS

Examples: PyGotham Young Coders, Pycon UK Education Summit, Big Red STEM Day
Do you have suggestions for strategies that you’d like to see added to this toolkit? Send them to education@python.org

Content by Meg Ray

CREDITS: This presentation template was created by Slidesgo, including icons by FlatIcon, and infographics & images by Freepik

Please keep this slide for attribution