

# ET-PoC-Study

## *Post-Questionnaire*

- How easy/hard did you experience the program comprehension tasks?

Very Easy	Easy	Neutral	Hard	Very Hard
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Optional Comment:

- eher einfach, besonders wenn noch mehr Info für str und keine Ahnung mehr was doppelpunkt bei List ist

- How do you feel after the study?

Very Exhausted	Exhausted	Neutral	Energetic	Very Energetic
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Optional Comment:

/

- You are presented with the displayed code snippets. Which of the excerpts did you find particularly easy/difficult? Why?

Please annotate them on the pages themselves.

- How do your problem-solving strategies change with the code snippets?

- manche waren direkt ersichtlich, bei schwierigen, ein Beispiel durchgehen für konkrete Werte
  - nach kurzem Überlegen Algorithmus erkannt

5. Did you detect any changes in yourself in comprehending the code snippets?  
If so, what did this change look like?

/

6. Did **Type Annotations** (`number: int` instead of `number`) help you comprehend the code snippets?

Yes	No
<input checked="" type="checkbox"/>	<input type="checkbox"/>

How did they help? Please elaborate.

- zum einen hat es Denkarbeit weggenommen
- Python eine Sprache, bei der man mit Typen um sich werfen kann
- Python kann sehr viel und bei Zahloperationen ist es sehr wichtig den Typ zu wissen

7. Beyond this study, do Type Annotations help you comprehend code in general?

Yes	No
<input checked="" type="checkbox"/>	<input type="checkbox"/>

Why? Please elaborate.

- generell für den leichteren Überblick
- besonders wenn es COde ist mit dem man weiterarbeitet
- besonders wichtig input und output, damit man ein robustes Programm hat

8. Were earlier or later code snippets easier for you to comprehend?

- die früheren waren einfacher

9. Do you have any suggestions or ideas for improvement for us?

- kleine Referenz von Standardfunktionen wäre super
- noch größere Diversität der Schwierigkeit, bis auf eine Aufgabe sicher bei allem