

THE ROLE OF ENGLISH IN ADAPTING THE PYTHON PROGRAMMING LANGUAGE TO THE UZBEK EDUCATIONAL AND TECHNOLOGICAL MODEL

Rahmatullayeva Barchinoy Xikmatullo qizi

First year student at TIAME NRU

[*rahmatullayevabarchinoy1@gmail.com*](mailto:rahmatullayevabarchinoy1@gmail.com)

+998334941931

Boltaboyeva Charosoy Muzaffar qizi

First year student at TIAME NRU

[*charosoyboltaboyeva0107@gmail.com*](mailto:charosoyboltaboyeva0107@gmail.com)

+998943060131

Scientific Advisor: Karimov Feruzjon

Teacher, English language Department

TIAME NRU

Abstract: This study explores the pivotal role of the English language in adapting the Python programming language to the Uzbek context, particularly in education, research, and software development. Drawing on international practices and comparative analysis, the paper highlights how English serves not only as a medium of instruction but also as a cultural and technological bridge for effective coding education. The study argues that without English language proficiency, full utilization of Python's ecosystem, including libraries, documentation, and community support, remains limited.

Keywords: English language, Python programming, localization, digital education, Uzbekistan, coding literacy

1. Introduction

Python is recognized globally as one of the most beginner-friendly yet powerful programming languages, favored for its simplicity, readability, and diverse applications ranging from data science to artificial intelligence and web development (Van Rossum & Drake, 2009). In recent years, the government of Uzbekistan has emphasized digital transformation through national programs such as the “Digital Uzbekistan–2030” strategy, promoting the integration of modern programming

languages, including Python, into general education and higher education curricula (Ministry of Digital Technologies, 2022).

However, this integration effort faces a significant linguistic barrier. Python, like most programming languages, is fundamentally rooted in English—not only in its syntax (e.g., keywords like `if`, `else`, `import`, `def`) but also in its extensive documentation, error messages, community forums, open-source libraries, and learning resources such as tutorials and MOOCs (Massanari, 2015; Guo, 2021). For non-English speaking learners, especially those in developing nations, this presents a dual challenge: mastering both programming logic and a foreign language simultaneously.

This intersection between language proficiency and computational literacy has drawn attention from educators and scholars worldwide. Pérez-Marín & Pascual-Nieto (2010) highlight that natural language plays a foundational role in shaping a student's ability to think algorithmically. Similarly, Lee and Hsu (2019) emphasize that in multilingual environments, programming languages are not neutral tools but are embedded within dominant linguistic ideologies—most often, English.

In the context of Uzbekistan, where English is still being established as a second language and not widely spoken outside urban centers, the dependence on English for learning Python can create accessibility gaps. As Shodiev (2020) notes, the gap in English literacy between urban and rural students contributes to unequal access to digital opportunities. Therefore, while the push for digital education is commendable, it must be accompanied by strategic efforts to integrate English learning with coding instruction.

2. Methods

This study employs a mixed-methods approach, combining both quantitative and qualitative data to investigate how English proficiency influences the effective acquisition and application of Python programming skills among Uzbek university students. The rationale behind this methodology is grounded in previous research that

underscores the socio-linguistic challenges faced by non-English speakers in programming environments (Guo, 2021; Lee & Hsu, 2019).

2.1 Literature Review

A comprehensive literature review was conducted, examining peer-reviewed journal articles, educational reports, and case studies from countries with similar linguistic and educational contexts, such as China, Turkey, Indonesia, and Brazil. These studies provided a comparative foundation for understanding the role of English in programming education outside the Anglosphere. Key themes included language-based cognitive barriers, the effectiveness of localized content, and the outcomes of bilingual programming instruction.

2.2 Online Student Survey

A structured online survey was distributed to 85 first-year undergraduate students from the ICT and Software Engineering departments at TIAME NRU and INHA University in Tashkent. The survey included Likert-scale and open-ended questions designed to assess the following:

Students' self-reported English proficiency levels (reading, writing, comprehension).

Frequency of using English-based vs. Uzbek-based learning resources.

Perceived difficulties in understanding Python's syntax, documentation, or error messages due to language limitations.

Preference for instruction language in programming classes. The results were analyzed using descriptive statistics and thematic coding for qualitative responses. This helped identify key trends in student experiences and perceptions.

2.3 Instructor Interviews

Semi-structured interviews were conducted with three computer science instructors who currently teach Python in both Uzbek and English. These interviews focused on:

- The instructors' observations regarding student performance based on English proficiency.
- Pedagogical strategies for teaching programming bilingually.

Challenges in using Uzbek-translated materials. Interview responses were transcribed and analyzed through thematic coding, emphasizing language-related instructional barriers.

2.4 Textual Analysis of Learning Materials

A side-by-side textual comparison was performed between original English Python tutorials and their Uzbek-translated counterparts available from local publishers and educational portals. This analysis examined translation accuracy, clarity of programming logic, and the conceptual fidelity of key Python concepts (e.g., loops, data structures, error handling). Particular attention was given to how English terms were adapted or replaced in Uzbek and how that affected learner comprehension.

2.5 The Role of English as a Research Variable

Importantly, English was not just a medium in this study—it was also an independent variable influencing learning outcomes. As highlighted by O'Reilly (2018), programming cannot be divorced from its linguistic medium, especially when that medium is not the learner's first language. Therefore, the presence or absence of English proficiency was treated as a critical factor shaping student engagement, resource accessibility, and problem-solving capability in Python. The methodological design aimed to holistically capture the multi-layered impact of English language skills on Python programming within Uzbekistan's educational and technological context. Through triangulating data from students, instructors, and materials, the study ensures both reliability and depth in its findings.

3. Results

3.1 Survey Insights:

- 92% of students reported difficulty understanding Python errors or documentation without English support.
- 68% rely heavily on English YouTube tutorials or Stack Overflow answers.
- Only 23% felt that current Uzbek-language programming resources are sufficient.

3.2 Instructor Feedback:

- Teachers noted that students with intermediate English levels grasp Python logic faster and complete tasks with greater independence.
- Lack of English literacy delays debugging and self-learning.

3.3 Textual Comparison:

- Uzbek-translated materials often fail to capture the nuance of Python's logic or technical instructions.
- Literal translations sometimes confuse learners (e.g., translating append as "ilova qilmoq" rather than explaining list operations contextually).

3.4 Global Comparison:

- Countries like Turkey and Indonesia have launched bilingual Python platforms but still emphasize English as the "native language of programming" (Lee & Hsu, 2019).

4. Discussion

The results of the study underscore a critical reality: English language proficiency plays a foundational role in the process of learning Python programming among Uzbek students. The findings from the student survey revealed that more than 60% of participants experienced difficulty in understanding error messages, official documentation, and community forums due to insufficient English reading skills. These challenges echo the observations of Lee and Hsu (2019), who argue that programming is not only a technical activity but also a linguistic one.

Instructors interviewed in this study confirmed that students with higher English proficiency showed greater confidence in exploring advanced Python libraries, troubleshooting code, and participating in online developer communities. This aligns with research by Guo (2021), which found that non-native English speakers often rely on translated resources, limiting their exposure to the latest developments and nuanced explanations in original English materials.

Interestingly, the textual analysis showed that many Uzbek-translated tutorials oversimplify or omit critical programming concepts, particularly in areas such as object-oriented programming and exception handling. This dilution may be

unintentional but has significant pedagogical consequences, reinforcing the importance of dual-language instruction rather than complete translation. Moreover, the qualitative responses from students highlighted a sense of frustration when dealing with English-dominant learning platforms like Stack Overflow or GitHub. While these platforms are rich with resources, the language barrier limits active engagement. As Pérez-Marín and Pascual-Nieto (2010) argue, the inability to comprehend technical English creates a form of "digital exclusion," wherein students are present in digital environments but cannot participate meaningfully.

The discussion also brings attention to equity concerns. Students from rural backgrounds or public-school systems, where English instruction is often less intensive, face steeper learning curves compared to peers from elite urban schools. This reinforces the need for contextualized pedagogy that addresses both computational and linguistic scaffolding.

In sum, the role of English in learning Python in Uzbekistan is not peripheral—it is central. It influences how students access information, solve problems, and imagine themselves as future contributors to the global tech ecosystem.

5. Conclusion

This study set out to examine the role of the English language in adapting the Python programming language to Uzbekistan's educational and technological context. The findings provide compelling evidence that English proficiency significantly impacts students' ability to effectively learn and apply Python. Through mixed methods—including surveys, interviews, and textual analysis—it became clear that English functions not merely as a programming syntax but as a gateway to the broader ecosystem of coding resources, collaboration platforms, and documentation. The reliance on English, while logical from a global perspective, presents a systemic challenge for Uzbek learners who are still in the process of acquiring the language.

To address this gap, the following recommendations are proposed:

Integrative Curriculum Design: English for Specific Purposes (ESP) courses should be tailored to include programming terminology and documentation literacy.

Bilingual Teaching Materials: Instead of full translations, materials should be designed with parallel English–Uzbek sections to preserve technical precision while enhancing comprehension.

Instructor Training: Professional development for Python instructors should include language-sensitive teaching strategies, especially for mixed-ability groups.

Localization of Tools: Development of language plugins or localized interpreters for beginner-level learners may reduce initial barriers.

Ultimately, for Uzbekistan to fulfill its digital transformation goals under the "Digital Uzbekistan–2030" initiative, it must address the linguistic challenges inherent in modern programming education. This means viewing language not as a secondary concern but as a core component of digital literacy.

References:

1. Guo, P. (2021). Python and the Language of Programming: A Sociolinguistic Analysis. *Journal of Digital Learning*, 29(2), 101–114.
2. O'Reilly, J. P. (2018). *Programming Beyond Syntax: Why Language Matters in Code*. O'Reilly Media.
3. Pérez-Marín, D., & Pascual-Nieto, I. (2010). *Natural Language in Programming Environments: A User-Centered Approach*. Springer.
4. Atkinson, R. K. (2020). Language, Code, and Cognition: The Role of English in Computing Education. *Educational Technology Journal*, 45(3), 198–212.
5. Wang, Y., & Chen, H. (2019). Bridging the Gap: Teaching Python in ESL Contexts. *Journal of Applied Linguistics and Language Research*, 6(2), 90–102.
6. Temur, B. & Ismailov, M. (2022). Challenges in Teaching Programming Languages in Uzbekistan. *Uzbekistan Journal of Education and Technology*, 5(1), 33–45.
7. Karimov, F. (2023). English as a Gateway to Global Digital Literacy: A Perspective from Central Asia. *TIIAME Research Digest*, 2(4), 58–67