IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

Java and the Development of Programming Skills: Assessing Students' Perceptions and Limitations

Dr. Jessica Rose E. Fernandez[1], Josephine E. Vidal[2], Harderly M. Pananggananan[3]
Faculty, College of Engineering and Information Technology, Surigao City, Philippines [1]
BSIT Students, College of Engineering and Information Technology, Surigao City, Philippines[2,3]

Abstract: *This study assessed the perceptions of Bachelor of Science in Information and Communications Technology (BSICT) students at Surigao del Norte State University (SNSU) regarding the role of basic Java programming in developing key programming skills. It focused on four core Java components—syntax, methods, classes, and file handling—and examined their relationship to the enhancement of students' communication, problem-solving, and critical thinking skills. Findings reveal that students perceive Java as a valuable tool for improving both technical proficiency and cognitive abilities. The study underscores the importance of foundational programming knowledge in shaping essential skills and recommends curriculum improvements to better align academic training with industry needs.*

Keywords: Academic training, Curriculum improvement, Foundational programming knowledge, Industry needs

I. INTRODUCTION

In recent years, the integration of Java programming into Bachelor of Science in Information and Communications Technology (BSICT) curricula received significant attention due to its perceived impact on students' essential programming skills. Recognizing Java's versatility and industry relevance, this study assessed BSICT students' perceptions of basic Java programming and its effectiveness in enhancing their programming skills, particularly communication, problem-solving, and critical thinking abilities.

Several studies explored the relationship between programming education and student skill development. One study [1] investigated the programming proficiency of BSIT students, identifying challenges with loop control structures and data type conversion. Another [2] analyzed student programming errors in Java courses, revealing difficulties with abstract concepts like multi-dimensional arrays, loops, and functions. These findings highlighted a need for instructional strategies emphasizing practical application and visualization. A further investigation [3] correlated motivation and self-efficacy with Java programming performance, showing that higher motivation predicted better academic outcomes. Additionally, a study [4] assessed the relationship between Java programming and graduate attributes in BSIT students, finding a positive correlation between Java programming and problem-solving and communication skills. Another study [25] emphasized Java's suitability as an introductory language due to its structured and object-oriented nature, which fosters logical thinking and problem-solving abilities. Collectively, these studies underscore the significant role of Java programming in developing essential programming competencies.

Despite previous insights, a gap remained in research specifically addressing how basic Java programming influenced the development of communication, problem-solving, and critical thinking skills among BSICT students. Previous research primarily focused on general programming anxiety, motivation, and performance, with limited attention given to the direct impact of Java programming education on these essential competencies. This study addressed this gap by investigating BSICT students' perceptions of basic Java programming and its role in enhancing their programming skills, emphasizing communication, problem-solving, and critical thinking. A quantitative, survey-based approach, utilizing a Likert scale (strongly agree, agree, disagree, strongly disagree), was employed to collect measurable data on students' perceptions and the impact of Java programming on their skills. The findings aimed to assist educators and

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

curriculum developers in designing targeted interventions and instructional strategies to foster these essential abilities among BSICT students.

This study assessed BSICT students' perceptions of basic Java programming in relation to their programming skills. It analyzed the relationship between Java programming components—syntax, methods, classes, and file handling—and the development of communication skills. Additionally, the study determined the impact of Java programming on students' problem-solving abilities and evaluated how learning Java programming influenced critical thinking skills among BSICT students. Finally, based on the findings, the study proposed recommendations for instructional strategies to enhance the effectiveness of Java programming education in developing essential programming competencies.

In Fig. presented a conceptual framework illustrating the relationship between Java programming features and the development of programming skills. The framework highlighted the core components of Java programming (syntax, methods, classes, and file handling) and their potential impact on three key programming skills: communication, problem-solving, and critical thinking. The title, "Java and the Development of Programming Skills: Assessing Students' Perceptions and Limitations," indicated that the research investigated students' views on how well these Java features contributed to skill development, acknowledging potential limitations or challenges. The visual separation of Java features and programming skills implied that the study analyzed the correlation or causal link between the two, potentially examining whether proficiency in specific Java features led to improvements in particular programming skills. The framework set the stage for a quantitative or qualitative study exploring the effectiveness of Java instruction in fostering these essential skills.

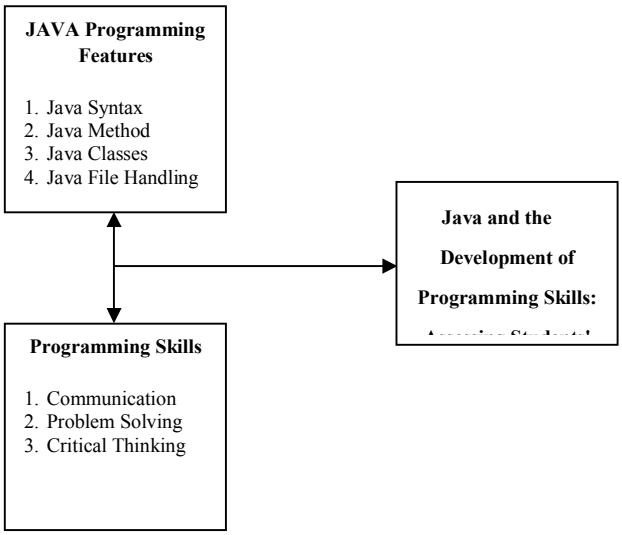

Fig. 1 A conceptual framework of the study

Java syntax formed the foundation of Java programming, defining the structure and rules for writing code. Several studies examined its impact on programming skills. One study [5] examined the difficulties students faced in understanding Java syntax, finding that its complexity often led to common errors, hindering learning progress. Another study [6] supported this, demonstrating that students who frequently practiced syntax exercises performed better in problem-solving tasks. A further study [7] found that interactive learning tools, such as code visualization and debugging environments, improved students' comprehension of Java syntax. Additionally, a study [9] investigated the effectiveness of online-based learning in Java programming, finding that students performed better with flexible learning environments. Another study [8] argued that traditional lecture-based instruction was less effective than hands-on programming exercises in improving syntax mastery. This emphasized the role of structured syntax practice in enhancing students' logical reasoning, which was explored by assessing their perceptions of the relationship between syntax mastery and problem-solving. This aligned with the study's objective of evaluating how different teaching methodologies influenced students' skill development. Understanding their preferences—whether for online-based

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

learning or hands-on programming—helped determine which approach should be more emphasized for effective Java programming instruction.

Understanding and implementing Java methods played a crucial role in structuring code efficiently, directly influencing problem-solving and critical thinking skills. One study [10] emphasized that challenges in comprehending programming concepts, including methods, hindered students' ability to solve problems effectively, highlighting the need for instructional approaches that enhanced conceptual understanding. Similarly, Turkmen and [11] found that students with strong programming foundations demonstrated better implementation of methods, leading to improved logical reasoning and analytical thinking. Another study [12] reinforced this perspective by reporting that cooperative learning strategies significantly enhanced students' grasp of programming constructs, resulting in greater proficiency in tackling complex coding tasks. Additionally, [13] demonstrated that precise feedback mechanisms in programming instruction led to a deeper understanding of methods, ultimately strengthening problem-solving and critical thinking skills. A further study [14] highlighted that mastering programming constructs allowed students to break down problems effectively and implement solutions efficiently. Collectively, these studies underscored the significance of Java methods in fostering essential cognitive skills, supporting the need for effective teaching strategies that enhanced students' programming capabilities.

A thorough comprehension of Java classes was fundamental to object-oriented programming, facilitating modularity and reusability, and thereby enhancing learners' problem-solving and critical thinking abilities. One study [15] observed that students who possessed a robust conceptual grasp of object-oriented programming principles, particularly classes, exhibited advanced analytical and problem-solving competencies. Similarly, [16] reported that the implementation of cooperative learning strategies contributed to an enhanced conceptualization of Java classes, leading to improvements in logical reasoning and structured coding practices. Another study [17] further substantiated this by demonstrating that the provision of precise feedback in programming instruction significantly improved students' cognitive assimilation of object-oriented concepts, thereby reinforcing their problem-solving and critical thinking capacities. Furthermore, [14] emphasized the importance of mastering Java classes, noting that such proficiency enabled students to structure their code systematically, promoting organized thinking and efficient implementation of solutions. Collectively, these studies highlighted the pivotal role of Java classes in fostering essential cognitive skills, underscoring the necessity of effective pedagogical strategies in programming education.

The integration of Java file handling into programming curricula was explored for its impact on enhancing students' communication, problem-solving, and critical thinking skills. One study [14] analyzed the concept of programming skills in developing problem-solving abilities and found that engaging in programming activities, such as file handling, enabled students to collaborate and devise solutions effectively. This study emphasized that mastering basic programming skills, including file operations, fostered higher-level reasoning essential for tackling complex problems. Another perspective was offered by [18], who discussed the role of programming challenges in enhancing logical reasoning. The study demonstrated that tasks involving file handling required analytical thinking, thereby improving problem-solving strategies and attention to detail. This aligned with the notion that practical programming exercises contributed to the development of critical thinking skills. Similarly, [19] emphasized the importance of problem-solving skills in Java programming. The article highlighted that practicing file handling operations not only sharpened analytical abilities but also enhanced critical thinking by requiring developers to evaluate different approaches to coding challenges. Furthermore, [20] provided a comprehensive guide on enhancing problem-solving skills in Java programming. The guide underscored that understanding file handling mechanisms was crucial for developing efficient solutions, as it involved breaking down complex problems and applying algorithmic strategies—key components of critical thinking. Additionally, [21] discussed the significance of file handling in Java, noting that it was integral for storing program output and performing various operations. The article suggested that mastering file handling was essential for effective communication between a program and its data sources, thereby enhancing overall programming proficiency.

While existing studies extensively examined Java programming concepts such as syntax, methods, classes, and file handling, a gap remained in understanding how BSICT students at SNSU perceived these topics in relation to their programming skill development. Most research focused on the general impact of Java programming constructs on

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

problem-solving and critical thinking; however, there was limited investigation into students' specific difficulties, learning preferences, and instructional needs within the SNSU academic context. Additionally, prior studies emphasized the effectiveness of cooperative learning, precision feedback, and other instructional strategies, but few explored how these methodologies aligned with the unique experiences of novice programmers in SNSU's learning environment. This study aimed to address these gaps by assessing students' perceptions and identifying key factors that influenced their proficiency in Java programming at SNSU.

II. RESEARCH METHOD

Research Design

This study employed a quantitative descriptive-correlational research design to assess the relationship between students' perceptions of basic Java programming concepts (such as Java Syntax, Java Methods, Java Classes, and Java File Handling) and their development of programming skills. This design was appropriate as it enabled the collection of measurable data from a specific group—second-year BSICT students—and allowed for an in-depth analysis of how students' perceptions related to their programming skill development, as supported by relevant references [22-24].

Participants and Sampling Method

The participants in this study were second-year Bachelor of Science in Information and Communications Technology (BSICT) students at SNSU for the academic year 2024-2025. In Fig. 2 showed total of 87 students were selected using purposive sampling, as they were currently engaged in learning Java programming as part of their curriculum. The inclusion criteria required participants to be officially enrolled second-year students who had completed or were currently enrolled in Java programming-related courses.

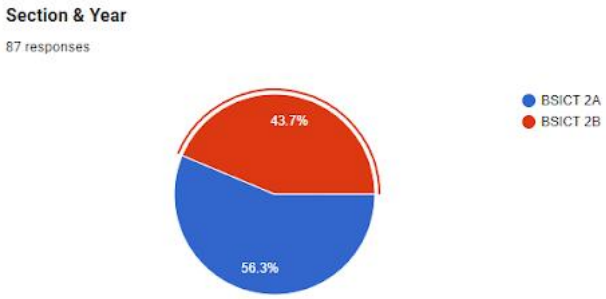

Fig. 2 Population of the Respondents

Data Collection Method

Data were collected using a structured, self-administered questionnaire designed specifically for this study. The questionnaire consisted of Likert-scale items grouped into four main sections: Java Syntax, Java Methods, Java Classes, and Java File Handling. Each section focused on assessing students' perceptions of how these Java programming concepts contribute to the development of essential programming skills—namely, communication, problem-solving, and critical thinking.

The survey instrument included sections assessing students' perceptions of four key Java programming concepts. The Java Syntax section explored students' perceptions of their ability to write clear and readable code, reflecting their communication skills within a programming context. The Java Methods section focused on modular coding practices and task decomposition, assessing their contribution to problem-solving skills. Questions in the Java Classes section evaluated how object-oriented programming concepts improved students' ability to organize code and think abstractly, thereby developing critical thinking. Finally, the Java File Handling section addressed students' ability to work with data storage, exception handling, and input/output operations, highlighting the application of critical thinking and problem-solving in more advanced programming scenarios.Each item utilized a 5-point Likert scale, ranging from

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

Strongly Disagree (1) to Strongly Agree (5), to measure the respondents' level of agreement with the statements. The questionnaire was administered either in printed form or electronically, based on the respondents' accessibility and convenience.

Data Analysis and Techniques

Collected data were encoded and analyzed using both Jamovi and Microsoft Excel. Descriptive statistics such as frequency, percentage, and mean were used to summarize the demographic data and survey responses collected via Google Forms. The analysis focused on evaluating BSICT students' perceptions of basic Java programming components—such as syntax, methods, classes, and file handling—and their relationship to the development of essential programming skills, including communication, problem-solving, and critical thinking.

III. RESULTS AND DISCUSSION

Table 1 presented the results of a survey assessing students' perceptions of their ability to utilize Java syntax effectively. The data showed a strong positive response across all items, with mean scores consistently above 3.78 (on a scale where a higher score indicated stronger agreement). The "Verbal Interpretation" column consistently indicated "Agree," further reinforcing the positive perception. This suggested that the students, on average, felt confident in their ability to use Java syntax for various tasks, including explaining their code to peers, documenting code structure clearly, providing clear explanations during debugging, structuring solutions logically, resolving logic issues through debugging, implementing problem-solving strategies using appropriate syntax, evaluating alternative syntax structures for efficiency, debugging systematically, and strategically modifying syntax to optimize performance.

TABLE 1: Results of a survey assessing students' perceptions

Statements	Mean	Verbal Interpretation
Whenever I write Java programs, I find myself...		
explaining my use of Java syntax to peers effectively	3.84	Agree
communicating Java code structure clearly in comments and documentation.	3.86	Agree
providing clear explanations when debugging syntax errors with peers.	3.78	Agree
using Java syntax to structure solutions logically.	3.82	Agree
debugging syntax errors to resolve logic issues.	3.82	Agree
using appropriate syntax to implement problem-solving strategies.	3.78	Agree
evaluating alternative syntax structures for efficiency.	3.84	Agree
debugging syntax errors systematically.	3.89	Agree
modifying syntax strategically to optimize performance.	3.86	Agree
MEAN	3.83	Agree

The high mean score of 3.83 for all items combined strongly indicated a positive overall perception of their Java syntax skills. This finding could have been interpreted as evidence of effective learning or a high level of inherent aptitude among the students surveyed. Further analysis might have explored potential correlations between these self-reported skills and actual programming performance on specific tasks.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

TABLE 2: Results on efficiency in terms of code completion

Statements	Mean	Verbal Interpretation
Whenever I use Java methods, I find myself…		
explaining the purpose of each method in my program clearly.	3.77	Agree
using meaningful method names that enhance code readability.	3.86	Agree
writing comments to describe the functionality of each method.	3.79	Agree
designing methods that efficiently solve specific programming tasks.	3.90	Agree
debugging method-related errors effectively.	3.75	Agree
structuring methods to reduce code redundancy.	3.85	Agree
evaluating different method structures to determine the best approach.	3.77	Agree
identifying logical errors in method implementation and debugging them.	3.79	Agree
analyzing the impact of method choices on program efficiency.	3.80	Agree
MEAN	3.81	AGREE

Table 2 presented survey results assessing students' self-reported abilities regarding Java methods. Similar to Table 1, the findings revealed a predominantly positive perception. The mean scores for all items consistently exceeded 3.75, and the "Verbal Interpretation" column uniformly showed "Agree," indicating a general agreement with the positive statements. This suggested that students felt confident in their abilities related to Java methods. Specifically, they reported feeling confident in explaining method purposes, using meaningful method names, writing descriptive comments, designing efficient methods, debugging method-related errors effectively, structuring methods to reduce redundancy, evaluating different method structures, identifying and debugging logical errors in method implementation, and analyzing the impact of method choices on program efficiency.

The overall mean of 3.81 further reinforced this positive perception of their Java method skills. As with Table 2, these self-reported skills could have been further validated by comparing them to actual performance metrics in coding assignments or assessments. The high scores suggested a strong understanding of method usage, potentially indicating effective instruction or a high level of student aptitude.

Table 3 displayed survey results focusing on students' self-assessed understanding and application of Java classes. The data revealed a highly positive perception, mirroring the trends observed in Tables 1 and 2. All mean scores were above 3.7, with most exceeding 3.8, and the "Verbal Interpretation" column consistently showed "Agree." This indicated a strong level of confidence among students regarding their skills in working with Java classes.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

TABLE 3: Results focusing on students' self-assessed understanding and application of Java classes

Statements	Mean	Verbal Interpretation
Whenever I create Java classes, I find myself...		
clearly discussing to my peers in naming and organizing class structures.	3.70	Agree
explaining the relationships between different classes in my program.	3.84	Agree
documenting classes to improve maintainability.	3.89	Agree
designing classes that logically represent real-world objects.	3.86	Agree
implementing encapsulation to protect class data.	3.79	Agree
using inheritance to extend functionality efficiently.	3.84	Agree
evaluating different class designs before implementation.	3.85	Agree
identifying and fixing logical flaws in class implementation.	3.83	Agree
debugging complex class relationships effectively.	3.83	Agree
MEAN	3.83	AGREE

Specifically, students reported feeling confident in their abilities to discuss class naming and organization, explain relationships between classes, document classes for maintainability, design classes that logically represented real-world objects, implement encapsulation, utilize inheritance efficiently, evaluate different class designs, identify and fix logical flaws in class implementation, and debug complex class relationships. The overall mean of 3.83 further emphasized this positive assessment. This high level of self-reported proficiency suggested effective teaching methods or a high level of student aptitude. As with previous tables, a comparison of these self-reported skills with actual performance on class-related programming tasks would have provided valuable additional insight.

Table 4 presented the results of a survey assessing students' self-reported skills in Java file handling. The data indicated a generally positive perception, although slightly less pronounced than the findings for Java syntax, methods, and classes (Tables 1-3). While the mean scores were still mostly above 3.7, one item ("writing robust code that prevents file corruption") showed a lower mean of 3.68. Nevertheless, the "Verbal Interpretation" column consistently indicated "Agree," suggesting overall agreement with the positive statements. Students reported confidence in writing comments to explain file operations, organizing file handling code for readability, discussing file handling concepts effectively, identifying and fixing file handling errors, handling exceptions properly, and evaluating different file-handling techniques.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

TABLE 4: Results of a survey assessing students' self-reported skills in Java file handling

Statements	Mean	Verbal Interpretation
Whenever I use Java file handling, I find myself...		
writing comments to explain file operations clearly.	3.79	Agree
organizing file handling code for better readability.	3.91	Agree
discussing file handling concepts effectively.	3.87	Agree
identifying and fixing errors related to file handling.	3.79	Agree
handling exceptions properly when dealing with files.	3.77	Agree
writing robust code that prevents file corruption.	3.68	Agree
evaluating different file-handling techniques before implementation.	3.80	Agree
identifying security risks related to file access.	3.84	Agree
debugging complex file-handling issues.	3.86	Agree
MEAN	3.81	AGREE

The slightly lower scores on writing robust code and identifying security risks (3.84) might have suggested areas where additional instruction or practice could have been beneficial. The overall mean of 3.81 suggested a reasonable level of proficiency in Java file handling, but the variation across items indicated potential areas for improvement in specific aspects of file handling skills. Further investigation could have explored the reasons behind the slightly lower scores on certain items and their correlation with actual performance on file handling tasks.

IV. CONCLUSION

This study examined the perceptions of 87 BSICT students at Surigao del Norte State University regarding basic Java programming and its connection to their programming skills. The demographic profile showed a slight majority of male respondents (54.0%), with most students enrolled in BSICT 2A (57.5%) and predominantly aged 19 to 20 years (66.7%).

The findings revealed that students generally agreed on the positive influence of Java syntax (Mean = 3.83), methods (Mean = 3.81), classes (Mean = 3.83), and file handling (Mean = 3.81) on their development of key programming skills—specifically communication, problem-solving, and critical thinking. These components were seen as helpful in enabling students to write code effectively, solve programming challenges, and articulate their logic clearly.

Overall, the results indicate that students have favorable perceptions of basic Java programming in relation to the development of their programming skills. This supports the Programming Skills Development Model, emphasizing that foundational Java concepts significantly contribute to students perceived growth in essential programming competencies. These insights may serve as a basis for enhancing programming instruction and curriculum design in information technology education.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

ACKNOWLEDGMENT

The researchers express their sincere gratitude to the administration of Surigao del Norte State University for their invaluable support in this study. We are deeply thankful for the provision of necessary data and for granting permission to conduct this research, enabling the team to successfully complete this project.

REFERENCES

[1]. S. D. Pacris Jr., Level of Proficiency of BSIT Students in Basic Programming. [Online]. Available: https://www.researchgate.net/publication/346554352_LEVEL_OF_PROFICIENCY_OF_BSIT_STUDENTS_IN_BASIC_PROGRAMMING

[2]. Y. Chen, L. Yao, J. Zhang, and Z. Wang, "A Study of Misconceptions and Missing Conceptions of Novice Java Programmers," 2022. [Online]. Available: https://www.academia.edu/75765064/A_Study_of_Misconceptions_and_Missing_Conceptions_of_Novice_Java_Programmers

[3]. B. E. Resuelo, M. M. Reynoso, and M. C. T. Demapula, "Motivation and Self-Efficacy in Java Programming among B.S. Information Technology and B.S. Information Systems Students," Journal of Namibian Studies, 2023. [Online]. Available: https://namibian-studies.com/index.php/JNS/article/view/3173

[4]. J. R. E. Fernandez, A. M. M. D. Dy, and M. Gales, "The Correlation between Java Programming and BSICT Graduate Attributes of Information Technology Students at Surigao del Norte State University," 2024. [Online]. Available: https://paper.researchbib.com/view/paper/401424

[5]. A Smith, "Predictors Of Java Programming Self-Efficacy Among Engineering Students InA Nigerian University," 2020. [Online]. Available: https://arxiv.org/abs/0909.0074

[6]. J. Johnson and C. Lee, "ClassCode: An Interactive Teaching and Learning Environment for Programming Education in Classrooms," 2021. [Online]. Available: https://arxiv.org/abs/2001.08194

[7]. M. Garcia, A. Lopez, and R. Torres, "Improving CS1 Programming Learning with Visual Execution Environments," Information, vol. 14, no. 10, p. 579, 2019. [Online]. Available: https://www.mdpi.com/2078-2489/14/10/579

[8]. Y. Chen and A. Patel, "ChatGPT, Can You Generate Solutions for My Coding Exercises? An Evaluation on Its Effectiveness in an Undergraduate Java Programming Course," 2022. [Online]. Available: https://arxiv.org/abs/2305.13680

[9]. N. Nordin, "The Effectiveness of Online-Based Learning in Java Programming Language: Student Perceptions and Performance," Journal of Technology and Operations Management, vol. 16, no. 1, pp. 25–37, 2021. [Online]. Available: https://www.researchgate.net/publication/354060561

[10]. A A. Lawan and A. Abuhassan, "What is Difficult in Learning Programming Language Based on Problem-Solving Skills," 2019. [Online]. Available: https://www.academia.edu/40958231

[11]. N. Turkmen and N. Caner-Yıldırım, "An Analysis of JAVA Programming Behaviors: Affect, Perceptions, and Syntax Errors among Low-Achieving, Average, and High-Achieving Novice Programmers," 2020. [Online]. Available: https://www.researchgate.net/publication/268207542

[12]. Garcia, "Cooperative Learning Strategies in Programming Education: Enhancing Comprehension of Programming Concepts," 2021. [Online]. Available: https://manuelgarcia.info/publication/cooperative-learning-computer-programming

[13]. G.-J. Hwang, Y.-H. Tung, and J.-W. Fang, "The Effect of Precision Feedback on Students' Understanding of Programming Constructs," Journal of Educational Computing Research, vol. 61, no. 2, pp. 388–412, 2023. [Online]. Available: https://journals.sagepub.com/doi/10.1177/07356331221087773

[14]. R. Harimurti, M. Purnama, and S. Widodo, "Analysis of Programming Skills Concept in Developing Problem Solving Skills," 2019. [Online]. Available: https://www.researchgate.net/publication/352380838

[15]. G. Türkmen and S. Caner, "The Investigation of Novice Programmers' Debugging Behaviors to Inform Intelligent E-Learning Environments: A Case Study," 2020. [Online]. Available: https://www.researchgate.net/publication/268207542

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

[16]. M. B. Garcia, "Cooperative Learning in Computer Programming: A Quasi-Experimental Evaluation of Jigsaw Teaching Strategy with Novice Programmers," Education and Information Technologies, vol. 26, pp. 4839–4856, 2021. [Online]. Available: https://manuelgarcia.info/publication/cooperative-learning-computer-programming

[17]. G.-J. Hwang, Y.-H. Tung, and J.-W. Fang, "Promoting Students' Programming Logic and Problem-Solving Awareness with Precision Feedback: A Two-Tier Test-Based Online Programming Training Approach," Journal of Educational Computing Research, vol. 61, no. 2, pp. 388–412, 2023. [Online]. Available: https://journals.sagepub.com/doi/10.1177/07356331221087773

[18]. MoldStud, "Developing Problem-Solving Skills Through Programming," 2021. [Online]. Available: https://moldstud.com/articles/p-developing-problem-solving-skills-through-programming

[19]. Geekster, "Problem Solving Skills in Java Programming," 2021. [Online]. Available: https://blog.geekster.in/problem-solving-skills-java-programming

[20]. Masai School, "Enhancing Problem-Solving Skills in Java Programming: A Comprehensive Guide," 2021. [Online]. Available: https://www.masaischool.com/blog/enhancing-problem-solving-skills-in-java-programming-a-comprehensive-guide

[21]. GeeksforGeeks, "File Handling in Java," 2025. [Online]. Available: https://www.geeksforgeeks.org/file-handling-in-java

[22]. Cabaron, I. M, "Relationship of Programming Skills and Perceived Value of Learning Programming among Information Technology Education Students in Davao Del Sur " 2022. Available: https://www.academia.edu/85947626/Relationship_of_Programming_Skills_and_Perceived_Value_of_Learning_Programming_among_Information_Technology_Education_Students_in_Davao_Del_Sur

[23]. Dela Rosa, A. P. M, "Effectiveness of an Online Course in Programming in a State University in the Philippines " 2022. Available:https://www.researchgate.net/publication/365821072_Effectiveness_of_an_Online_Course_in_Programming_in_a_State_University_in_the_Philippines

[24]. Grawemeyer, B., Halloran, J., England, M., & Croft, D, "Feedback and Engagement on an Introductory Programming Module," 2022. https://arxiv.org/abs/2201.0124

[25]. Malik, D. S, "Java Programming: From Problem Analysis to Program Design, "(8th ed.). 2020. https://www.cengage.com/c/java-programming-from-problem-analysis-to-program-design-8e-malik/9781337560092/