

Development of Web Application to Support Program Learning of Python and Ruby with Error Accumulation and Analysis Facility

Shunsuke Doi* Yoshiro Imai

Graduate School of Engineering, Kagawa University

2217-20 Hayashi-cho, Takamatsu, 761-0396, Japan

E-mail: s17g471@stu.kagawa-u.ac.jp*, imai@eng.kagawa-u.ac.jp

ABSTRACT

Beginners of programming sometimes faced to such situation that they themselves considered to be very poor programmer due to receipt of several types of errors from compilers, interpreters and language processors. But we have to learn and teach programming in some computer languages from lower education to higher one. It is necessary for us to utilize suitable tools for effective learning of current computer language programming. We have designed and implemented a Web application to support program learning of Python and Ruby. With this tool, it will be more fruitful for learners to recognize numbers of errors, error types, and history to change numbers and types of errors in order to gain individual understanding level of programming in Python and Ruby.

KEYWORDS

Programming education, Web service, Sever-client computing, Error recording and analyzing

1 INTRODUCTION

Programming has been one of the most important subjects in higher education, especially science and technologies. For a relatively long time, for example for ten years, there has been changing Paradigm in Programming Languages. So we have to catch up to these paradigm. Conventional programming education had been summarized as follows; (1) Algorithm oriented (2) Scientific calculation focused. Both of them had been considered to be very important for science and technologies. The usual aims of programming education had been normally and conventionally specified for teaching themes from solution of Quadratic equation to application of Monte Carlo simulation by means of Programming languages such as FORTRAN, PL/I, and/or assembly language.

The principle, "Practice makes perfect", has been considered to be very useful in programming education for many decade years. We had been training our programming skills based on the above principle so that we must learn during half year what our seniors had learnt in their half years. Tasks based on such an approach sometimes seemed to be very tough and time-consuming for not only beginners but also skilled engineers. It triggered to open a new era of object-oriented programming methodology. Why was object-oriented programming so attractive to teachers, engineers and researchers of that time? There were some reasons for code reusability of object-oriented programming and from antithesis to algorithm eccentrics. The former was on the basis of technical issues, while the latter was on the conceptual meaning. Recently, Python and Ruby became popular and seem to have been useful not only for engineering education but also for other learners who are out side of engineering domain[1]. We think it will be more necessary for learners to study programming style effectively and efficiently by means of useful tool to provide smart programming educational environment. So we have been developing educational tool realized by Web application for effective and efficient programming education using Python, Django[2].

This paper describes characteristics for three major functions of the tool in the next section. It also illustrates the detailed processing flows of the three functions in the third section. Its last section summaries our conclusion of this paper.

2 MAJOR FUNCTIONS OF TOOL

Our tool has configured as server client computing system and user interface is implemented on the browser in JavaScript. So our tool as Web application can run on the major browsers. This section describes the three major function of our tool to provide Programming learning environment of Python and Ruby through Web Application. The

three functions are as follows; program execution function, history-of-execution function, and error-information display function.

2.1 Program execution function

Any users of our tool can access to our prototype server through the major browsers even they have no Python nor Ruby environment in side of their own PCs. Basically, with tablets and smartphones which have compatible browsers with the major PC browsers, users can utilize our tool. The browser displays the user interface (UI) of our tool, Figure 1 shows one type of UI of our tool. Users select the programming language among Python and Ruby, the topmost of UI is the selector of programming language. They specify the filename to store their program in the writing box below the language selector of UI. They write or paste their program in Python or Ruby on the framework in the middle of UI And then remote server obtains users' data and program from UI on the browser, computes receiving program and then performs feedback of results into the corresponding UI after user clicks “forward” button in blue color located on the middle of UI. The result of program executed will have been displayed on the bottom framework of UI. Figure 1 shows Program Execution Result on UI of our tool.

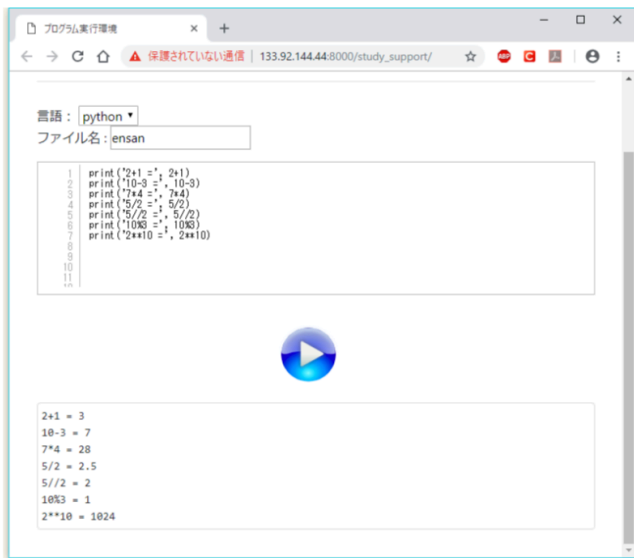


Figure 1. Program Execution Result.

2.2 History-of-execution function

Our tool can provide file management service and time-series ordered categorization one. And it can kindly alert which files had error-termination due to

実行日時	言語	ファイル名	エラー
9月 26, 2018, 6:06 p.m.	python	ensan.py	False
9月 25, 2018, 4:20 p.m.	python	list.py	False
9月 21, 2018, 2:38 p.m.	python	hello.py	False
9月 21, 2018, 2:38 p.m.	python	pi.py	False
9月 21, 2018, 2:37 p.m.	python	pi.py	False
9月 21, 2018, 2:36 p.m.	python	pi.py	False
9月 21, 2018, 2:35 p.m.	python	ensan.py	False
9月 21, 2018, 2:27 p.m.	python	local.py	True
9月 21, 2018, 2:27 p.m.	python	local.py	True
9月 21, 2018, 2:27 p.m.	python	local.py	False

Figure 2. History-of-execution Result.

the latest situation of program execution, which are shown in Figure 2. So users can easily remind their own programs written in the past and smoothly reutilize program selected for new application. Moreover, they can remember the other types of program, which could not execute correctly as they liked, by means of pink color alerting in Figure 2. In this case, they will be able to consider why their program cannot execute correctly with a philosophical eye and think about any type of errors in the relevant program calmly.

2.3 Error-information display function

Figure 3 shows Error-information Display Result. In order to think about type of error calmly (for ex-

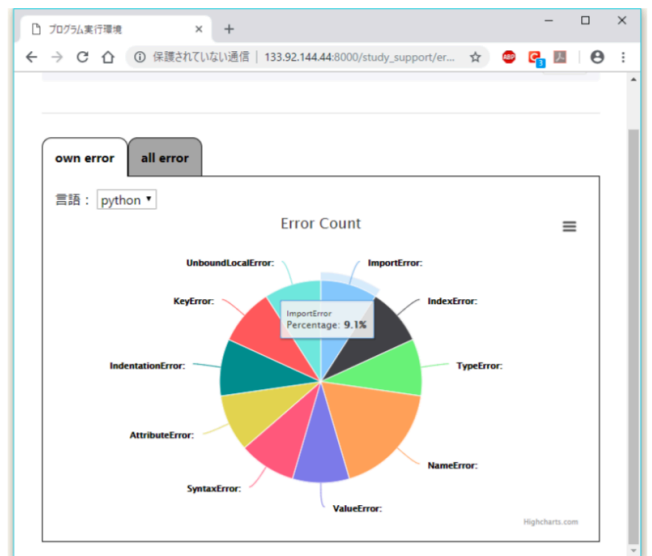


Figure 3. Error-information Display Result.

ample see [3]), users can utilize error-information display function of our tool. With error-information display function, each user can recognize self-tendency for him/her to write Python/Ruby program with any type of error included. In other words, our tool can illustrate that each user has some tendency of writing program which is subject to his/her intrinsic error. This is one of the important improvement of his/her programming skill in programming education through usage of our tool.

3 PROCESSING FLOWS OF TOOL

This section explains and illustrates the detailed processing flows of the three functions described in the above section. All of these flows are entirely implemented in the server side of our tool.

3.1 Program Execution Function

Figure 4 shows detailed processing flow for program execution. Data which are received in the client side have been transmitted from client into

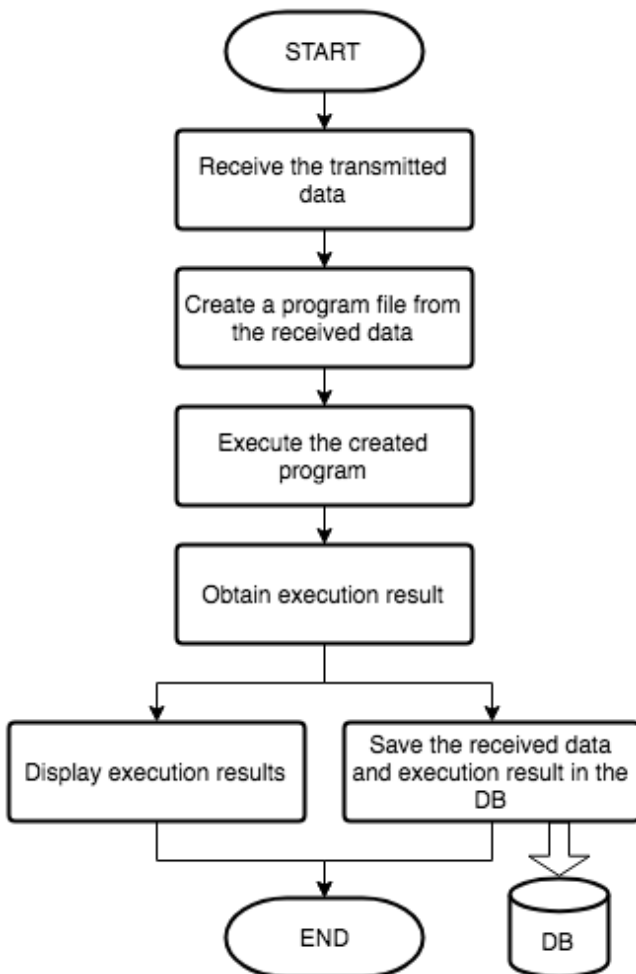


Figure 4. Processing Flow for Program Execution.

the server side. Server creates a program file whose name and other attributes are inherited from user input and stores the received text into it as program code. Server invokes the relevant language interpreter specified by user and then applies the previously stored program file into such a interpreter. Server waits to obtain eventual execution results from the interpreter after it executes. Such execution results are used for displaying on the client as well as saving into Database (DB) for future utilization. DB subsystem is loosely connected with server by means of SQL statement.

Our tool is designed and implemented as Web application of server-client model and has been precisely configured as three major parts, namely UI on the client side, server and DB subsystem behind server.

3.2 History-of-Execution Function

Figure 5 shows detailed processing flow for history-of-execution. History-of-Execution function is organized of two separated modules in Figure 5 and it is implemented in a series of following proce-

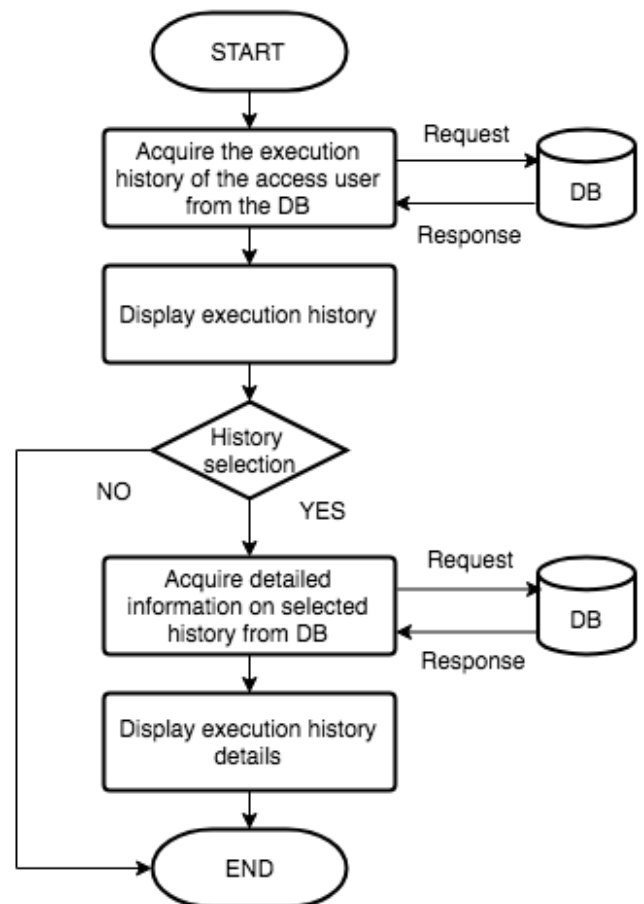


Figure 5. Processing Flow for History-of-Execution.

dures; Users select history-of-execution function, and then this request is transferred to the server. Server accesses DB subsystem and retrieves necessary information on such DB in order to acquire the focused execution history related with the relevant users through SQL statements. These information shows general all view of history-of-execution for the specific users who call this function and will be displayed on the UI of the client side.

If users want to know the detailed contents for the selected information, they shall advance to inform the UI to show more detail information. In Figure 5, 'YES' will be selected on the switch of "History selection". Server accesses again DB subsystem and also retrieves more suitable and more precise information on such DB in order to acquire the detailed contents such as program codes and execution results including possible error information if it has been existed. These information shows detail information of history-of-execution for the specific users who call this function and also will be displayed on the UI of the client side.

With utilization of DB subsystem, our tool can provide more powerful file management facility against remote users rather than normal PC itself where Python and/or Ruby programming is directly available.

3.3 Error-information Display function

It has been considered to be good idea that learners share information about their understanding levels, compare individual levels among them and utilize the relevant results from comparison into improvement of their understanding levels together with one another through some suitable ways.

At usage of traditional PC environment where only one user can write and execute his/her own program independently from others, it used to be very difficult for even friends learning the same subject to accomplish the above sharing information, mutual comparison and improvement together with mutual help. While our tool can provide not only individual learner-based error-information retrieval but also users-group-based error-information retrieval, aggregation and comparison. The former help each user to recognize self-tendency for him/her to write Python/Ruby program with some types of errors as previously described. The latter will support that group of users can share information about error types of their programming, compare individual information about error types and moreover under-

stand both of similarity and difference of their error patterns through aggregation of error information. Figure 6 shows detailed processing flow for error-information display. In order to realize such sharing, comparing and aggregating, error-information display function is implemented in a series of following procedures; Users select error-information display function, and then this request is transferred to the server. Server accesses DB subsystem and retrieves necessary error-information on such DB in order to acquire the focused execution history with error occurrence related with the relevant user and his/her group through SQL statements. These error-information of individual user as well as group of users have been aggregated in parallel in the form of graphical representation in order to realize easy understanding and comparing by user himself/herself as well as group of users. The graphical representation will be displayed on the UI of the client. One of the results has been already shown in Figure 3.

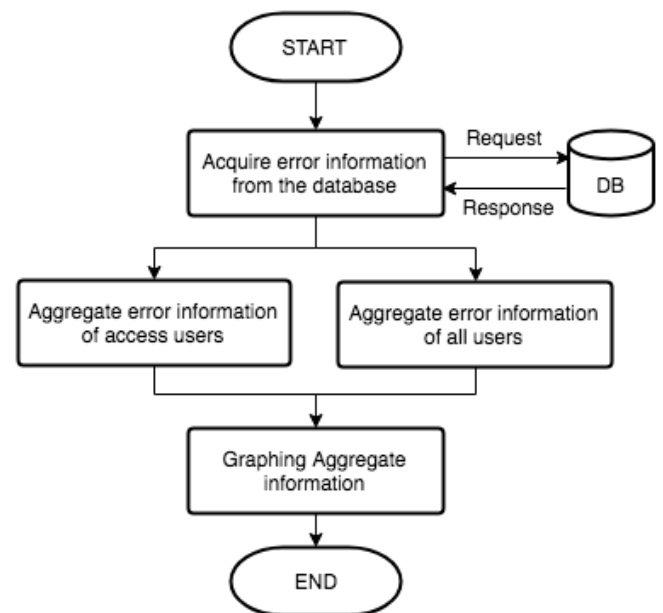


Figure 6. Processing Flow for Error-information Display.

4 DISCUSSION

Computing education, especially Programming education, is in enormous demand. More number of students will need programming in the future than today, they say. The book[4] written by Mark Guzdial of Georgia Institute of Technology presented the argument that student will not be all going to use programming in the same way and for the same purposes. He wrote "What do we

mean when we talk about teaching everyone to program? When we target a broad audience, should we have the same goals as computer science education for professional software developers? How do we design computing education that works for everyone?” And he had proposed use of a learner-centered design approach to create computing education for a broad audience. He reviewed the history of the idea that programming was not just for the professional software developer. And then he added research studies on teaching computing in liberal arts programs, to graphic designers, to high school teachers, and so on.

Philip J. Guo of San Diego said in his proceeding paper[5], “People from nearly every country are now learning computer programming, yet the majority of programming languages, libraries, documentation, and instructional materials are in English. What barriers do non-native English speakers face when learning from English-based resources? What desires do they have for improving instructional materials? We investigate these questions by deploying a survey to a programming education website and analyzing 840 responses spanning 86 countries and 74 native languages. We found that non-native English speakers faced barriers with reading instructional materials, technical communication, reading and writing code, and simultaneously learning English and programming.”

Therefore, we certainly believe that many numbers of students need programming and to learn programming for several purposes. And it is very important to employ a learner-centered design approach to create computing/programming education. At the same time, we must understand that non-native English speakers used to face many kinds of barriers when learning from English-based resources. It is a key idea for us to develop easy-to-use tool for programming education.

5 CONCLUSION

This paper introduces our tool of Web application based on server client computing which can provide useful programming education environment for learners of Python and Ruby programming. Our tool has been implemented with Django[2] and configured of the following three parts, namely UI on the client, server to handle the following functions, and Database subsystem. The three major functions which characterize our tool are as followed, (i) program execution function to run

your programs in Python and Ruby for remote users, (ii) history-of-execution function to provide more powerful file management facility against remote users rather than what normal PC possibly does for its user, and (iii) error-information display function to realize information-haring, error-type-comparing and group-error-aggregating.

With error-information display function, for example, users can understand self-tendencies for them to write Python/Ruby programs because such a function illustrates what kinds of error types they cause during their programming processes. With our tool, users will receive advanced information to recognize their intrinsic characteristics in programming and moreover to improve their programming styles more effectively and efficiently.

Acknowledgements

Authors are thankful to Dr. Koji Kagawa and Dr. Tomohiko Takagi of Kagawa University for their suggestive comments to our system development. They also appreciate daily support and discussion from the members of Imai’s Laboratory of Kagawa University.

REFERENCES

- [1] PYPL PopularitY of Programming Language index, <http://pypl.github.io/PYPL.html>
- [2] Django, a free and open-source Web framework, written in Python <https://www.djangoproject.com/>
- [3] Python ErrorCodes Summaries (in Japanese) <http://nonbiri-tereka.hatenablog.com/entry/2014/06/02/083206>
- [4] Mark Guzdial, “Learner-Centered Design of Computing Education: Research on Computing for Everyone,” Synthesis Lectures on Human-Centered Informatics, Vol.8, No.6, pp. 1-165, June, 2015.
- [5] Philip J. Guo, “Non-Native English Speakers Learning Computer Programming: Barriers, Desires, and Design Opportunities,” ACM Conference on Human Factors in Computing Systems (CHI), 14 pages, 2018.