International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 4
ISSN: 2088-5334

Quantitative Analysis of Online Courses for Learning the JavaScript Programming Language

Sergii Sharov [a,*], Supriya Chakraborty [b], Anatolii Zemlianskyi [a], Vira Kolmakova [c], Tetiana Prosiankina-Zharova [d]

[a] Department of Computer Sciences, Dmytro Motornyi Tavria State Agrotechnological University, Zaporizhzhia, Ukraine
[b] Department of Computer Science & Engineering, Brainware University Kolkata, Kolkata, India
[c] Department of Computer Science, Information and Communication Technologies, Pavlo Tychyna Uman State Pedagogical University, Uman, Ukraine
[d] Institute of Telecommunications and Global Information Space, National Academy of Sciences of Ukraine, Ukraine

*Corresponding author: *segsharov@gmail.com*

***Abstract*— Today, the JavaScript programming language ranks first both in the world well-known programming rankings and those in Ukraine in particular. The purpose of the research is quantitative analysis of online courses on learning JavaScript on well-known MOOC platforms. In order to achieve the purpose of our research, we used theoretical and quantitative (descriptive) methods. Also, we analyzed such well-known online platforms as Alison, Codecademy, Udemy, edX, Coursera as of December 2023. The number of online courses on each platform, their cost, an approximate user level, video length, etc. was used as criteria for the analysis. The analyzed MOOC was found to offer 1,795 online courses on learning the JavaScript programming language. Among them, 1,077 courses (60%) are paid, 201 courses (11%) can be taken for free, and 517 courses (29%) require optional payment. Most of the online JavaScript courses (788 courses, 44%) are developed for the Beginner level. Only 37 courses are available for experienced JavaScript programmers, which is 2% of the total number of online JavaScript courses. It was found that Codecademy and Udemy online platforms offer their users additional opportunities to learn JavaScript: programming projects (67 online courses); coding exercises (99 online courses); Practice test (55 online courses). The obtained results made it possible to formulate some proposals for choosing the optimal MOOC which helps to learn JavaScript, considering several selection criteria.**

***Keywords*— MOOC; online course; coding; programming language; JavaScript.**

Manuscript received 17 Mar. 2024; revised 10 May 2024; accepted 11 Jul. 2024. Date of publication 31 Aug. 2024.

I. Introduction

Today, possession of digital competencies is seen as a necessity [1], which plays an important role in employment [2], as well as in the increase of the employees' productivity in various industries [3], and in the economic efficiency of enterprises. At the same time, in the labor market, there is a high demand for IT specialists capable of developing software [4], in particular, websites and Internet services. As noted in the study by K. Mon et al. [5], the vast majority of dynamic websites are developed with the powerful and flexible JavaScript (JS) programming language. Its popularity among other programming languages is proved by high positions in international (for example, TIOBE, Stack Overflow) and Ukrainian (DOU.UA) ratings. As a result, a programmer who knows the JS language will always be able to get a job and a decent salary.

All the necessary programming skills can be developed at an educational institution, at advanced training courses, during self-education, etc. In the conditions of the information society and due to recent crises, academic activities have become more focused on distance or blended learning than classroom work. It has led to the emergence of new means of transferring social experience to students and the increase of their educational motivation. For example, Malik et al. [6], Alfarsi et al. [7], and Hiranishi [8] used a proprietary web application to teach programming. The analysis of possibilities provided by different environments for learning object-oriented programming is covered in the work by Batiha et al. [9]. Skalka et al. [4] built their own programming learning strategy based on using LMS Moodle. The importance of interactive online tools when learning programming is emphasized in the work by Hosseini et al. [10]. The capabilities of a mobile application for learning JS are explored in the work by Maskeliūnas et al. [11]. Without a doubt, the use of software applications and online environments for learning programming is impossible without the appropriate methodology, which is reflected in the works by [4], [6], [12], and other researchers.

One of the modern online educational resources that fully implement mass-character and openness is massive open online courses (MOOC). The advantages, conditions and potential risks of MOOC are highlighted in the works by Chansanam et al. [13]. Cilliers et al. [14] and Rõõm et al. [15] studied external and internal factors affecting the progress in taking online courses and their completion. Online MOOC platforms have proven to be useful for learning such programming languages as C [12], [16]; Python [17], [18]; Java [19], [20]. For this, standard MOOC capabilities (video content, testing), integrated software tools and learning strategies were used. Thus, an important factor in acquiring programming skills with the help of MOOC is the automatic evaluation of the program code fragments, which is reflected in [12]. The capabilities of integrated conversational agents when learning the Java language are reflected in the works by Aguirre et al. [19] and González-Castro et al. [21]. Babori [16] studied the potential of MOOC discussion forums to identify students' learning strategies in taking courses on algorithmizing and programming.

Recently, there has been a steady increase in online courses on various programming languages. It especially concerns those tools and environments that are most relevant among programmers and correspond to the modern trends in the IT sphere development. In view of this, the purpose of our research is quantitative analysis of online courses for learning the JavaScript programming language on well-known MOOC platforms. The novelty of this work lies in the coverage of current information on the quantitative indicators of online JavaScript courses that can be used to develop programming skills in a distance format. The obtained results will make it possible to formulate proposals for choosing the optimal online platform, which helps to learn JavaScript, taking into account several selection criteria.

II. Materials and Method

A. Characteristics and Popularity of JavaScript

The interpreted programming language JavaScript appeared in 1995 as a support tool for the Java language. It is currently one of the most popular programming languages for creating interactive web pages, surpassing Java in this regard. Typically, the dynamism and functionality of web pages is provided by the simultaneous use of Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript. HTML is used to form the content of a web page. CSS allows users to apply style to the content of the site, to customize the display of the text, images, location of site elements, etc. JS provides the interaction of the web page with the user, and it processes the data without reloading the page. Since JS is an essential tool in web development, programmers learn JavaScript after mastering HTML and CSS.

Since JS is a cross-platform language, dynamic web pages can run on different operating platforms. In addition to front-end development, JavaScript can be used to create browser extensions, applications for mobile operating systems (together with the React Native framework), server applications (Node.js is used), and desktop applications. JS is easy to use and productive; it quickly processes user requests without contacting the server. Having considered the analysis of the program code fragments of well-known programming languages, Abdulkareem and Abboud [22] found out that according to Halstead's metrics, the JS language occupied an intermediate position between the Python and Java languages.

The JavaScript language ranks first in the international and domestic rankings. According to the TIOBE index (https://www.tiobe.com/tiobe-index) as of August 2023, JS ranked 6th in the overall rating among 278 programming languages and environments. This index indicates a fairly high level of JS popularity among Internet users. The rating is scored each month after the analysis of requests in 25 search engines with the highest rating of Similarweb. The criteria of the analysis include the name of the programming language with possible false positives, the confidence coefficient and other conditions for data filtering. According to a survey conducted by the public platform Stack Overflow (https://survey.stackoverflow.co/2023/#technology) as of May 2023, the JavaScript language ranked first in the category "Technology" among 50 programming languages. By the way, according to this rating, HTML/CSS and Python took 2nd and 3rd places, respectively. More than 87,000 selected software developers from 185 countries participated in the survey, which indicates a significant sample of data for analysis and the validity of the results obtained.

If we consider the popularity of programming languages among Ukrainian programmers, the research done by the developers of the DOU.UA online platform will be quite interesting. They conducted a survey among 9,060 Ukrainian IT specialists and found out that in 2023 JavaScript took the first place in popularity (19.1% of the total number of the surveyed respondents). As for Java (14%) and Python (13.4%), they took the second and third places. The rating included 20 programming languages which are the most popular in Ukraine (https://dou.ua/lenta/articles/language-rating-2023/).

Proskura et al. [23] surveyed 105 students who were getting their education at one of the largest well-known national universities of Ukraine. Most of the surveyed students studied in senior courses, so they were aware of the ways of their professional growth. The results of the survey showed that JavaScript occupies the first place (51.5% of the total number of the surveyed respondents) among the WEB programming languages which students independently choose to implement projects in IT companies; those students want to improve or are already improving their coding skills in this programming language.

B. Methodological Approaches and Tools for Learning Programming

Training of future IT specialists, in particular programmers, can be implemented through formal and informal education. Learning effectiveness is influenced by such external factors, as meeting the requirements of a potential/actual employer, the need to be assessed in the discipline, encouragement and assistance of fellow programmers. Internal factors include persistent motivation to learn a specific programming language, the desire to get a job in a specific specialization [24], or to try one's hand at a new field of knowledge. The work by Çetinkaya and Baykan [25] insists on identifying students who can acquire programming skills quickly. The results of the students' survey can be used by teachers and parents to guide the students in the development of their future professional career related to programming.

Learning programming is a complex process that involves understanding the paradigm and fundamental concepts of programming, the use of software structures, the logic of program operation [9], [26]. This is especially true for understanding and improving the software with an open-source code, which is quite often written by different programmers [27]. Therefore, it is recommended to use various approaches and techniques taking into account the specifics of the programming language, individual characteristics of students and other factors. For example, to understand a given software code, it is advised to use comments that will allow recalling the features of the software structure and its purpose. Mastery of programming skills depends on a sufficient number of learning tasks that have practical benefits for the future profession [12]. In case the tasks provide students with practical problems from real life, it contributes to the rise of their cognitive interest and satisfaction with learning. The work by Hosseini et al. [10] indicates the advantage of working examples at the early stages of the programming skills formation. With the help of the examples, future programmers can learn the basic patterns of the program code, basic program structures which depend on the condition of the task, and they can understand the semantics of a specific programming language in more detail. For better understanding of software constructions, one can use task templates with program code, on the basis of which students can create complete programs. At the same time, the work by [28] emphasizes the quality of the programming tasks templates, since incorrectly compiled templates will not allow students to complete the tasks and compile a workable program.

Our own experience confirms the opinion about the existence of several algorithms and different ways of solving the same problem. Creativity in programming depends on the programmer's experience, the tools chosen to solve the task, and the time spent. If the educational task has only one solution to the problem, which is determined in advance by the course developers, it reduces the opportunities for the development of creative thinking. The negative impact of detailed instructions for a practical task is indicated in [17]. Therefore, acquisition of programming skills should not be limited to practicing only one method or one software construction. Litherland and Kluge [29] suggest that programming skills should be formed through several stages of interaction with the program, starting from understanding the purpose of creating the program and finishing with error handling and obtaining a finished software product.

Acquisition of programming skills can be done with the help of appropriate software or online educational environments. At this, distance education is provided in full [30], and the educational content becomes more visual and interactive. For example, to test theoretical knowledge on various topics in a programming course, one can use the web application described in [6]. The specific feature of this application is the implementation of an algorithmic model of the analysis of the given problem, which allows students to consistently move from the formulation of the problem to the formation of the software code. In order to visually start the work of the programming language compiler, you can use the web tool described in the research by [8]. The author's software development, described in the work by [31], implements a hybrid method of teaching programming through the use of a visual designer and a code editor. The works by [11], [27] emphasize the development of applications that, due to the use of multimedia, allow visualizing the operation of software structures and various types of algorithms. The simultaneous use of program block designers and multimedia files allows students to better understand the structure of the program and the logic of its operation. The importance of interactive learning when learning programming is indicated in [10].

Providing feedback between the teacher and student is essential to understanding the basics of programming, including JavaScript. In this case, the teacher's professionalism and developed social competence [32] become important, especially when communication takes place in the conditions of distance learning. In case the teacher is not available, the role of an assistant can be assigned to the corresponding software tool, which is described in the work by [5]. An effective approach to learning programming is the implementation of automatic validation of the user-entered programming code [26]. At the same time, if the user wants to independently check the functionality of the JS code on a working web page, it can be done without any additional software. It is enough to activate the developer mode in the Internet browser (hot key F12).

C. MOOC Capabilities for Learning Programming

Today, due to their inherent characteristics, MOOC are widely used in the educational process of higher education institutions and during lifelong learning. Their use enables higher education institutions to provide more flexible learning opportunities [33], to strengthen their own university brands through the creation of online courses for everyone [34]. Online courses on MOOC platforms are useful for working people who want to improve their skills in their spare time [35]. MOOC allows taking advantage of the best educational practices from leading institutions of higher education, well-known international organizations and other institutions.

According to J. Skalka et al. [4], MOOC capabilities are best suited for teaching large numbers of students with different levels of training. The work by [36] draw attention to the fact that MOOC ensure the principles of massive character and openness. At the same time, MOOC allows implementing the principle of an individual approach, since each registered user can build his/her own educational trajectory and take courses at the time convenient for him/her. Learning with the help of MOOC allows users not only to get access to a variety of educational content, but also to form digital competencies. While taking online courses, users can develop self-regulation skills for their own educational activities, as well as the ability for independent learning and self-control [18], which are the basis of lifelong learning. The work by [14] draws attention to the possibilities of MOOC for expanding knowledge that can be used in further employment.

Traditionally, an online MOOC consists of video lectures, tests for each educational topic, and a final test. Depending on the specifics of the online course, it can have different means of providing feedback and evaluation, including discussion forums, quizzes, and practical tasks. Having such features as multimedia and interactive online tools, MOOC are significantly different from a traditional lecture [7], where the teacher offers theoretical material, sometimes supporting it with visual aids. As a result, MOOC are often seen as a useful addition to the learning environment [20].

Learning with the help of MOOC involves a high degree of student motivation, which affects self-monitoring of one's own achievements and further distance learning. There are some factors which can lead to a decrease in motivation to complete an online course. They are limited feedback [18], [13], low quality of educational content [14], and previous programming experience [15]. In case the online course is used as an additional resource for learning programming within the educational institution, the teacher will be able to help and explain the issues which are unclear. Also, the teacher can analyze user actions with an external online monitoring platform, as described in the work by [36]. The received information will allow users to pay attention to problematic moments when taking the programming course and make appropriate adjustments.

In the case of self-study of the programming course, the user will have many more difficulties. As a result of the inability to independently master an online programming course due to insufficient understanding of the educational material, students do not complete it to the end. A promising solution to this problem is the use of conversational agents/assistants built into MOOC, which will offer users help in taking the online course [19]. Their advantage means using different ways of user communication (through text interface or voice messages) with the help of natural language [21].

Acquisition of programming skills involves writing a program code and its evaluation. The user gains practical experience with each developed program in a specific programming language. Therefore, such online courses should contain fragments of the program code with a detailed explanation, as well as the tasks for developing program structures within a separate topic. An effective addition to learning programming is the use of external MOOC subsystems for automatic verification of the program code fragments, which is described in the work by [12]. Their use provides instant feedback between the online course and the student's practical work.

D. Data Collection and Preprocessing

With the help of analysis and synthesis of scientific literature, we studied the issues related to the specific features of programming skills formation, the use of electronic resources and software tools for learning programming. A quantitative (descriptive) research method was used to determine the number of online JavaScript courses. The selection of online courses was made with a rubricator or an automatic search by the keyword "JavaScript". It was assumed that the word "JavaScript" could be included in the title of the online course. There are also courses that include one or more topics on learning the JavaScript language. During the analysis, the following characteristics were taken into account for each MOOC online platform: the number of online JavaScript courses, the cost, an approximate level of the user, the amount of video material within a separate course, the estimated time for the course completion. Such well-known online platforms as FutureLearn, Alison, Codecademy, Udemy, edX, Coursera were analyzed as of December 2023. The choice of online platforms depended on the previous research [24], [37] and the authors' practical experience of taking online courses.

On FutureLearn platform, the sample for the analysis was selected by the search for courses in the thematic catalogue "Subjects" – "IT & Computer Science" – "JavaScript". There are 3 online courses on learning the JavaScript programming language in the catalogue. This platform was not used for further quantitative analysis since the number of online courses is low there.

The search for online courses on Alison platform was carried out with an automatic search. Thus, it was possible to find 46 online courses for learning the JavaScript language. In case of using the rubricator, users are offered the thematic block "Explore Courses" – "Free Online IT and Computing Courses", which contains 866 online courses, some of which concern learning JavaScript. The authors consider this method of searching for online JS courses inexpedient, as it takes a lot of time.

On Codecademy platform, the search for the necessary online courses was carried out with the help of the rubric "Catalog" – "JavaScript". Through this search method, the user can find 80 online courses. The automatic search by the keyword "JavaScript" makes it possible to find 84 courses. The possibility to filter the found online courses is available only in the case of using the rubricator. Therefore, this method of the sample formation was chosen as the main one.

The sample of online courses for learning the JavaScript language on Udemy platform amounted to 1,198 courses. This number can be obtained if the user goes step-by step to the rubrics "Categories" – "Development" – "Programming languages" – "JavaScript" or to such rubrics as "Categories" – "Development" – "Web Development" – "JavaScript".

The search for online courses on learning the JavaScript language on edX platform was carried out in such thematic catalogues as "Learn" – "View all topics" – "Computer Programming" – "JavaScript". In this way, we found 14 online courses that contained the word JavaScript in the title or implied learning the JavaScript language in the course content. The automatic search for online courses by the keyword helped us to find 67 courses, most of which were not related to the JavaScript language. Therefore, the authors consider this method of the sample formation inexpedient for achieving the purpose of the research.

The sample of online courses on learning the JavaScript language on Coursera platform was formed through the search for online courses by the keyword "JavaScript", which amounted to 457 courses. On Coursera online platform, there is no separate section devoted to the JavaScript language, so this method was not used to form a targeted sample.

III. Results and Discussion

On Alison online platform, the user can find the desired course on learning the JavaScript language with an automatic search by the keyword “JavaScript”. In this way, the user is offered to take 46 online courses. All courses are different in their content, deadline, and viewing video materials. Additional information is provided for each online course: the number of students on the course, the estimated number of hours to take the course, a brief description of the modules included in the course, the course developer, the skills that will be formed while taking the online course. Additional filtering options of the automatic search help you to quickly find the desired course and start studying. Quantitative indicators are presented in Table 1.

TABLE I
THE NUMBER OF ONLINE COURSES ON LEARNING JAVASCRIPT ON ALISON PLATFORM

Course Level	Count	Course Type		Course Duration (hours)				Price
		Certificate	Diploma	2-3	4-6	7-10	11-15	
Beginner	30	28	2	17	9	3	1	+-
Intermediate	7	7	0	6	0	0	1	+-
Advanced	9	6	3	2	4	0	3	+-
Summary	46	41	5	25	13	3	5	

As we can see from Table 1, the user is offered 41 online courses (89% of the total number of courses on learning the JavaScript language), upon completion of which a certificate can be obtained. Alison online platform also offers 5 courses (11% of the total number of courses) that provide a diploma. As a rule, such courses are long-term and contain a lot of information. All courses on the platform are free and do not require tuition fees. It should be noted that the certificate or diploma certifying the completion of the online course is paid.

All the online courses are differentiated by level to consider the users’ previous training. On the Alison platform, most of the online courses are aimed at the Beginner level (30 courses, 65% of the total number of JavaScript courses). More experienced programmers can take advantage of the online courses aimed at the Intermediate (7 courses, 15%) and Advanced (9 courses, 20%) levels. Also, the online courses are differentiated according to the different amount of video material that should be viewed during training. The distribution of online courses by the length of video material is presented in Figure 1.

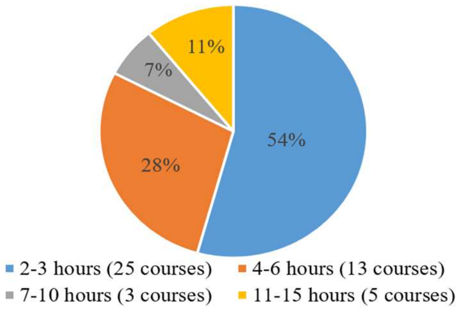

Fig. 1 The number of online courses on Alison platform according to the length of video material

Most of the online courses on Alison platform are of short duration, taking up to 2-3 hours (25 courses, 54% of the total number of JS courses) and 4-6 hours (13 courses, 28%). In our opinion, this online resource is primarily designed for beginners in programming. Therefore, it offers more online courses that involve watching a small number of videos to familiarize oneself with the JavaScript programming language.

One can learn the JavaScript programming language on Codecademy online resource. To do this, go to the rubric “Catalog” – “JavaScript”, which contains 80 online courses. Additional information is offered for each course: a brief description of the course, the number of students, an approximate level of the user training, the estimated number of hours spent on taking the course, a brief description of the modules included in the course, the skills that will be developed while taking the online course, the number of quizzes, and availability of the certificate. The specific feature of these online courses is programming projects that students will develop during the course. The additional characteristics of the online course will allow the user to choose the most optimal course for himself/herself and only then enroll in it. Quantitative indicators of the online courses on Codecademy online platform are presented in Table 2.

TABLE II
THE NUMBER OF ONLINE JAVASCRIPT COURSES ON CODECADEMY PLATFORM

Course Level	Count	Price		Course Duration (hours)		
		Paid	Free	<5	5-10	10-20
Beginner	38	3	35	31	5	2
Intermediate	41	14	27	28	10	3
Advanced	1	1	0	0	1	0
Summary	80	18	62	59	16	5

As we can see from Table 2, most of the online courses (62 courses, 78% of the total number of the courses on learning JavaScript) are free to take. At the same time, a certificate of the successful completion of the online course is included in paid plans. All of the online courses on learning JavaScript on Codecademy platform are differentiated according to the user level. There is nearly equal number of online courses for the Beginner level (38 courses, 48% of the total number of courses on learning JS) and Intermediate level (41 courses, 51% of the total number of courses). There is only one online course that is available for the Advanced level programmers. It is called “Learn Advanced React” and it is just 1% of the total number of courses on learning JavaScript. Most of the online courses contain tasks for practical application of the acquired competencies. For instance, 32 online courses (84%), which are created for the Beginner level, include practice projects. Among 41 online courses, which are meant for a more advanced level, 35 courses (85%) also mean realization of practice projects.

The same way as on Alison online platform, differentiation of online courses can be made through the different amount of video material that must be viewed to complete the course. The distribution of online courses by the length of video material is presented in Figure 2.

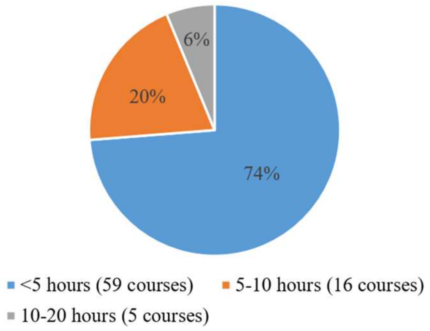

Fig. 2 The number of online courses on Codecademy platform according to the length of video material

Most of the online courses (59 courses, 74% of the total number of JS courses) are short-term, taking up to 5 hours to study. There are much fewer courses that take up to 5-10 hours. There are 16 online courses of this kind, which is 20% of the total number of online courses for learning the JavaScript language. There are almost no long-term courses that involve studying the material for 10-20 hours on Codecademy platform (5 courses, 6%).

With the help of Udemy online platform, users can learn the JavaScript programming language through 1,198 online courses. Going to the page of a specific course the user can get some additional information, namely a brief description of the course, the names of the topics covered within the course, the titles of video lectures within a separate topic, the number of coding exercises and articles, cost, etc. In addition, Udemy online platform has a powerful filtering service that allows users to choose the optimal course considering various parameters of the sample: rating, duration of the video content, topic, subcategory, training level, language, cost, functionality, subtitles. Quantitative indicators of online courses on Udemy online platform are presented in Table 3.

TABLE III
THE NUMBER OF ONLINE JAVASCRIPT COURSES ON UDEMY PLATFORM

Course Level	Count	Price		Features			
		Paid	Free	Subtitles	Quizzes	Coding Exercises	Practice test
Beginner	517	436	81	371	101	53	10
Intermediate	159	141	18	115	28	9	4
Expert	10	8	2	6	1	0	1
All levels	512	474	38	380	94	37	40
Summary	1,198	1,059	139	872	224	99	55

As we can see from Table 3, most of the online courses (1,059 courses, 88% of the total number of JS courses) are paid. At the same time, to provide an opportunity to learn for everyone, regardless of financial situation, the developers of Udemy online platform offer free courses to users. In total, on the resource we have 139 free online courses on learning the JavaScript programming language, which is 12% of the total number of courses. We consider the multilingual interface of the online platform, which is available in 30 languages, to be a positive practice. In addition, to increase the target audience, 872 online courses (72% of the total number of JS courses) contain subtitles in different languages. The same way as on Codecademy platform, online courses on Udemy platform provide a possibility to test the acquisition of knowledge in practice. For this, users are offered coding exercises (99 courses, 8.2% of the total number of JavaScript courses), Practice test (55 courses, 4.5% of the total number of courses) and Quizzes (224 courses, 18.7% of the total number of courses).

On Udemy platform, online courses are differentiated by the length of video material, which allows users to determine whether they have some free time to complete the course. The distribution of online courses according to the number of hours spent on watching videos is shown in Figure 3. As we can see in Figure 3, the number of courses with a minimum amount of video material is the least. There are only 115 courses of this kind, which is 10% of the total number of online courses for learning the JavaScript language. There is also a small number of long-term online courses that involve viewing video material for more than 17 hours. They make up 15% (184 courses) of the total number of online JS courses. The number of online courses with a total length of video material in the range of 1-3 hours (315 courses, 26%) and 6-17 hours (324 courses, 27%) is the largest. It can be concluded that users are offered a fairly wide selection of online courses on learning the JavaScript language, taking into account their business.

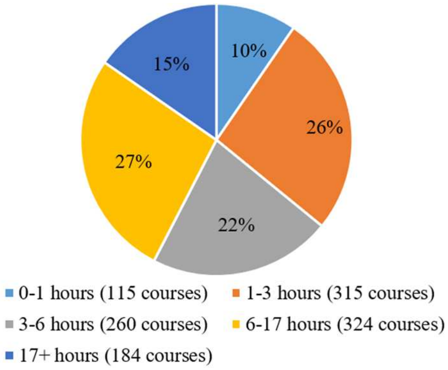

Fig. 3 The number of online courses on Udemy platform according to the length of video material

One can acquire JavaScript programming skills on edX online platform. The user will be offered 14 online courses in the "JavaScript" thematic directory. For more detailed information about the course, the user should go to its page. It contains information about the number of weeks, the cost, the names of the topics covered in the course, the course developer, the knowledge and skills that will be formed after the course's completion. In this search mode, there is no filtering service, so the user will have to manually view each online course separately. Quantitative indicators of online courses on edX online platform are presented in Table 4. As we can see from Table 4, most of the online courses (13 online courses, 93% of the total number of JavaScript language courses) are developed for the Beginner level. All online courses are presented in English. All courses are partially free.

TABLE IV
THE NUMBER OF ONLINE JAVASCRIPT COURSES ON EDX PLATFORM

Course Level	**Count**	**Price**	**Course Duration (hours)**		
			<10	**10-20**	**>20**
Introductory	13	+-	1	7	5
Intermediate	1	+-	0	0	1
Summary	14		1	7	6

In the free mode, the user gets access to the course materials until a certain date without the possibility to get a certificate after the course completion. If the course has been paid for, the restrictions are removed. A special feature of the platform is clearly established terms for taking the online course. If the term has expired, the online course is transferred to the "Archived" status. Training takes place for a few weeks, several hours a week. Most of the online JavaScript courses are medium-term, taking up to 10-20 hours (7 courses, 50%), and long-term, which involve learning for more than 20 hours (6 courses, 43%).

On Coursera online platform, users can learn the JavaScript programming language with the help of 457 online courses. They can get access to them through an automatic search by the keyword "JavaScript". Users can get more detailed information about the course by going to its page. There, they can find information about an approximate level of training, the skills that the user will receive after the course completion, the names of the topics covered in the course, the names of the videos that must be viewed during the course and their length. There is some useful information such as the number of videos, tests and educational materials offered for self-study. Such detailing will allow the student to choose the most optimal course that meets his/her needs. In addition, Coursera online platform has a filtering service that allows users to narrow down the selection of JS courses with the following parameters: Subject, Level, Duration, Educator, Language, Subtitles, etc. Quantitative indicators of online courses on Coursera online platform are presented in Table 5.

TABLE V
QUANTITATIVE INDICATORS OF ONLINE COURSES ON COURSERA ONLINE PLATFORM

Course Level	**Count**	**Price**	**Duration (Months)**		
			>1	**1-3**	**4-6**
Beginner	190	+-	97	92	1
Intermediate	191	+-	106	85	0
Advanced	17	+-	11	6	0
Mixed	59	+-	24	35	0
Summary	457		238	218	1

As we can see from Table 5, online courses for learning the JavaScript language on Coursera platform are mostly focused on the Beginner (190 courses) and Intermediate (191 courses) levels. There is almost the same number of such courses, which in total makes up 83% of the total number of online courses for learning the JavaScript language. There are also 13% (59 courses) of online courses aimed at different levels of user training. There are fewer courses for advanced users, the part of which is approximately 4% (17 courses).

While analyzing information about the selected online courses, we noticed that all of them are partially free. Coursera online platform provides a way of commercialization which means that the user is provided with a free trial version of the online course with full access to the educational materials for 7 days. At the same time, the course developers note that they can provide financial aid to pay for the course. As for the time spent on the online course, the developers measure it in weeks and in hours per week. If we consider this indicator, the full completion of the online course on Coursera platform takes about 1-4 weeks (238 courses, 52%) and 1-3 months (218 courses, 47%), which is 99% of the total number of online courses on learning the JavaScript language. On the Coursera platform, there are only 2 large online courses (1% of the total number of courses). Generalized data on quantitative indicators of the online courses which are located on the analyzed online platforms are presented in Table 6.

TABLE VI
QUANTITATIVE INDICATORS OF THE ONLINE COURSES LOCATED ON THE ANALYZED ONLINE PLATFORMS

Indicators	**MOOC**					**Count**
	Alison	**Codecademy**	**Udemy**	**edX**	**Coursera**	
Total	46	80	1,198	14	457	1,795
Course Level						
Beginner	30	38	517	13	190	788
Intermediate	7	41	159	1	191	399
Advanced	9	1	10		17	37
Mixed	-	-	512	-	59	571
Price						
Free	-	62	139	-	-	201
Paid	-	18	1,059	-	-	1,077
Mixed	46	-	-	14	457	517

The analysis of Table 6 allows us to conclude that the majority of courses (1,198 courses) on learning the JavaScript language are located on Udemy online platform. This is 67% of the total number of online courses found on all the online platforms. Quite a lot of courses for learning the JavaScript language are located on Coursera online platform (457 courses, 25%). The least of all courses for programmers who use the JavaScript language are located on edX online platform (14 courses, 0.8%). Most of the online courses are aimed at the Beginner level, with 788 online courses for learning the JavaScript language. This is 44% of the total number of online courses found on all the online platforms.

571 (32%) online courses are offered for study without considering the level of students' training. Unfortunately, the majority of online courses (1,077) are paid, the part of which is 60%. At the same time, 29% of the online courses are available for learning in a blended mode, which allows users to familiarize themselves with the educational content of the course and decide whether to pay for the educational content or the certificate after its completion.

The high demand for programmers in the labor market has led to the increase in the number of users who want to learn programming. Since the IT industry is constantly developing, we advise everyone interested in learning the most promising and common programming languages. In this aspect, the JavaScript language is quite an optimal option for learning. It is used as a basis for creating dynamic web pages; it does not require any additional programming environments, except for the browser; it is easy to learn, and you will not be distracted by complex concepts in programming; there are plenty of ready-made software structures and modules that can be used for training and further work [5].

We believe that learning JavaScript through MOOC is a good alternative to face-to-face learning. For this, users are offered a variety of online courses, the number of which increases every year. So, over the past two and a half years, the number of online JS courses on Alison online platform has increased by 33 (there were 13 courses); on Udemy online platform it has increased by 73 (there were 1,125 courses); on Coursera online platform it has increased by 186 (there were 271 courses); on Codecademy online platform it has increased by 46 (there were 34 courses) [37].

At the same time, the features of MOOC require the use of additional approaches to the educational process when learning programming. As it is stated in the work by [24], the quality of learning programming languages with the help of online courses depends on the quality of educational content, availability of practically-oriented tasks, tools for increasing motivation and providing feedback, and a possibility of automated verification of the program code. Developers of online courses have a great responsibility to provide users with high-quality educational content, which means a combination of text, visual and audio information. In addition, scientists emphasize the compliance of the educational content with the copyright standards, that is, the educational material must be unique [34]. In this context, E. Kaila and K. Lemström [20] point out that it is appropriate to use scheduled online courses while studying the discipline along with mastering unscheduled online courses.

The ease of using MOOC [13] influences users' willingness to take online courses so that to acquire programming skills. If the online platform is inconvenient or too difficult, the user can freely switch to another MOOC platform. In this context, a multilingual interface is important, as well as the availability of a service for filtering courses based on various parameters. Udemy online platform implements these criteria most fully. The online course should take into account the busyness of users and their desire to learn programming in their free time [33]. This requirement is solved by all the analyzed online platforms, which differentiate courses by the length of video material or the number of weeks necessary to complete the course.

Acquisition of programming skills involves the availability of practical activities during the compilation of algorithms and writing the program code in a specific programming language. At the same time, familiarization with examples in textbooks and during lectures has a much smaller effect. As a result, it is possible to encourage users to take a specific online course with the help of attractive tasks that will be of practical importance [10]. During quantitative analysis of online courses on learning the JavaScript language, it was found that such platforms as Codecademy online platform, which has programming projects (67 online courses), and Udemy, which offers coding exercises (99 online courses) and Practice test (55 online courses), will allow users to develop their programming skills more completely.

Part of the students, according to A. Babori's research [16], may have problems planning their own learning with the help of MOOC. It ultimately affects the quality of learning while taking the online course. At the same time, some of the courses have clear terms of study, a sequence of topics to be studied, and assignments given for the following week [33]. For example, there are online courses on edX and Coursera platforms, which provide regulation of weeks and hours per week during their course.

As noted in works by [13], [15], lack of basic programming knowledge and lack of prior programming experience can prevent beginners from learning programming quickly. Therefore, an important factor in the successful study of an online course is the focus on user training. If we take into account all the analyzed platforms, 788 beginner-level online courses and 571 online courses without level restrictions will be useful for a beginner in JavaScript programming. More advanced programmers are recommended to pay attention to 399 online courses of the Intermediate level. Professionals in programming can improve their skills with 37 online courses at the Advanced level.

There is no doubt that it is users who make their choice of an online resource for learning the JavaScript programming language. After searching for the desired course, the user can familiarize himself/herself with its description by viewing the topics of the modules included in the course. Only after realizing that the course is needed, the user can add it to his/her own account and start studying.

IV. Conclusion

Thus, the obtained results of the quantitative analysis showed that well-known MOOC offer users 1,795 online courses on learning the JavaScript language. Such online platforms as Udemy (1,198 courses, 67% of the total number of online JS courses) and Coursera (457 courses, 25%) offer the most number of courses on this topic. One can take online courses on a paid (1,077 courses, 60%) and free (201 courses, 11%) basis. A blended payment option is also available, offering 517 (28%) online courses.

Almost all the courses located on MOOC platforms are differentiated by the level of user training. Among the entire array of the online JavaScript language courses, 44% (788 courses) are developed for the Beginner level. Advanced programmers are offered 37 online courses, which is 2% of the total number of JavaScript courses. 32% of courses are offered to take without reference to the level of training. Some online platforms, in particular Codecademy and Udemy, allow users to improve the process of developing programming skills through programming projects (67 online courses), coding exercises (99 online courses) and Practice tests (55 online courses).

In our opinion, MOOC has sufficient capabilities to teach users programming with JavaScript. To a greater extent, the completion of the online course depends on the users and their choice of a specific online course. In further prospect, we plan to research students' attitude towards the possibility of using online courses as an additional tool while learning programming at university.

References

[1] V. Voronkova, V. Nikitenko, R. Oleksenko, R. Andriukaitiene, J. Kharchenko, and E. Kliuienko, "Digital Technology Evolution of the Industrial Revolution From 4G to 5G in the Context of the Challenges of Digital Globalization," *TEM Journal*, vol. 12, no. 2, pp. 732–742, May 2023, doi: 10.18421/TEM122-17.

[2] B. S. Rêgo, D. Lourenço, F. Moreira, and C. S. Pereira, "Digital transformation, skills and education: A systematic literature review," *Industry and Higher Education*, vol. 0, no. 0, pp. 1–14, 2023, doi:10.1177/09504222231208969.

[3] A. Androniceanu and I. Georgescu, "Digital competences and human development: a canonical correlation analysis in Romania," *Polish Journal of Management Studies*, vol. 28, no. 1, pp. 43–61, Dec. 2023, doi: 10.17512/pjms.2023.28.1.03.

[4] J. Skalka *et al.*, "Conceptual framework for programming skills development based on microlearning and automated source code evaluation in virtual learning environment," *Sustainability (Switzerland)*, vol. 13, no. 6, pp. 1–30, 2021, doi: 10.3390/su13063293.

[5] K. T. Mon, K. H. Wai, N. Funabiki, and H. H. S. Kyaw, "Value Trace Problems for Code Reading Study of JavaScript Programming," *International Journal of Information and Education Technology*, vol. 12, no. 5, pp. 430–435, May 2022, doi: 10.18178/ijiet.2022.12.5.1637.

[6] S. I. Malik, R. Mathew, A. Al-Sideiri, J. Jabbar, R. Al-Nuaimi, and R. M. Tawafak, "Enhancing problem-solving skills of novice programmers in an introductory programming course," *Computer Applications in Engineering Education*, vol. 30, no. 1, pp. 174–194, Jan. 2022, doi: 10.1002/cae.22450.

[7] S. I. Malik, M. Al-Emran, R. Mathew, R. M. Tawafak, and G. AlFarsi, "Comparison of E-learning, M-learning and game-based learning in programming education," *International Journal of Emerging Technologies in Learning*, vol. 15, no. 15, pp. 133–146, 2020, doi:10.3991/ijet.v15i15.14503.

[8] H. Hiranishi, Y. Imai, A. Ohno, P. Podržaj, and T. Hattori, "Application of Web-based Visual Compiler to Computer Education," *IEEJ Transactions on Electronics, Information and Systems*, vol. 142, no. 3, pp. 389–394, 2022, doi: 10.1541/ieejeiss.142.389.

[9] Q. Batiha, N. Sahari, N. Aini, and N. Mohd, "Adoption of Visual Programming Environments in Programming Learning," *Int J Adv Sci Eng Inf Technol*, vol. 12, no. 5, pp. 1921–1930, 2022, doi:10.18517/ijaseit.12.5.15500.

[10] R. Hosseini *et al.*, "Improving Engagement in Program Construction Examples for Learning Python Programming," *Int J Artif Intell Educ*, vol. 30, no. 2, pp. 299–336, 2020, doi: 10.1007/s40593-020-00197-0.

[11] R. Maskeliūnas, A. Kulikajevas, T. Blažauskas, R. Damaševičius, and J. Swacha, "An interactive serious mobile game for supporting the learning of programming in javascript in the context of eco-friendly city management," *Computers*, vol. 9, no. 4, pp. 1–18, 2020, doi:10.3390/computers9040102.

[12] Y. Zhao, L. Guo, H. Liu, and W. Zheng, "Design of programming experiment course platform based on MOOCs," *International Journal of Emerging Technologies in Learning*, vol. 14, no. 10, pp. 208–216, 2019, doi: 10.3991/ijet.v14i10.10330.

[13] W. Chansanam, K. Poonpon, T. Manaku, and U. Detthamrong, "Success and Challenges in MOOCs: A Literature Systematic Review Technique," *TEM Journal*, vol. 10, no. 4, pp. 1728–1732, 2021, doi:10.18421/TEM104-32.

[14] L. Cilliers, H. Twinomurinzi, and O. Murire, "Motivational Factors that Influence the Course Completion Rate of Massive Open Online Courses in South Africa," *International Journal of Learning, Teaching and Educational Research*, vol. 22, no. 6, pp. 212–225, 2023, doi:10.26803/ijlter.22.6.12.

[15] M. Rõõm, P. Luik, and M. Lepp, "Learner success and the factors influencing it in computer programming MOOC," *Educ Inf Technol (Dordr)*, vol. 28, no. 7, pp. 8645–8663, 2023, doi: 10.1007/s10639-022-11535-1.

[16] A. Babori, "Analysis of Discussion Forums of a Programming MOOC," *TEM Journal*, vol. 10, no. 3, pp. 1442–1446, 2021, doi:10.18421/TEM103-56.

[17] L. Feklistova, M. Lepp, and P. Luik, "Learners' performance in a mooc on programming," *Educ Sci (Basel)*, vol. 11, no. 9, 2021, doi:10.3390/educsci11090521.

[18] M. Säde, R. Suviste, and P. Luik, "Measuring the post-impact of programming MOOCs: Development and validation of an instrument," *Educ Sci (Basel)*, vol. 11, no. 12, Dec. 2021, doi:10.3390/educsci11120811.

[19] C. C. Aguirre, N. González-Castro, C. D. Kloos, C. Alario-Hoyos, and P. J. Muñoz-Merino, "Conversational agent for supporting learners on a mooc on programming with java," *Computer Science and Information Systems*, vol. 18, no. 4, pp. 1271–1286, 2021, doi:10.2298/CSIS200731020C.

[20] E. Kaila and K. Lemström, "To Schedule or Not to Schedule: The Effects of Course Structure on Programming MOOC Performance," *Informatics in Education*, vol. 22, no. 4, pp. 635–650, 2023, doi:10.15388/infedu.2023.27.

[21] N. González-Castro, P. J. Muñoz-Merino, C. Alario-Hoyos, and C. D. Kloos, "Adaptive learning module for a conversational agent to support MOOC learners," *Australasian Journal of Educational Technology*, vol. 37, no. 2, pp. 24–44, 2021, doi: 10.14742/AJET.6646.

[22] S. A. Abdulkareem and A. J. Abboud, "Evaluating Python, C++, JavaScript and Java Programming Languages Based on Software Complexity Calculator (Halstead Metrics)," *IOP Conf Ser Mater Sci Eng*, vol. 1076, no. 1, p. 012046, Feb. 2021, doi: 10.1088/1757-899x/1076/1/012046.

[23] S. Proskura, S. Lytvynova, and O. Kronda, "The use of WEB-oriented Technologies in the Process of WEB-programming Teaching for Technical Universities Students," in *CEUR Workshop Proceedings*, CEUR-WS, 2022, pp. 129–140. doi: 10.31812/educdim.4724.

[24] S. Sharov, S. Tereshchuk, A. Tereshchuk, V. Kolmakova, and N. Yankova, "Using MOOC to Learn the Python Programming Language," *International Journal of Emerging Technologies in Learning*, vol. 18, no. 2, pp. 17–32, 2023, doi:10.3991/ijet.v18i02.36431.

[25] A. Çetinkaya and Ö. K. Baykan, "Prediction of middle school students' programming talent using artificial neural networks," *Engineering Science and Technology, an International Journal*, vol. 23, no. 6, pp. 1301–1307, Dec. 2020, doi: 10.1016/j.jestch.2020.07.005.

[26] H. N. Go, S. W. Kim, and Y. Lee, "Design and Implementation of a Programming Automatic Assessment System in Jupyter Notebook," *Int J Adv Sci Eng Inf Technol*, vol. 13, no. 3, pp. 1080–1086, 2023, doi:10.18517/ijaseit.13.3.18457.

[27] N. R. Ravari, R. Latih, and A. M. Zin, "Multi-Language Program Understanding Tool," *Int J Adv Sci Eng Inf Technol*, vol. 13, no. 4, pp. 1554–1560, 2023, doi: 10.18517/ijaseit.13.4.18019.

[28] A. Birillo *et al.*, "Detecting Code Quality Issues in Pre-written Templates of Programming Tasks in Online Courses," in *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 2023, pp. 152–158. doi:10.1145/3587102.3588800.

[29] K. Litherland and A. Kluge, "Learning to program as empirical inquiry: using a conversation perspective to explore student programming processes," *Computer Science Education*, 2023, doi:10.1080/08993408.2023.2290410.

[30] Z. Kanetaki, C. Stergiou, G. Bekas, C. Troussas, and C. Sgouropoulou, "Analysis of Engineering Student Data in Online Higher Education during the COVID-19 Pandemic," *International Journal of Engineering Pedagogy*, vol. 11, no. 6, pp. 27–49, 2021, doi:10.3991/ijep.v11i6.23259.

[31] R. Daskalov, G. Pashev, and S. Gaftandzhieva, "Hybrid Visual Programming Language Environment for Programming Training," *TEM Journal*, vol. 10, no. 2, pp. 981–986, 2021, doi:10.18421/TEM102-63.

[32] S. Sharov, M. Vorovka, T. Sharova, and A. Zemlianska, "The impact of social networks on the development of students' social competence," *International Journal of Engineering Pedagogy*, vol. 11, no. 3, pp. 84–98, 2021, doi: 10.3991/IJEP.V11I3.20491.

[33] A. Deacon, S. Walji, J. Jawitz, J. Small, and T. Jaffer, "Seizing opportunities: MOOC takers making time for change," *Critical Studies*

in Teaching and Learning, vol. 7, no. Special Issue, pp. 20–37, Oct. 2019, doi: 10.14426/cristal.v7iSI.203.

[34] S. White, S. White, and K. Borthwick, "MOOCs, learning designers and the unbundling of educator roles in higher education," *Australasian Journal of Educational Technology*, vol. 36, no. 5, pp. 71–84, 2020, doi: 10.14742/ajet.6111.

[35] C. T. Swai, Q. Liu, and L. Wu, "Teachers' professional growth on SCT strategies: Mwalimu hub MOOC analytics," *Interactive Learning Environments*, pp. 1–15, 2023, doi: 10.1080/10494820.2023.2174141.

[36] D. Jaramillo-Morillo, M. Solarte, and G. Ramírez-González, "Extraction of Student Interaction Data from an Open edX Platform," *Revista Ingenierías Universidad de Medellín*, vol. 20, no. 38, pp. 87–98, Mar. 2021, doi: 10.22395/rium.v20n38a5.

[37] S. Sharov, V. Kolmakova, T. Sharova, and A. Pavlenko, "Analysis of MOOC on Programming for IT Specialist Training," *TEM Journal*, vol. 10, no. 4, pp. 1884–1894, 2021, doi: 10.18421/TEM104-52.