Programming and scripting languages are the foundation of the development of software and web pages and applications. They are the main tools that developers use to communicate directions to computers. This allows anyone to create all kinds of software, such as websites, phone applications, computer systems and computer games. (htt)

Some examples of programming languages include Rust, Java, C++, Go, and SQL. Example of scripting languages include Ruby, JavaScript, PHP, Python, and Bash. Some languages such as Python and JavaScript can be used as both programming and scripting languages meaning there would be an overlap between the two types of languages. (htt1) (htt2)

The benefits of using programming languages are that they run quickly and efficiently, they have strong type checking, meaning they help catch errors and make sure the data is used correctly, and they are great for big complex projects. Some drawbacks may be that it takes more time to develop software because it’s slower to write and it will include a more complex syntax, making the code harder to write and understand. The benefits of scripting languages include simple syntax making it easier to learn and use, quick to write so it’s great for getting things done fast and it is perfect for automation like controlling web pages and writing small scripts. Some drawbacks of using scripting languages would be that they are not as fast as programming languages, and they usually can’t interact directly with the hardware therefore they have less access to the system. (htt3) (htt2) (htt1)

Programming languages are often used to build software, games, and operating systems. Scripting languages are great for automating tasks, running system tools, and making websites more interactive. (htt3)

Server-side scripting languages operate on the web server, generating dynamic content before it is sent to the user's browser. Common examples include PHP and Node.js (JavaScript). They work with databases, manage things like logins, and change the website based on what the user does. (htt4)

Server-side scripting is important because it lets websites show dynamic and personalized content. It keeps sensitive tasks on the server, organized websites, and connects to databases. If we didn’t have websites like social media and online stores wouldn’t work. (htt4) (htt5)

If I were to start a new language today, I would choose JavaScript, because it has dual capabilities where it can be used for both programming languages as well as scripting languages. I also like the fact that it works with both client- and server-side development and has a large community support. (htt4) (htt5)

# References

(n.d.). Retrieved from https://www.unosquare.com/blog/scripting-and-programming-languages-differences-advantages-and-optimal-use-cases/#:~:text=Programming%20languages%20are%20often%20more,device%20drivers%20and%20system%20utilities.

(n.d.). Retrieved from https://www.computerscience.org/resources/computer-programming-languages/

(n.d.). Retrieved from https://www.indeed.com/career-advice/career-development/top-scripting-languages

(n.d.). Retrieved from https://www.techtarget.com/searchapparchitecture/tip/Scripting-vs-programming-languages-Where-they-differ

(n.d.). Retrieved from https://www.veltris.com/blogs/digital-engineering/server-side-scripting-languages/#:~:text=What%20is%20Server%2DSide%20Scripting,that%20are%20responsive%20and%20secure.

(n.d.). Retrieved from https://www.upwork.com/resources/server-side-scripting-back-end-web-development-technology