Task 2

Research and prepare a document on the following topics.

• Understand IDE : VS Code

➢ IDE

An **Integrated Development Environment (IDE)** is a software application that helps programmers develop software code efficiently. It increases developer productivity by combining capabilities such as software editing, building, testing, and packaging in an easy-to-use application. Just as writers use text editors and accountants use spreadsheets, software developers use IDEs to make their job easier. Some examples of IDEs are Eclipse, IntelliJ idea, visible Studio, NetBeans, Phpstorm, and WebStorm.

The tools available in an IDE

- **Authoring:** The first tool for developing code is the ability to write the new code directly into the IDE's platform. IDEs provide this tool, and some can work with many coding languages.
- **Modifying:** The programmers can modify, or update, code within the same IDE program. Modifications occur in the plain text version of the code.
- **Compiling:** Compiling code translates it from readable code into machine code, and an IDE expedites this process.
- **Deploying:** After the compiling phase, you can deploy the new code through the IDE. This activates the code, meaning users can interact with it or see the changes.
- **Debugging:** IDEs typically include a debugging tool that removes errors from the code as you write it, which can save time and energy.
- **Syntax highlighting:** Because the IDE knows the intricacies of programming languages, it highlights certain bits of code to provide visual cues. This helps with authoring, modifying, and debugging.
- **Auto-completion:** The IDE's familiarity with programming language allows it to predict what code you might enter next. Auto-completion helps the programmer focus on more important elements of writing code.

The benefits of using IDEs

Developers, programmers, and software engineers spend a lot of time writing and editing code. Without IDEs, they may use different programs to accomplish each task, which requires transferring and reformatting the code to a new program. However, IDEs streamline and expedite the processes of authoring, modifying, and deploying code. Some other benefits of IDEs include:

- **Increases developer productivity:** The number of tools available through an IDE increases the productivity of developers because they can use one program to accomplish most of their tasks. This can allow them to accomplish more each day.
- **Easier to learn:** With every tool in the same program, new employees can learn the company's coding style and primary tools much quicker. It also may be easier for a new developer to learn one program instead of multiple for each stage of a project.
- **Provides instant feedback:** IDEs typically include debugging software, so the program provides instant feedback as the developer writes code. This helps the developer multitask.
- **Allows greater access**: IDEs allow programmers to download, code and run programs from a personal computer. This can let them perform remote software development work because each developer can access the program from their computer.

Types of IDEs

Several IDE programs exist for a specific purpose, company, or development focus. Companies that use code may have different programming needs that affect the IDE they implement in their workflow. Here are some types of IDEs you might use in a development or programming position:

1. **Multi-language**
   A multi-language IDE is the most common type, and these programs support multiple programming languages. Some of these IDEs limit the number of languages available, while others are highly custom. Each multi-language IDE serves a specific function, so a company can invest in the program that best suits its needs.
2. **Mobile development**
   These IDEs are specifically for mobile development, such as coding for smartphones. Often, a multi-language IDE can transition to mobile development. These IDEs contribute to the development of applications for users to download onto their mobile devices. Often, these IDEs are specific to a technology brand or company to ensure the code works cohesively on its specific devices.
3. **Cloud-based**

Cloud-based IDEs, or web-based IDEs, operate online instead of connecting to a specific location. This provides developers with access to their code from anywhere if they have a stable internet connection. Cloud-based IDE usage often increases as the functions become more advanced.

4. **HTML**
   HTML IDEs focus specifically on web development. Web design uses HTML, so professionals who primarily design websites may benefit from this IDE. HTML IDEs are among the most popular because web creation and design are common programming specialties. These IDEs may overlap with cloud-based IDEs, making them accessible from anywhere.
5. **Language-specific**
   The opposite of the multi-language type, language-specific IDEs function with only one programming language. These IDEs may be good for companies that work with one type of programming and rarely access other languages through their IDE. Language-specific IDEs often have specialized add-ons or features that allow you to program more effectively in that language.

➢ VS Code

Visual Studio Code (VS Code) is a free open-source code editor for development and debugging modern cloud and web applications which is available for free on Linux, OS X and Windows. VS Code supports more than 30 different programming, markup, and database languages, some of which are JavaScript, C#, C++, PHP, Java, HTML, R, CSS, SQL, Markdown, TypeScript, Less, Sass, JSON, XML and Python.

Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times.

Key Features

- **Code Editing:** VS Code provides syntax highlighting, autocompletion, and code formatting for a variety of programming languages, making it easy for developers to write and edit code.
- **Integrated Terminal:** VS Code comes with an integrated terminal, allowing developers to execute commands and run scripts without having to leave the editor.
- **Git Integration:** VS Code has built-in support for Git, allowing developers to easily manage source code changes and collaborate with other developers.

- **Debugging:** VS Code has a powerful debugger that supports multiple languages and allows developers to set breakpoints, step through code, and examine variables.
- **Extensions:** VS Code has a rich ecosystem of extensions that provide additional functionality and support for specific programming languages, frameworks, and tools.
- **IntelliSense**: VS Code provides intelligent code completion, code suggestions, and parameter hints for a wide range of languages and frameworks, making it easier for developers to write code quickly and accurately.
- **Live Share:** VS Code has a live share feature that allows developers to collaborate in real time on the same codebase, regardless of their location.
- **Task Runner:** VS Code has a built-in task runner that allows developers to automate common tasks, such as building, testing, and deploying code.
- **Snippets:** VS Code has snippets feature that allows developers to save and reuse commonly used blocks of code, saving time and improving code consistency.
- **Customizable User Interface:** VS Code has a customizable user interface that allows developers to personalize their coding environment with themes, icons, and other preferences.