

## SETTING UP DOCKER FOR PROGRAMMING ENVIRONMENT

**Explanation:** Docker is used to create a virtual environment that standardizes the programming setup across different operating systems. This ensures that all users, regardless of their OS, can work in a consistent environment without compatibility issues.

### Key Points:

- Docker creates a virtual computer on your machine.
- Instructions for setup are available on the CSS wiki.
- Docker images are the basis for creating virtual machines.
- The command `docker pull CSS images/Linux Lab` downloads the necessary image.
- The image size is approximately 3.1 gigabytes.

## CREATING AND CONFIGURING DOCKER CONTAINERS

**Explanation:** After downloading the Docker image, a container is created to run the virtual machine. This container allows access to files from both Docker and the host PC.

### Key Points:

- Create a permanent directory to avoid deletion by the system.
- Use the command `docker run -v [local directory]:[container directory] -it CSS images/Linux Lab /bin/bash` to create and configure the container.
- The container maps directories between the host machine and the virtual machine.
- Ensure the container is running to connect it with development tools like Visual Studio Code.

## USING VISUAL STUDIO CODE WITH DOCKER

**Explanation:** Visual Studio Code can be configured to work with Docker containers, allowing seamless development within the virtual environment. This setup includes installing necessary extensions and connecting to the running Docker container.

### Key Points:

- Install the Remote Development extension in Visual Studio Code.
- Use the "Open a remote window" command to attach to the running Docker container.
- Create a project folder within the virtual machine for organized development.
- Visual Studio Code can open and manage files directly from the Docker container.

## COMPILING AND RUNNING CODE IN DOCKER

**Explanation:** Code can be compiled and executed within the Docker container using terminal commands. This process includes setting up necessary tools and ensuring proper file management.

### Key Points:

- Use terminal commands like `g++ hello.cpp` to compile code.
- Docker images come pre-installed with tools like GCC, Valgrind, Clang Format, and Clang Tidy.
- Create and manage files within the Docker container using terminal commands.

- Ensure proper formatting and debugging by configuring Visual Studio Code settings.

## DEBUGGING AND FORMATTING CODE

**Explanation:** Debugging and formatting are essential parts of the development process. Visual Studio Code provides tools to set breakpoints, format code, and manage project settings.

### Key Points:

- Set breakpoints and conditional breakpoints for effective debugging.
- Use Clang Format for consistent code formatting.
- Configure Visual Studio Code settings and tasks files for project management.
- Ensure all necessary files are included in the compilation process.

## MANAGING PROJECT FILES

**Explanation:** Proper management of project files ensures smooth development and collaboration. Docker allows access to files from both the host machine and the virtual environment.

### Key Points:

- Create and manage directories within the Docker container.
- Access project files from both the host machine and the virtual machine.
- Use Visual Studio Code to manage and edit project files.

- Ensure all files are properly included in the compilation and debugging process.