as

Name: 12

Roll Number: 12

Q. 1

```
const { Command, flags } = require("@oclif/command");
const { mdToPdf } = require("md-to-pdf");
const fs = require("fs");
const prompt = require("prompt-sync")();

const MarkdownContent = (
    assignment_name,
    student_name,
    roll_number,
    dir_name,
    file_list
) => {
    const header = `# ${assignment_name}\n`;
    const title = `Name: ${student_name} \n\n Roll Number: ${roll_number}\n\n`;

    let program_content = "";

    for (let i = 0; i < file_list.length; i++) {
        const data = fs.readFileSync(`${dir_name}/${file_list[i]}`);

        program_content += `### Q. ${i + 1} \n`;
        program_content += "```\n";
        program_content += data;
        program_content += "\n```";
    }

    return `${header} ${title} ${program_content}`;
};

class CodeToPdfCommand extends Command {
    async run() {
        const { flags } = this.parse(CodeToPdfCommand);

        const dir = flags.dir;
        const output = flags.output || "assignment";

        if (dir) {
            fs.readdir(dir, async (err, files) => {
                const assignment_name = prompt("Assignment Name: ");
                const name = prompt("Enter Name: ");
                const roll_no = prompt("Enter Roll Number: ");

                const md = MarkdownContent(
                    assignment_name,
```

```
                    name,
                    roll_no,
                    dir,
                    files
                );

                fs.writeFileSync(`${output}.md`, md);

                const pdf = await mdToPdf({ path: `${output}.md`
}).catch(console.error);
                if (pdf) {
                    fs.writeFileSync(`${output}.pdf`, pdf.content)
                }
            });
        } else {
            this.log("Bruh, mention a directory");
        }
    }
}

CodeToPdfCommand.description =
    "Convert a bunch of files to an Assignment Submission";

CodeToPdfCommand.flags = {
    version: flags.version({ char: "v" }),
    help: flags.help({ char: "h" }),
    dir: flags.string({ char: "d", description: "Directory with all Files" }),
    output: flags.string({ char: "o", description: "Output File Name" }),
};

module.exports = CodeToPdfCommand;
```